

Article

PROLIFIC: Deep Reinforcement Learning for Efficient EV Fleet Scheduling and Charging

Junchi Ma *, Yuan Zhang, Zongtao Duan and Lei Tang

School of Information Engineering, Chang'an University, Xi'an 710061, China; 2021224012@chd.edu.cn (Y.Z.); ztduan@chd.edu.cn (Z.D.); tanglei24@chd.edu.cn (L.T.)

* Correspondence: majunchi@chd.edu.cn

Abstract: Electric vehicles (EVs) are becoming increasingly popular in ride-hailing services, but their slow charging speed negatively affects service efficiency. To address this challenge, we propose PROLIFIC, a deep reinforcement learning-based approach for efficient EV scheduling and charging in ride-hailing services. The objective of PROLIFIC is to minimize passenger waiting time and charging time cost. PROLIFIC formulates the EV scheduling problem as a Markov decision process and integrates a distributed charging scheduling management model and a centralized order dispatching model. By using a distributed deep Q-network, the agents can share charging and EV supply information to make efficient interactions between charging and dispatch decisions. This approach reduces the curse of dimensionality problem and improves the training efficiency of the neural network. The proposed approach is validated in three typical scenarios with different spatiotemporal distribution characteristics of passenger order, and the results demonstrate that PROLIFIC significantly reduces the passenger waiting time and charging time cost in all three scenarios compared to baseline algorithms.

Keywords: electric vehicles; charging scheduling; reinforcement learning; ride-hailing services; charging stations



Citation: Ma, J.; Zhang, Y.; Duan, Z.; Tang, L. PROLIFIC: Deep Reinforcement Learning for Efficient EV Fleet Scheduling and Charging. *Sustainability* **2023**, *15*, 13553. <https://doi.org/10.3390/su151813553>

Academic Editor: Lei Zhang

Received: 3 August 2023

Revised: 29 August 2023

Accepted: 8 September 2023

Published: 11 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, EVs have emerged as a hot research topic with the potential to save energy and reduce CO₂ emissions [1,2]. In fact, the global sales of new EVs reached 10.5 million in 2022, which was a remarkable increase of 55% compared to the previous year. This surge in demand for EVs can be attributed to several factors, including government policies aimed at reducing greenhouse gas emissions, advancements in battery technology, and the growing awareness of the benefits of EVs [3]. Large companies such as Uber have also announced plans to transition their fleet of oil cars to EVs, with a goal of achieving zero emissions in US and Canadian cities by 2030. The rise of intelligent vehicle technologies further amplifies the potential of EVs in modern transportation systems [4–7].

Despite the many advantages of EVs, such as reduced emissions and lower operating costs, one of the major challenges is the time it takes to refuel the vehicle's battery. For example, charging a BYD Qin from empty to full can take up to one hour even when using a 60 kW fast charger, which is considerably longer than the time it takes to fill up a tank of gasoline. Additionally, the number of charging stations available is still relatively low in many areas, leading to long wait times for charging or even unavailable charging options when the state of charge (SOC) of an EV is low [8]. These limitations in charging speed and availability can significantly impact not only the ease of use and adoption of EVs but also the efficiency of applications that rely on them such as ride-hailing services provided by companies like Uber.

Efficiently scheduling the charging and order dispatch in an EV fleet is a critical challenge for ride-hailing service providers. The large time required for EV charging can

interfere with ride services, resulting in longer wait times for passengers during rush hours. Moreover, improper charging management can lead to insufficient energy for transportation, causing EVs to refuse ride requests. This issue is crucial to improving the efficiency of ride-hailing service providers, as longer wait time and fewer available cars can lead to decreased customer satisfaction and lower profits.

Prior research has endeavored to provide pertinent recommendations for EV users, focusing primarily on factors such as charging time cost [9–13] and electricity price [14–16]. Battery swap station recommendations have also gained attention, often emphasizing path or electricity cost due to the notably shorter time associated with swapping compared to traditional charging stations [17–19]. Furthermore, a subset of recent studies has begun to incorporate charging path optimization into the decision-making process for selecting charging stations while considering energy constraints and recuperation [20–22]. A parallel stream of research has concentrated on the efficient scheduling and charging of electric vehicle fleets within ride-hailing services [23–26]. Our research aligns with this stream, specifically addressing the formulation of optimal actions for each vehicle in the fleet to achieve overarching goals, such as maximizing gross merchandise volume or minimizing passenger waiting time. Reinforcement learning has emerged as the mainstream approach to tackle these challenges [23–27]. These studies typically address charging and dispatch decisions in isolation, or they adopt a simplified assumption that the nearest charging station is invariably the optimal choice. This approach overlooks the intricate interdependencies between charging and dispatching decisions, potentially leading to suboptimal solutions that fail to minimize passenger waiting time and charging time effectively. The real challenge resides in finding a balance between charging and dispatch decisions within the context of EV ride-hailing services. Each decision process has its unique considerations, and the interaction between them can markedly affect the overall efficiency of the service. Therefore, there is a pressing need for more refined optimization strategies capable of addressing the combined challenges of charging and dispatch decisions in the realm of EV ride-hailing services.

This paper proposes PROLIFIC (deeP ReinfOrmance LearnIng For effIcient ev fleet scheduling and Charging). PROLIFIC is designed to minimize the passenger waiting time and charging time cost. The proposed algorithm takes into account both dispatch tasks and charging plans, using a deep Q-network (DQN) to make efficient long-term decisions. Given a fleet of EVs with arbitrary initial locations and battery levels, the proposed algorithm dispatches them to serve a set of unknown customer trip requests and outputs the specific charging station that each EV should use, which enables further optimization of the charging time cost. This level of optimization is not achievable by prior work, which only considered the nearest charging station or made recommendations based on static criteria. Moreover, the customer waiting time is minimized by optimizing dispatch tasks, while the charging time cost is reduced by selecting the most efficient charging stations and plans. To use DQN for EV fleet scheduling, the algorithm is trained using historical data to learn the optimal policy for dispatching and charging EVs. This policy specifies the best sequence of actions to take in response to different states of the system, such as the locations and battery levels of the EVs and the trip requests from customers.

Our contributions can be summarized as follows:

1. We propose PROLIFIC based on deep reinforcement learning that coordinates the order dispatch and charging behavior of EVs, enabling them to make optimal decisions on charging time and station selection while satisfying the ride-hailing needs of passengers.
2. Using distributed reinforcement learning decision making allows the agents to share information such as the utilization rate of charging piles. This communication helps to reduce resource conflicts and increase the efficiency of the charging process.

3. The simulation experiments conducted in three distinct scenarios showcases PRO-LIFIC's ability to effectuate a 61.56% to 78.84% reduction in charging waiting times and a 34.05% to 56.45% reduction in passenger waiting times in contrast to baseline models, highlighting its potential to improve the efficiency of ride-hailing companies and other EV applications.

2. Related Work

In recent years, researchers have focused on the problem of recommending optimal charging stations for EVs. The aim is to minimize the time and cost associated with charging by considering constraints such as the EV's remaining electricity and the availability of charging piles. A series of work formalized the scheduling problem as a Markov decision process (MDP) and proposed deep reinforcement learning algorithms for charging station recommendation. Zhang proposed a reinforcement learning model in which the state domain represented the available fast and slow charging piles, and the action domain represented the choice of charging station and pile [9]. DQN was then applied to compute the Q value for each charging station and pile, and the solution with the highest Q value was chosen as optimal. Li proposed a Regularized Actor–Critic approach that allowed EV drivers to balance user preference and external reward [10]. Additionally, some researchers have used representation learning techniques to improve the understanding of the transportation status. Zhang applied Spatiotemporal Heterogeneous Graph Convolution to create a dynamic station–vehicle graph and added the representation to the agent's state for better modeling of the relationship between EVs and charging stations [28]. In addition, many other computational methods and optimization strategies have been applied to this problem to find approximate solutions. An proposed a particle swarm-based optimization algorithm in which individual and global extremes are calculated in iterations, and the policy is updated to obtain the optimal solution [29]. Liu solves the charging scheduling problem by utilizing a genetic algorithm that defines the fitness function to include the time cost, the monetary cost, and the final residual power. The residual power is calculated iteratively using a differential evolutionary algorithm [30].

Several researchers have also considered the joint problem of order dispatch and charging operations. Shi proposed RLOAR that used decentralized learning and centralized decision making to optimize the scheduling of the ride-hailing fleet [23]. In this approach, the order information and EV status were collected in a time slot, and the Q value was calculated using the Bellman equation. The problem of maximizing the Q value of the fleet was then solved by transferring it to a bipartite graph matching problem, which was solved using the Kuhn–Munkres algorithm. Kullman proposed a Drafter algorithm in which every ride-hailing order immediately triggered a scheduling process [24]. While this approach compressed the dimension of order dispatching compared to time-slot-based approaches, the frequency of decision making was more intensive. Tang proposed an advisor–student reinforcement learning framework to optimize the online operations of EV fleets [25]. In this approach, the advisor model determined the number of taxis to serve the user demands and to charge in each area. The student model then used these results as a reference to generate movement based on the combinational optimization model. This allowed the problem to be decomposed into multiple steps, which accelerated the convergence of the large-scale model. Yu proposes an asynchronous learning framework [31]. The algorithm is primarily controlled by two loops. The outer loop calculates the Q-value of each action using the Q-Learning algorithm. The inner loop determines the vehicle's action choices, including order taking, repositioning, and recharging, based on the outputs of the outer loop, combined with stochastic approximation. Zalesak proposed a heuristic-based charging strategy with the objective of maintaining sufficient fleet sizes at various times of day [32]. In their approach, vehicles were scheduled based on their SOC, and the vehicle with the highest SOC was scheduled to charge with the lowest priority. Guo proposes a deep reinforcement learning-based optimization strategy to improve the quality of service degradation of the fleet and reduce the operating cost [33]. The method utilizes a deep reinforcement learning

network with convolutional neural layers to understand environmental states and actions for effective vehicle routing decisions and vehicle scheduling. Yi categorized the EVs by their SOC, and for those EVs with an SOC beyond a threshold, they were scheduled to reposition or charge using dynamic programming to minimize the overall time cost [34]. Yuan proposed proactive partial charging, which allowed an EV to become partially charged before its remaining battery level was running too low [35]. They employed a receding horizon optimization approach to minimize the idle driving time and waiting time.

In previous approaches that attempted to address both order dispatch and charging operations, certain limitations existed, particularly in their treatment of the charging operation. A pivotal factor influencing effective EV fleet scheduling and charging pertains to the attributes of charging stations. Variability in charging powers, accessibility of charging infrastructure, and their spatial distribution significantly impact the decision-making process for selecting optimal charging stations. Notably, prior methodologies often oversimplified the charging operation by consistently favoring the nearest charging station as the optimal choice. This approach, however, may not consistently yield the shortest waiting times for EVs. Unlike such rudimentary proximity-based strategies, the situation underscores the imperative for more sophisticated optimization techniques that account for the distinct characteristics of diverse charging stations.

Moreover, the interplay between order dispatching and charging decisions bears considerable influence on the overall performance. Although certain studies have concentrated on optimizing these decisions in isolation [32,34], neglecting their intricate interdependencies can lead to suboptimal results. This underscores the significance of devising more nuanced optimization strategies that comprehensively address both these aspects within the realm of EV ride-hailing services.

In contrast, this paper proposes a novel approach that considers the charging operation and order dispatch together, and it makes more finely granulated decisions regarding the selection of charging stations. By taking into account the predicted waiting times and balancing the trade-off between waiting time and charging time cost, our approach enables EVs to choose the charging station for their individual needs, leading to improved efficiency and customer satisfaction.

3. Preliminaries

In this section, we introduce necessary notations to describe the problem of EV scheduling. The symbols are listed in Table 1.

Definition (Road Network): The road network is defined as the strongly connected component of the directed graph $G(V^{road}, E^{road})$, where V^{road} and E^{road} denote the sets of all vertices and edges, respectively.

Definition (EV Fleet): The EV fleet is defined as the set $F = \{ev_1, ev_2, \dots\}$, where ev_i denotes the i th EV. $SOC_t^i \in [0, 1]$ is defined as the percentage of the remaining battery capacity of the i th EV at time t relative to its total battery capacity. $loc_t^i = (Lng_t^i, Lat_t^i)$ is defined to represent the position of the i th EV at time t , with Lng_t^i denoting longitude and Lat_t^i denoting latitude.

Definition (Passenger Waiting Time): The passenger waiting time Δt_{wait}^l can be defined in two cases. If the order is successfully received, Δt_{wait}^l is the time difference between the moment the EV responds to the order and the moment the order is initiated. However, if the order is canceled due to timeout, Δt_{wait}^l is defined as the maximum waiting time t_{PET} .

Definition (EV supply rate): To represent the relationship between the EV supply and order demand, we define the EV supply rate ϕ_p as follows:

$$\phi_p = |F_t^*| / (|O_t^*| + 1) \quad (1)$$

where F_t^* represents the set of EVs with a battery level greater than $\wp$ and available for decision making at time t . $\wp$ represents the threshold for sufficient battery power. $|O_t^*|$ represents the number of pending orders at time t .

Table 1. Symbol descriptions.

Symbol	Description
O	The set of all orders in the entire simulation period
O^*	The set of all orders served during the entire simulation period
O_t	The set of all pending orders at time t
O_t^*	The set of all pending and unexpired orders at time t
t_{PET}	The maximum waiting time for passengers after initiating an order
Δt_{wait}^l	The waiting time of the l th order
F	The set of all ride-hailing vehicles in the entire fleet
F_t	The set of idle ride-hailing vehicles at time t
F_t^*	The set of idle and sufficiently charged ride-hailing vehicles at time t
$\bar{F}_t^*$	The set of ride-hailing vehicles dispatched to respond to passenger requests at time t
C	The set of all charging stations
cs_j	The j th charging station in the charging system
cp_k^j	The k th charging area in the j th charging station
t_{dec}^i	The decision time of the i th ride-hailing vehicle
$t_{arr}^{i,j,k}$	The time when the i th ride-hailing vehicle arrives at the k th charging area of the j th charging station
$t_{ch}^{i,j,k}$	The time when the i th ride-hailing vehicle starts charging in the k th charging area of the j th charging station
$\tau_n^{i,j,k}$	The time when the n th charging pile in the corresponding charging area is released for the i th electric vehicle
$\Delta t_{tra}^{i,j,k}$	The travel time of the i th vehicle to the k th charging pile area of the j th charging station
$\Delta t_{wait}^{i,j,k}$	The queue waiting time of the i th vehicle in the k th charging pile area of the j th charging station
$\Delta t_{ch}^{i,j,k}$	The net charging time of the i th vehicle in the k th charging pile area of the j th charging station
$\Delta t_C^{i,j,k}$	The total time spent by the i th vehicle to go to the k th charging pile area of the j th charging station for charging
$\bar{v}_i$	Represents the average driving speed of the i th electric vehicle
$v_{j,k}$	The actual charging power of the k th charging pile area of the j th charging station
SOC_t^i	The ratio of the remaining power of the i th vehicle at time t to the entire battery
loc_t^i	The location of the i th vehicle at time t
ρ_t	The progress of the entire round corresponding to time t
$\mathcal{T}$	The set of discrete time steps
Y_t	The utilization rate of all charging areas at time t
$\wp$	The lower limit of the power state for ride-hailing vehicles to provide ride services
ξ	The critical value of whether the supply of ride-hailing vehicles is saturated
ϕ_p	The supply rate of ride-hailing vehicles in the entire operating area

Definition (Charging Station System): $C = (cs_1, cs_2, \dots)$, where cs_j represents the j th charging station. Each charging station cs_j is further divided into multiple charging areas with different power levels, which can be denoted as $cs_j = (cp_1^j, cp_2^j, \dots)$. Each charging area cp_k^j in the j th charging station contains $|cp_k^j|$ charging piles of the same power level.

Definition (Travel Time Cost): The travel time cost $\Delta t_{tra}^{i,j,k}$ is calculated as the distance d between the EV's current location and the designated charging area j , which is divided by the EV's average speed $\bar{v}_i$. In other words, it represents the estimated time it takes for the EV to travel to the charging area.

Definition (Charging Waiting Time Cost): The charging waiting time cost is calculated as follows:

$$\Delta t_{wait}^{i,j,k} = t_{ch}^{i,j,k} - t_{arr}^{i,j,k}, \quad (2)$$

where $t_{ch}^{i,j,k}$ denotes the moment when the i th EV starts charging and $t_{arr}^{i,j,k}$ denotes the arrival time to the charging station j . $t_{ch}^{i,j,k}$ depends on the length of the charging queue, which can be divided into two situations: (1) if available charging piles can be found in the k th charging area, the EV can start to be charged directly; (2) if all charging piles in the k th charging area are occupied, the EV has to wait in the queue until a charging pile is available. Thus, we can calculate $t_{ch}^{i,j,k}$ as follows:

$$t_{ch}^{i,j,k} = \begin{cases} t_{arr}^{i,j,k} & p_t^{i,j,k} \leq |cp_k^j| \\ \max\left(t_{arr}^{i,j,k}, \min\left(\tau_1^{i,j,k}, \dots, \tau_{|cp_k^j|}^{i,j,k}\right)\right) & p_t^{i,j,k} > |cp_k^j| \end{cases} \quad (3)$$

where $p_t^{i,j,k}$ denotes the position in the queue when the i th EV arrives in the k th charging area. In situation (1), represented by $p_t^{i,j,k} \leq |cp_k^j|$, the EV can be charged directly without waiting in line. In situation (2), represented by $p_t^{i,j,k} > |cp_k^j|$, the EV has to wait. Here, $\tau_n^{i,j,k}$ indicates the moment when the n th charging pile in the k th charging area becomes available for charging, and the i th EV is waiting in the queue. $\min\left(\tau_1^{i,j,k}, \dots, \tau_{|cp_k^j|}^{i,j,k}\right)$ indicates the earliest finish charging time among the latest $|cp_k^j|$ EVs that arrive in the k th charging area before the i th EV.

Definition (Pure Charging Time Cost): After starting to charge, the time required for charging is calculated using the following equation:

$$\Delta t_{ch}^{i,j,k} = (E_{recov} - E_{i,j,t}) / v_{j,k}, \quad (4)$$

where E_{recov} denotes the battery state when the charging is finished. $E_{i,j,t}$ represents the battery state of the EV when arriving at the j th charging station, and $v_{j,k}$ denotes the charging power of the charging pile in the k th charging area.

Definition (Utilization rate): Y_t denotes the utilization rate of various power charging piles in all charging stations at time t . In $Y_t = (Y_t^{1,1}, \dots, Y_t^{j,k}, \dots)$, $Y_t^{j,k}$ represents the utilization rate of individual charging piles with the specific formula as follows:

$$Y_t^{j,k} = (N_{ch}^{j,k,t} + N_{wait}^{j,k,t}) / |cp_k^j| \quad (5)$$

where $N_{ch}^{j,k,t}$ denotes the number of EVs being charged in the k th charging area in the j th charging station at time t , and $N_{wait}^{j,k,t}$ denotes the number of EVs queuing.

4. Method

In this section, PROLIFIC is presented to solve the EV charging and dispatching problem. We will introduce the system overview first and explain the charging scheduling management model and the order dispatching model.

4.1. System Overview

To coordinate the charging needs of EVs with passengers' ride requests, we propose a comprehensive framework for EV order dispatching and charging scheduling from a fleet perspective. As shown in Figure 1, this framework consists of two main components: the charging scheduling management model and the order dispatching model. The charging scheduling management model applies a deep reinforcement learning (DRL)-based charging scheduling strategy. The agent determines whether to charge for idle ride-hailing vehicles in sequence based on the DRL charging scheduling strategy. Vehicles that need

to be charged proceed to the designated charging area according to the agent's decision. Vehicles that do not need charging participate in the centralized order assignment process. By modeling it as an assignment problem and solving it using the Kuhn–Munkres algorithm [36], the most appropriate matching result is obtained.

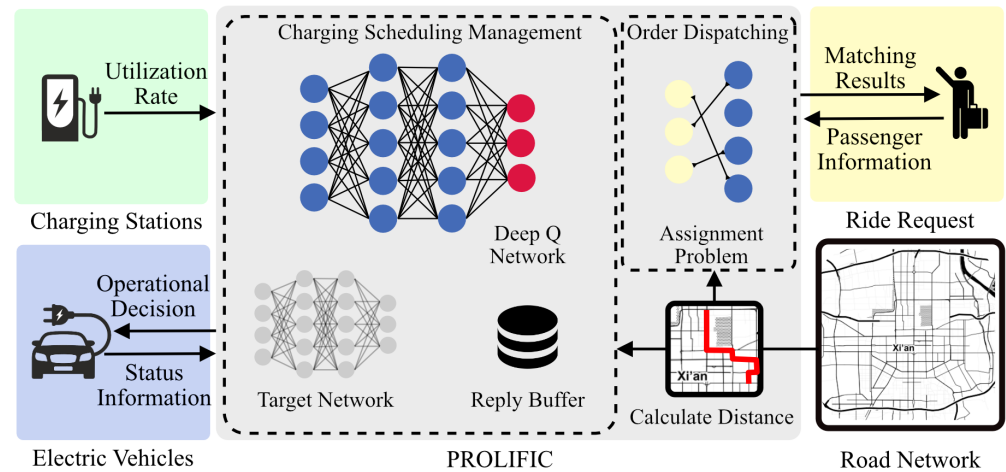


Figure 1. The overall framework of PROLIFIC.

Next, the DRL-based charging scheduling model and the centralized order dispatching model are introduced in detail. We model the real-time charging and order-dispatch process of EVs as a finite discrete-time MDP. A five-tuple $(S, A, P(\cdot, \cdot), R(\cdot, \cdot), \gamma)$ is applied to represent the MDP. Here, S denotes the state space, A refers to the action domain, $P(\cdot, \cdot)$ denotes the state transition probability, $P_a(s, s')$ is the probability that state s transitions to state s' after performing action a , $R(\cdot, \cdot)$ denotes the immediate reward, and $R_a(s, s')$ refers to the immediate reward received after state s transitions to state s' through action a .

Specifically, at time step t , the agent derives the decision result A_t based on the system state S_t , and the EV chooses the charging station or forgoes charging according to the decision result. Subsequently, the fleet, orders, and charging station modules undergo changes. The agent receives the reward R_t and can simultaneously observe the new system state S_{t+1} . This process continues, and the accumulated rewards over time are referred to as returns. The return U_t corresponding to time step t is calculated by $U_t = \sum_{n=t}^K \gamma^{n-t} R_n$, where $\gamma \in [0, 1]$ is the discount factor.

(1) State Space: The state of the system at time t is defined as $s_t = (s_t^1, s_t^2, \dots)$. s_t^i refers to the state vector of the i th EV at time t , which is represented as a tuple $s_t^i = (SOC_t^i, loc_t^i, \rho_t, \phi_p, Y_t)$.

- SOC_t^i denotes the remaining battery state.
- loc_t^i represents the position of the EV.
- $\rho_t \in [0, 1]$ represents the progress of the entire episode, i.e., the proportion of the current elapsed time to the total duration of an episode.
- ϕ_p represents the relationship between the EV supply and at time t .
- Y_t represents the utilization rate of charging piles.

(2) Action Domain: The action of the system at time t is defined as $a_t = (a_t^1, a_t^2, \dots)$, where a_t^i refers to the action vector of the i th EV at decision moment t . Each EV can take one of the following two actions:

- If the decision result is to charge, $a_t^i = \chi_t^i$, where χ_t^i denotes the selected charging area number.
- If the decision is not to charge, $a_t^i = 0$. The EV can accept orders only when it decides not to charge, and if it chooses not to accept orders and the system does not assign orders to that EV, then the EV remains in a waiting state at its current location.

(3) Reward Function: The system's reward function is defined as $R_{a_t^i}(s_t^i, s_{t+1}^i)$, representing the reward obtained by the i th EV after taking action a_t^i . The goal of this paper is to minimize both passenger waiting time and EV charging time costs while adhering to specific constraints. To achieve this, EVs with sufficient battery are actively encouraged to accept orders and incentivize them to choose charging stations with minimal time costs when there are a significant number of idle vehicles compared to pending orders. By doing so, we aim to maintain an adequate number of available EVs for passengers while ensuring that vehicles are charged efficiently and effectively. The reward functions are designed as follows.

The reward for the action $a_t^i = 0$, i.e., the EV chooses not to charge, is:

$$R_0(s_t^i, s_{t+1}^i) = \begin{cases} \frac{\alpha_{ods} \cdot \tilde{\phi}_p}{\Delta t_{wait}^l / t_{PET} + 1} & SOC_t^i > \varphi \\ -\alpha_{idle} & SOC_t^i \leq \varphi \end{cases} \quad (6)$$

where α_{ods} and α_{idle} are hyperparameters, $\tilde{\phi}_p$ means that when $\phi_p \leq \xi$, $\tilde{\phi}_p$ is equal to ϕ_p , and when $\phi_p > \xi$, $\tilde{\phi}_p$ is equal to ξ , ξ is the threshold for EV supply saturation, and when the EV supply rate exceeds this value, it means that the number of idle vehicles can fully meet the demand for pending orders. t_{PET} denotes the longest waiting time for an order; when the EV receives the l th order, Δt_{wait}^l denotes the actual waiting time for the order; when the EV involved in the pickup of the order, it does not receive the order $\Delta t_{wait}^l = t_{PET}$.

- When the EV has sufficient battery and the supply rate is low ($SOC_t^i > \varphi, \phi_p \leq \xi$), the agent receives a positive reward ($\alpha_{ods} \cdot \phi_p / (\Delta t_{wait}^l / t_{PET} + 1)$). This reward is linearly related to the supply rate for the decision not to charge, and it also has a negative correlation with the passenger's waiting time. Both these aspects motivate more EVs to accept orders and actively respond to passenger requests, thereby reducing passenger waiting time.
- When the EV has sufficient battery and the supply rate is adequate ($SOC_t^i > \varphi, \phi_p > \xi$), the agent receives a fixed positive reward ($\alpha_{ods} \cdot \xi / (\Delta t_{wait}^l / t_{PET} + 1)$) for deciding not to charge, providing a reward to EVs that accept orders and ensuring a certain scale of EVs to maintain the order-taking rate.
- When the EV has insufficient battery ($SOC_t^i \leq \varphi$), it is penalized severely ($-\alpha_{idle}$) for deciding not to charge, as it lacks the ability to accept orders.

The reward for the charging action ($a_t^i = \chi_t^i$) is:

$$R_{\chi_t^i}(s_t^i, s_{t+1}^i) = \begin{cases} -\beta_{ods} \cdot \Delta t_C^{i,j,k} & \phi_p \leq \xi \\ \beta_{ch} / (\Delta t_C^{i,j,k} + 1) & \phi_p > \xi \end{cases} \quad (7)$$

where β_{ods} and β_{ch} are hyperparameters, and $\Delta t_C^{i,j,k}$ denotes the charging time cost for the i th EV to charge at the k th charging area of the j th charging station.

- When the supply rate is low ($\phi_p \leq \xi$), indicating a higher demand for EVs to serve passengers, the agent's decision to charge at this time is penalized with a cost ($-\beta_{ods} \cdot \Delta t_C^{i,j,k}$) related to the charging time.
- Otherwise, when the supply rate is high ($\phi_p > \xi$), the system has sufficient available EVs to serve the passengers, so EVs can afford to spend more time charging without impacting the passenger service quality. An agent that decides to charge receives a positive reward ($\beta_{ch} / (\Delta t_C^{i,j,k} + 1)$), which is inversely proportional to the charging time cost, thus encouraging the agents to choose the appropriate charging station

to replenish their power. By incorporating the charging time cost ($\Delta t_C^{i,j,k}$) into the reward function, the agent is incentivized to choose the optimal charging station that minimizes the waiting time and charging time for the EV. This, in turn, can help reduce the overall charging time cost of the EVs in a round, leading to better efficiency and service quality for the ride-hailing platform.

In addition, the hyperparameters ($\beta_{ods}, \alpha_{idle}, \beta_{ch}, \alpha_{ods}$) should satisfy the following relationships:

- $-\beta_{ods} \cdot \Delta t_C^{i,j,k} > -\alpha_{idle}, \forall i, j, k$ constant holds. In situations where there are many pending orders and the supply rate is low ($\phi_p \leq \xi$), an EV with low SOC ($SOC_t^i \leq \xi$) may decide whether to charge or not based on the comparison between the penalties of remaining idle ($-\alpha_{idle}$) and choosing a charging station ($-\beta_{ods} \cdot \Delta t_C^{i,j,k}$). However, if the penalty for choosing any charging station is lower than the penalty of remaining idle ($-\beta_{ods} \cdot \Delta t_C^{i,j,k} \leq -\alpha_{idle}$), the EV may choose to continuously remain idle without charging. To prevent EVs from entering a cycle of continuous non-charging, the penalty for choosing any charging station must be greater than the penalty of remaining idle.
- $\beta_{ch} / (\Delta t_C^{i,j,k} + 1) > \alpha_{ods} \cdot \xi / (\Delta t_{wait}^l / t_{PET} + 1), \exists i, j, k$ holds. When the EV supply rate is high, it means that there are more idle EVs available than the pending orders, and thus, the system can afford to have some EVs charging to increase their order-taking potential without affecting passenger service quality. In this case, the agent can be motivated to choose the appropriate charging area and store power during idle time so that the EV can be ready to accept more orders in the future.

(4) Solution based on Deep Q-network: $Q^*(s_t, a_t)$ reflects the maximum value of taking action a_t in state s_t . Based on the optimal action value function $Q^*(s_t^i, a_t^i)$, the optimal action for each EV in a certain state can be determined. However, the charging scheduling management for ride-hailing services are highly complex, making it difficult to construct a mathematical model for state transition and policy. As a result, $Q^*(s_t^i, a_t^i)$ cannot be directly obtained. Therefore, a DQN is used to approximate the optimal action value function, which is denoted as $Q^*(s_t^i, a_t^i; \theta_n)$. Once the neural network is initialized and undergoes continuous training, it gradually approaches $Q^*(s_t, a_t)$. Eventually, the agent makes optimal decisions based on $Q^*(s_t^i, a_t^i; \theta_n)$.

The DQN consists of an input layer, hidden layers, and an output layer. Each layer in the neural network is a fully connected layer, and a linear transformation of the previous layer is applied to the activation function to achieve nonlinear mapping. The training of the neural network involves a process of backward propagation where we seek to minimize a predefined loss function. The definition of the loss function is as follows:

$$L_n(\theta_n) = \mathbb{E} \left[\frac{1}{2} (y_t^i - Q(s_t^i, a_t^i; \theta_n))^2 \right] \quad (8)$$

$$y_t^i = r_t^i + \gamma \cdot \max_{a^i \in A} Q(s_{t+1}^i, a^i; \theta_n^-) \quad (9)$$

We can obtain the gradient of the loss function concerning the neural network parameters using the chain rule of derivatives as follows:

$$\nabla_{\theta} L_n(\theta_n) = \mathbb{E} \left[(Q(s_t^i, a_t^i; \theta_n) - y_t) \nabla_{\theta} Q(s_t^i, a_t^i; \theta_n) \right] \quad (10)$$

The Adaptive Moment Estimation gradient descent method (Adam) [37] is utilized to train the neural network, thereby facilitating the update of θ_n . The formula is as follows:

$$m_n = \beta_1 m_{n-1} + (1 - \beta_1) \cdot \nabla_{\theta} L_n(\theta_n) \quad (11)$$

$$v_n = \beta_2 v_{n-1} + (1 - \beta_2) \cdot \nabla_{\theta} L_n(\theta_n)^2 \quad (12)$$

$$\theta_{n+1} = \theta_n - \eta \cdot \frac{m_n / (1 - \beta_1^n)}{\sqrt{v_n / (1 - \beta_2^n)} + \varepsilon} \quad (13)$$

m_n represents the first-order moments of the gradient at the n th iteration. v_n represents the second-order moments of the gradient at the n th iteration. In particular, m_0 and v_0 are both equal to 0. η refers to the learning rate, which determines the size of the step for each parameter update.

Research has shown that when using neural networks to directly approximate action value functions, the learning process can be unstable or even divergent, resulting in suboptimal performance [38]. To enhance the performance of deep reinforcement learning and improve the stability and efficiency of the convergence process, the following methods were employed:

- Experience Replay Mechanism [39]: A memory buffer of a certain capacity is constructed to store historical experiences. Once the storage capacity exceeds the minimum limit, the experience replay mechanism allows for mini-batch sampling of experience samples as training data in each training session. This mechanism not only allows for the reuse of historical experiences but also minimizes correlations among them, thereby improving the stability of the training process.
- Double DQN (DDQN) [40]: The target value y_t computed using DDQN consists of both the deep Q-network and the target network. The formula is as follows:

$$y_t^i = r_t^i + \gamma \cdot Q \left(s_{t+1}, \underset{a \in A}{\operatorname{argmax}} Q(s_{t+1}, a; \theta_n); \theta_n^- \right) \quad (14)$$

Here, θ_n denotes the parameters of the deep Q-network at the n th iteration, and θ_n^- denotes the parameters of the target network at the n th iteration. The target network is another neural network with the same structure as the deep Q-network but with different parameters for calculating the target value y_t . The former is updated during each training session using the Adam, while the latter is updated periodically according to $\theta_{n+1}^- = \lambda \theta_n + (1 - \lambda) \theta_n^-$. It avoids always selecting the maximum Q value from the target network's output, which, to a certain extent, mitigates the negative impact of overestimating the optimal action-value function and improves deep reinforcement learning performance.

- Epsilon Decaying Strategy: Initially, actions are selected randomly from the feasible action space with a high probability epsilon, promoting exploration. Over time, epsilon is gradually decayed, reducing the probability of random actions and increasing the likelihood of choosing actions with the highest expected reward, thus fostering exploitation. This strategy ensures a balance between exploring new actions and exploiting known good actions while adjusting the balance dynamically as the model learns.

4.2. Order Dispatching Model

After all idle EVs have made their charging decisions in sequence, the remaining vehicles are dispatched to serve passengers. The order dispatching problem is formulated as an optimization problem as follows:

$$\min \sum_{p=1}^{|\tilde{F}_t^*|} \sum_{q=1}^{|O_t^*|} b_{pq} \cdot \chi_{pq} \quad (15)$$

$$\text{s.t.} \sum_{p=1}^{|\tilde{F}_t^*|} \chi_{pq} \leq 1, \quad \forall q = 1, \dots, |O_t^*| \quad (16)$$

$$\sum_{q=1}^{|O_t^*|} \chi_{pq} = 1, \quad \forall p = 1, \dots, |\tilde{F}_t^*| \quad (17)$$

$$\chi_{pq} \in \{0, 1\}, \quad \forall p = 1, \dots, |\tilde{F}_t^*|, \forall q = 1, \dots, |O_t^*| \quad (18)$$

The order dispatching process aims to minimize the travel time cost of EVs for each ride match. To achieve this, the set of vehicles ready for dispatch at time t denoted by $\tilde{F}_t^*$ is identified. These are EVs with sufficient power ($SOC > \wp$) that decided not to charge. The set of ride orders that have been initiated before time t and have not yet been responded to, denoted by O_t^* , is also considered. We define the decision variables $\mathcal{X} = \{\chi_{pq} | p = 1, \dots, |\tilde{F}_t^*|; q = 1, \dots, |O_t^*|\}$, where χ_{pq} is a 0–1 variable that indicates whether the p th vehicle responds to the q th order. The cost required for the p th vehicle to respond to the q th order is denoted by b_{pq} , where $b_{pq} = d_{p,q} / \bar{v}_p$. Here, $d_{p,q}$ denotes the distance from the current location of the p th vehicle to the origin of the q th order, and $\bar{v}_p$ denotes the average travel speed of the p th vehicle.

The dispatching problem is formulated as an optimization problem, where the objective is to minimize the total travel time cost of all ride matches. Specifically, we seek to find a set of decision variables $\mathcal{X}$ that satisfy the following constraints: (1) each vehicle is assigned to at most one order, (2) each order is assigned to exactly one vehicle. The Kuhn–Munkres algorithm [36] is used to solve this optimization problem. Note that the formulation here assumes that $|\tilde{F}_t^*| \leq |O_t^*|$. A similar formulation can be made for the case $|\tilde{F}_t^*| > |O_t^*|$.

4.3. Summary of the Overall Algorithm

The complete algorithm is demonstrated in Algorithm 1. Firstly, the Q-network $Q(s_t^i; \theta)$, the target network $Q(s_t^i; \theta^-)$, and the experience replay buffer D are initialized (refer to lines 1–3). During each iteration of model training, the action value Q-network interacts with the electric vehicle charging scheduling environment to obtain interaction samples (see lines 4–16). For every idle vehicle, an action is selected based on the ϵ -greedy policy according to the action value network (refer to lines 7–12). Then, the KM algorithm is employed to assign pending and unexpired orders to available vehicles (lines 13–15). The selected vehicle is dispatched to charge, serve the order or remain idle (see lines 16–17). The $(s_{t-1}^i, a_{t-1}^i, r_{t-1}^i, s_t^i)$ quadruple corresponding to the previous state is then collected (see lines 18–19).

After the interaction between the agent and the environment is completed, the model updates the parameters of the Q-network (see lines 21–25). When the capacity of the replay buffer exceeds the minimum limit, a batch of SARS quadruples is extracted from the experience replay buffer, and θ is updated through gradient descent (see lines 21–23). After a fixed number of steps, θ^- is updated based on θ and θ^- (see lines 24–25). Finally, the algorithm returns the electric vehicle charging scheduling strategy π^* as the recommended charging scheduling plan.

Algorithm 1 Complete Algorithm with Environment Interaction

Input: Capacity and power information of charging stations C , average driving speed of the ride-hailing fleet F , spatiotemporal distribution model of origin–destination of orders, hyperparameters of the reward function, number of episodes N , critical supply rate threshold ξ , and critical battery level threshold $\wp$

Output: Scheduling policy π^*

```

1: Initialize Q-network  $Q(s_t^i; \theta)$ 
2: Initialize target network  $Q(s_t^i; \theta^-)$ 
3: Initialize replay buffer  $D$ 
4: for episode  $\leftarrow 1$  to  $N$  do
5:   Initialize environment
6:   while episode termination condition = FALSE do
7:     Get idle vehicles  $F_t$ 
8:     for  $i \leftarrow 1$  to  $|F_t|$  do
9:       if random number  $p < \epsilon$  then
10:        Select a random action  $a_t^i$ 
11:       else
12:        Select the best action  $a_t^i$  according to  $Q(s_t^i; \theta)$ 
13:       Filter available vehicles  $\tilde{F}_t^* = \{ev_i | SOC > \wp \wedge action \neq charge\}$ 
14:       Update the pending and unexpired order set  $O_t^*$ 
15:       Assign orders to vehicles using KM algorithm
16:       for  $i \leftarrow 1$  to  $|F_t|$  do
17:        Execute action and calculate reward  $r_t^i$ 
18:        if  $t \neq 0$  then
19:         Store transition  $(s_{t-1}^i, a_{t-1}^i, r_{t-1}^i, s_t^i)$  in replay buffer
20:       Update state  $s_t \leftarrow s_{t+1}$ 
21:       if update condition for  $\theta$  then
22:        Sample minibatch of transitions from  $D$ 
23:        Update  $\theta$  using gradient descent
24:       if update condition for  $\theta^-$  then
25:        Update  $\theta^-$ 
26: return  $\pi^*$ 

```

5. Experiment Setup

5.1. Datasets

In this study, we used the actual road network data of Xi'an city, randomly generated initial data of ride-hailing vehicles, and specific charging station data as input parameters for the simulation system. The enclosed area formed by four points of latitude and longitude (108.86, 34.37), (108.86, 34.45), (108.96, 34.37), and (108.96, 34.45) was selected as the operational zone, which was located in the urban center of Xi'an, a vibrant area for the city's economic development. At the beginning of each simulation, the GPS coordinates [41–43] of 300 operating ride-hailing EVs within the area were randomly generated. Each EV had an average driving speed of 40 km/h and the same battery capacity of 60 KW·h. The initial remaining power was randomly generated within the range of 40% to 50% of the total battery capacity. Considering the spatial distribution characteristics of the currently established charging stations in Xi'an, we have delineated the geographic distribution of these stations in our simulation environment, as illustrated in Figure 2. The path distance between the origin and the destination was calculated using the Baidu Map API.

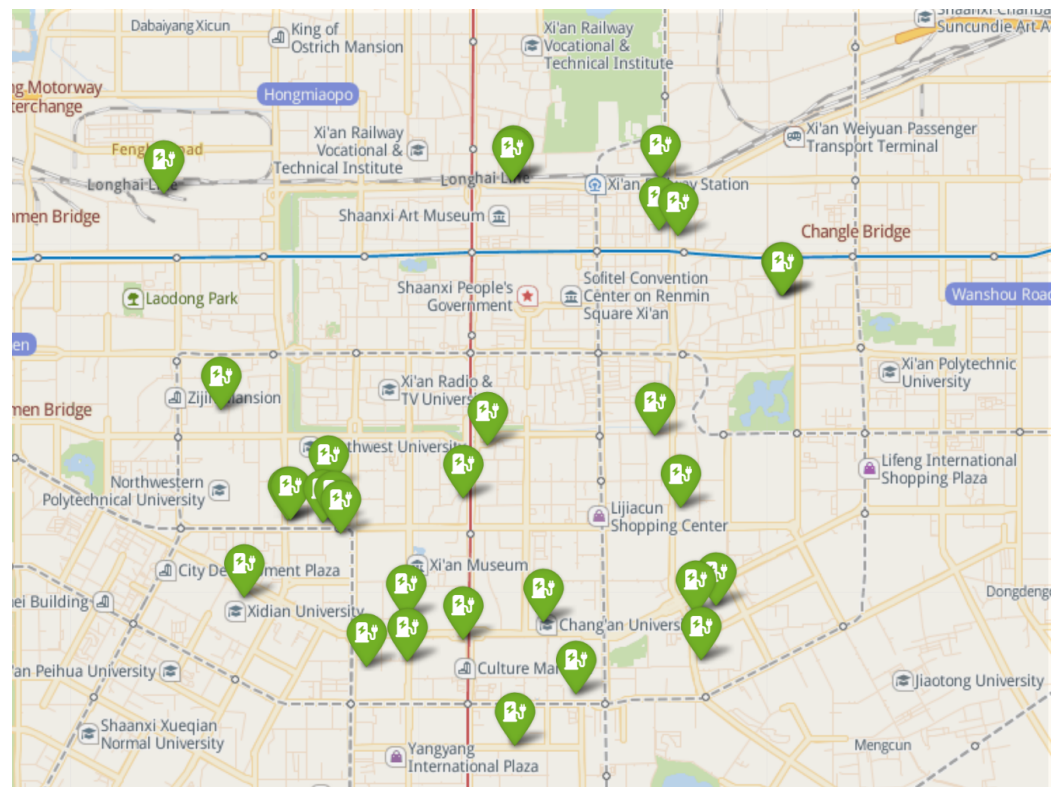


Figure 2. Geographic distribution of charging stations.

5.2. Experimental Scenario Setting

To comprehensively evaluate the performance and applicability of the model, we designed three test scenarios with different spatiotemporal distribution characteristics:

1. *Uniform.* This scenario presumes that requests for various pickup and drop-off locations (origin and destination) are randomly generated within the operational area, showcasing a uniform distribution. Temporally, the distribution of orders adheres to a Poisson process with consistent expectation values.
2. *Weekday.* In this scenario, the spatial distribution of requests for pickup and drop-off locations is identical to the *uniform* scenario, meaning that they are randomly generated within the operational area. In terms of time, the order distribution deviates as it manifests distinct peak and off-peak periods, reflecting the real-world dynamics of urban demand where certain times of day, such as rush hours, experience a surge in requests.
3. *Weekend.* This scenario encapsulates the city's leisure travel patterns during non-working days. The coordinates of the pickup and drop-off locations are determined based on Points of Interest (POIs) that have been collected via the BaiduAPI. The distribution of these POIs, encompassing a wide variety of categories, is illustrated in Figure 3. Notably, a distinct pattern of aggregation is observed at POIs associated with leisure activities, such as commercial and recreational venues, sports facilities, cultural centers, and tourist attractions. These places serve as both primary destinations and potential starting points for travel during weekends. Similar to the *weekday* scenario, the temporal distribution also showcases peak and off-peak periods.

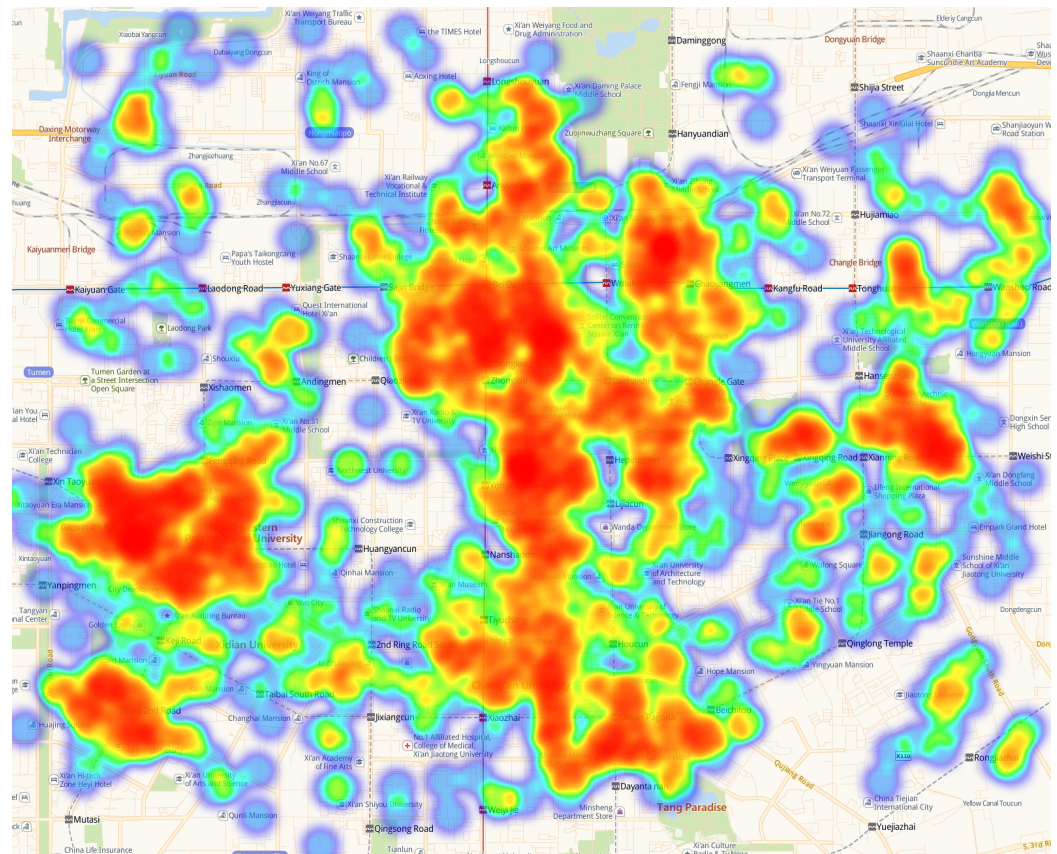


Figure 3. The spatial distribution of all POI categories.

Upon starting the simulation, order requests were randomly generated according to a specific scenario setting. To generate more realistic simulation data, the number of order requests in different time periods of *weekday* and *weekend* scenarios was determined based on the distribution of order periods in the Xi'an city taxi operation development report [44]. To illustrate these data more clearly, Figure 4 depicts the proportion of total daily orders that occur at different times of the day. The distribution of orders demonstrates a clear peak-and-valley pattern. The period exhibiting the lowest quantity of orders aligns with the interval around 3 a.m., constituting a mere 0.23% of the total daily requests. Conversely, the period marked by the highest volume of orders aligns with the interval around 8 a.m., accounting for a significant 6.51% of the total daily orders. The system screened available vehicles for charging every 2 min and then assigned orders to vehicles that have not been charged to satisfy passenger requests.

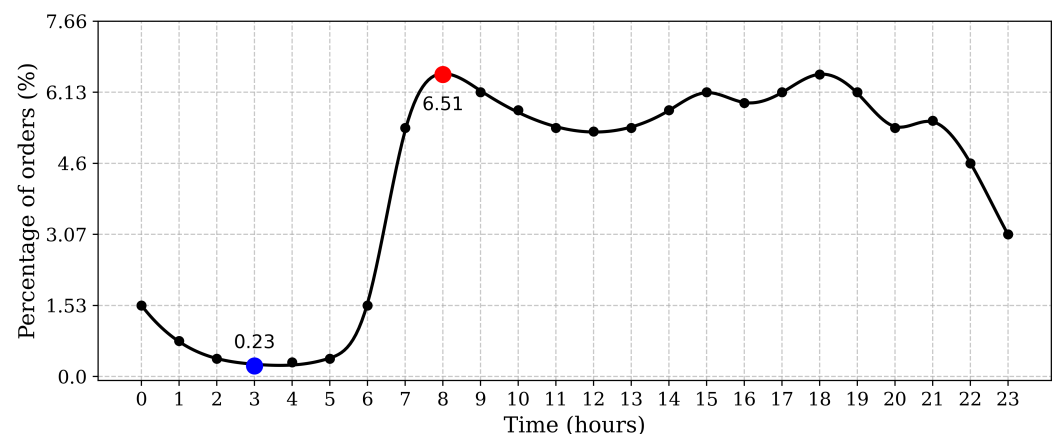


Figure 4. Temporal distribution of orders throughout the day.

5.3. Parameter Settings

The parameters and training process of reinforcement learning were set as follows: the Q-network consisted of an input layer, two hidden layers, and an output layer with dimensions 30, 256, 256, and 26, respectively. Given these settings, the parameter size of the model was 80,410. Leaky ReLU activation functions with a slope of 0.01 were used, except for the input layer. To prevent overfitting, a 0.2 dropout rate was implemented in the hidden layers [45]. The experience pool had a capacity of 50,000 tuples with training starting after collecting 10,000 tuples. An episode corresponded to the process of an EV's scheduling within a 24 h period, consisting of 720 time slices. The discount factor γ was set to 0.99, and the hyperparameters in the reward function were set as follows: $\alpha_{idle} = 10$, $\alpha_{ods} = 1.5$, $\beta_{ch} = 3.5$, $\beta_{ods} = 1$, $\zeta = 2$, $\wp = 0.2$. During training, the mini-batch size for sampled tuples was set to 100, and the Adam optimizer updated the Q-network parameters with a learning rate r of 5×10^{-4} .

5.4. Baselines

In this study, we utilized the following two strategies as baselines.

Baseline1: Ride-hailing Dispatch with Greedy and Linear Programming (RDGLP). Assign EV by solving a linear programming problem where the weights correspond to the distances between vehicles and pick-up locations. The Kuhn–Munkres algorithm is employed for problem solving. The model is mathematically formulated as follows:

$$\min \sum_{p=1}^{|\tilde{F}_t^*|} \sum_{q=1}^{|\mathcal{O}_t^*|} d_{pq} \cdot \chi_{pq} \quad (19)$$

$$\text{s.t.} \sum_{p=1}^{|\tilde{F}_t^*|} \chi_{pq} = 1, \quad \forall q = 1, \dots, \min(|\mathcal{O}_t^*|, |\tilde{F}_t^*|) \quad (20)$$

$$\sum_{q=1}^{|\mathcal{O}_t^*|} \chi_{pq} \leq 1, \quad \forall p = 1, \dots, |\tilde{F}_t^*| \quad (21)$$

$$\chi_{pq} \in \{0, 1\} \quad (22)$$

Here, d_{pq} represents the distance from vehicle p to the departure point of passenger q . The constraints in Equation (20) ensure that each order is responded to by a vehicle when the number of orders is less than or equal to the number of ride-hailing vehicles. When the number of orders exceeds the number of ride-hailing vehicles, the $|\tilde{F}_t^*|$ orders with the longest wait times are selected to ensure they are all responded to by ride-hailing vehicles. An EV whose SOC falls below a threshold $\wp$ is dispatched to the nearest charging station with a preference for high-power charging areas.

Baseline2: Reinforcement Learning for Operational Actions in Ride-hailing (RLOAR). The baseline method adopts a ride-hailing dispatch strategy proposed by Shi [23]. This strategy, called RLOAR, allows idle vehicles to choose between serving orders, charging, or remaining idle. If the charging action is chosen, the vehicle selects the nearest charging station. Tunable parameters are set for these actions to optimize the dispatch and charging decisions. To find the maximum action value function, a linear assignment problem is formulated and solved using the KM algorithm. To approximate the optimal state value function, a neural network is used. The state value function, along with the reward function, forms the optimal action value function. The parameters of the neural network are updated through a centralized learning method. More specifically, the state value function network comprises an input layer, two hidden layers, and an output layer, with respective dimensions of 5, 200, 200, and 1. The parameter size of the model is 41,601. The model employs the Adam algorithm for gradient descent and the ReLU for activation.

6. Experimental Results

This section presents experimental results comparing our model with two baselines in three different scenarios.

6.1. Uniform Scenario

Figure 5 displays the performance of PROLIFIC and two baseline algorithms. Although the initial PROLIFIC scheduling algorithm's performance was inferior to the baseline algorithms in all metrics, its performance improved as the training process proceeded. After multiple training sessions, the algorithm's performance in all metrics caught up with or surpassed the two baselines. Upon performance stabilization, the optimal charging scheduling strategy can be obtained by stopping the training. The initial fluctuations in the PROLIFIC method are due to the model's broad decision-making scope and its Epsilon-Decaying strategy. In the early stages, the model undergoes trial-and-error to adapt to the environment and makes random actions to gather information, leading to suboptimal decisions. As the model learns, it identifies more optimal strategies, reducing its reliance on random actions, which results in improved and stabilized performance.

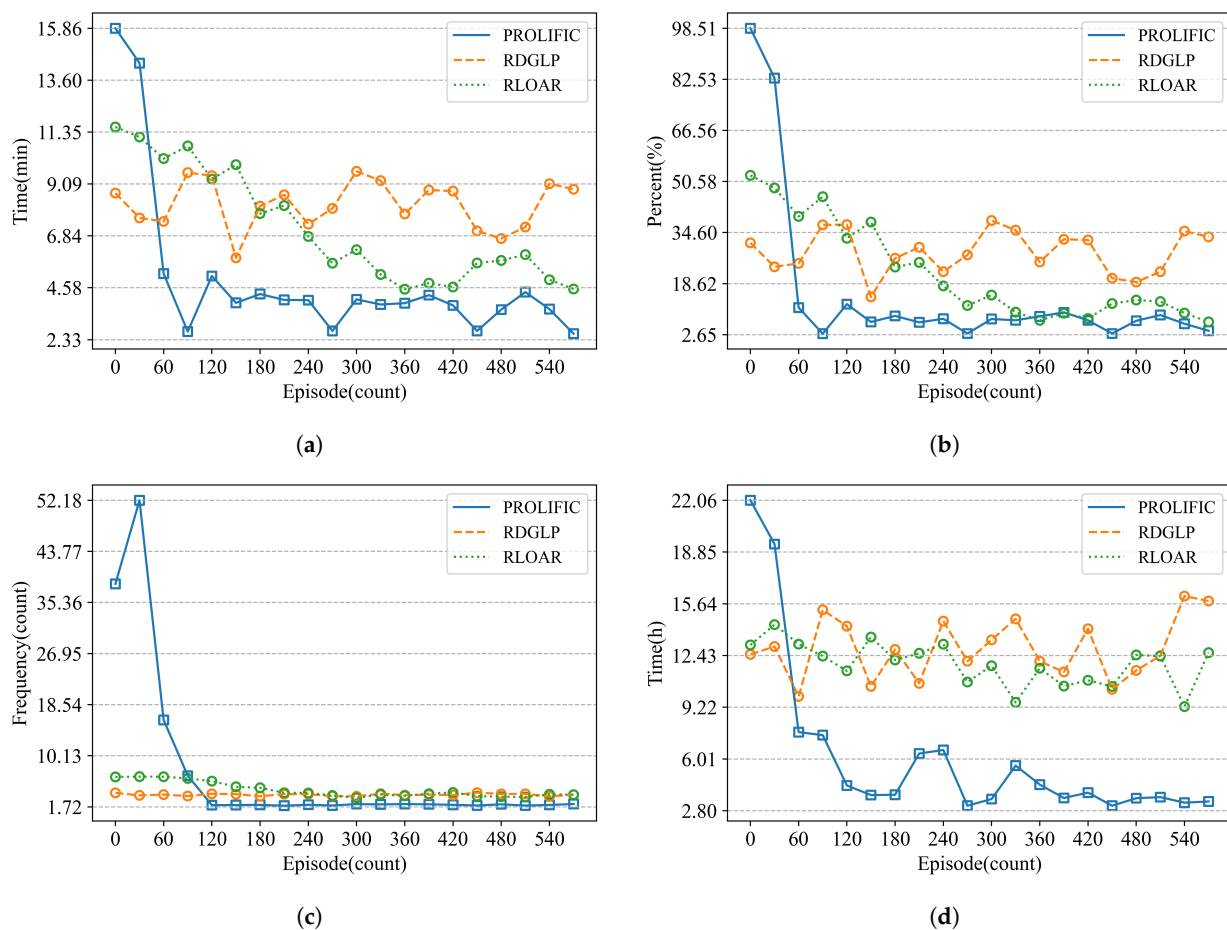


Figure 5. The results in *uniform* scenario. (a) Passenger waiting time; (b) Rate of cancelled orders; (c) Charging frequency; (d) Total charging time.

Figure 5a,b show the performance of each model in responding to passengers, where Figure 5a represents the average waiting time per passenger in an episode, and Figure 5b represents the proportion of passengers who were not responded to within 15 min in an episode relative to all passengers. PROLIFIC's performance was slightly better than that of RLOAR, while it demonstrated a substantial advantage in comparison to RDGLP.

The results for charging scheduling are presented in Figure 5c,d. Figure 5c shows the average charging frequency per EV in an episode, while Figure 5d shows the average total

charging time per EV in an episode. PROLIFIC's charging frequency was comparable to the two baselines, but its total charging time was significantly lower. This was attributed to PROLIFIC's ability to select suitable charging stations, avoiding overcrowding and the use of slow charging piles. In this scenario, the two baselines had similar performance in charging scheduling in terms of all charging dispatch metrics due to the long travel distances and high demand of orders per hour, resulting in shorter idle times for EVs. PROLIFIC's capability to choose appropriate charging stations was validated in this scenario.

6.2. Weekday Scenario

Figure 6 shows the results for the *weekday* scenario. From Figure 6a,b, it can be seen that PROLIFIC significantly outperformed RLOAR in terms of the rate of cancelled orders, and RLOAR outperformed RDGLP.

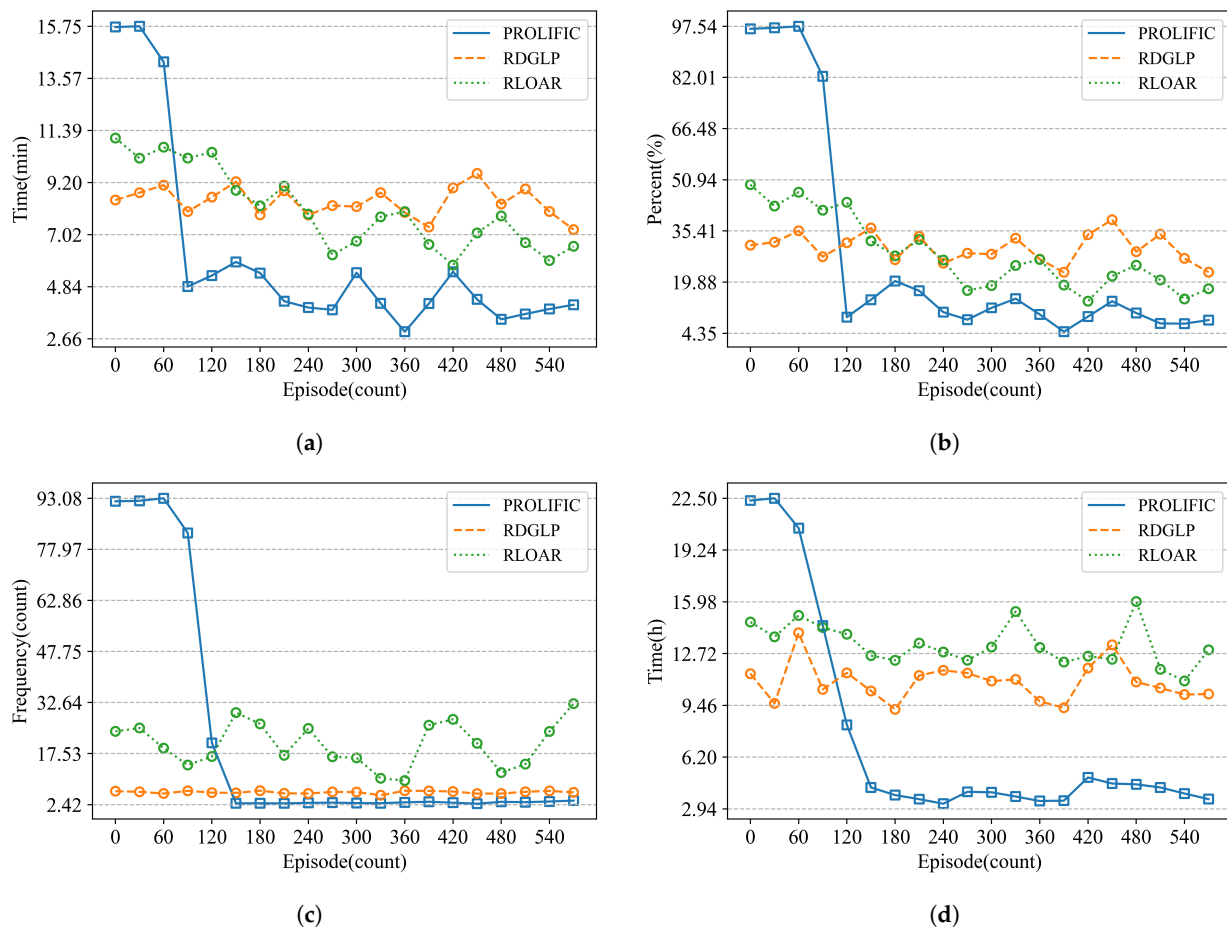


Figure 6. The results in *weekday* scenario. (a) Passenger waiting time; (b) Rate of cancelled orders; (c) Charging frequency; (d) Total charging time.

From Figure 6c,d, it was evident that both PROLIFIC and RDGLP exhibited lower charging frequencies compared to RLOAR. Additionally, PROLIFIC demonstrated significantly lower charging costs than the two baseline methods. While both RLOAR and RDGLP followed the principle of charging at the nearest station, the total charging time cost of RLOAR was higher due to its higher charging frequency per episode. RLOAR set the reward value for the pass action to 0 and assigned a positive reward value for the charging action when an EV was near a charging station. This reward configuration often led to a higher frequency of charging actions by EVs during low-demand periods. On the other hand, PROLIFIC's reward function for charging actions was based on the total charging time, making the rewards more sensitive to the charging duration. During low-demand periods, the rewards for charging and not charging were both positive, with the reward

for charging possibly being greater or less than the reward for not charging, depending on the specific situation. As a result, a long charging time would lead to a lower reward value (see Equation (7)), potentially discouraging EVs from choosing to charge during low-demand periods.

The results in the *weekday* scenario validate that RLOAR might not effectively adapt its strategy during low-demand periods, while PROLIFIC demonstrated the ability to identify order trends and select appropriate timing and charging stations for charging. PROLIFIC's approach allows for more dynamic charging decisions based on the actual charging time cost, which contributes to its superior performance in this scenario compared to the baseline methods.

6.3. Weekend Scenario

The results in the *weekend* scenario (as shown in the Figure 7) demonstrated that PROLIFIC effectively handled the peak–valley characteristics of order quantity and non-uniform spatial distribution of charging stations. It outperformed the two baseline methods in terms of total charging time and passenger waiting time, indicating the effectiveness of the algorithm in selecting appropriate charging stations and scheduling charging times. The results also highlighted the limitations of simply choosing the nearest charging station, which could lead to congestion and increased charging times, even when RLOAR uses reinforcement learning methods to schedule the actions of each vehicle.

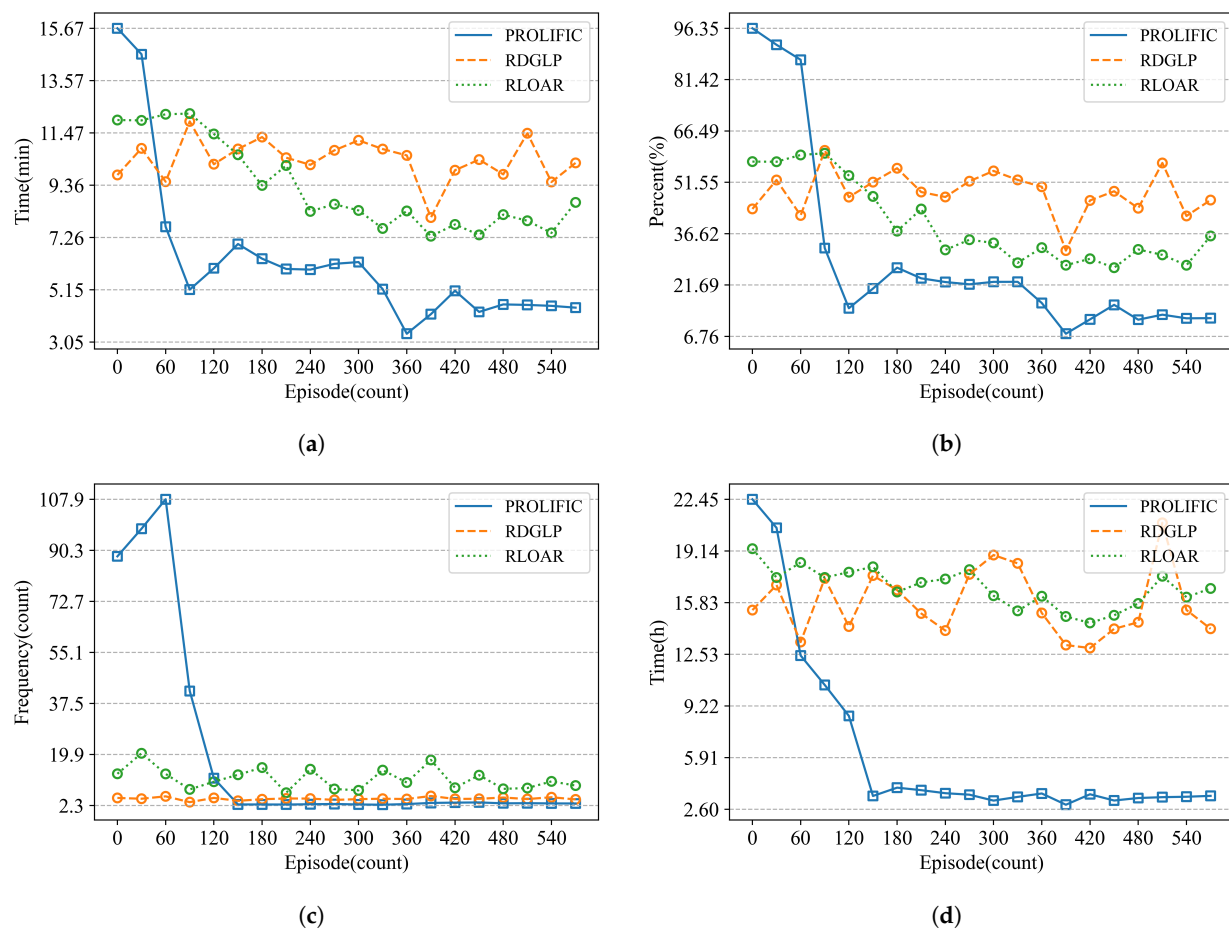


Figure 7. The results in *weekend* scenario. (a) Passenger waiting time; (b) Rate of cancelled orders; (c) Charging frequency; (d) Total charging time.

6.4. Discussion

Table 2 presents the performance of PROLIFIC and the baselines across different scenarios. In addition to the metrics used in the previous subsection, we also divide the

total charging cost into more delicate parts, including travel time cost, charging waiting time cost, and pure charging time cost.

Table 2. Comparison of EV fleet scheduling and charging performance for PROLIFIC, RDGLP, and RLOAR under different conditions.

Category	Uniform			Weekday			Weekend		
	PROLIFIC	RDGLP	RLOAR	PROLIFIC	RDGLP	RLOAR	PROLIFIC	RDGLP	RLOAR
Passenger Waiting Time (min)	3.47	7.96	5.26	4.16	8.51	6.63	4.58	10.23	7.90
Rates of Cancelled Orders (%)	5.97	27.06	10.43	9.51	31.11	18.99	12.71	47.42	30.31
Charging Frequency (count)	2.05	3.77	3.64	3.17	6.08	21.82	3.13	4.71	9.55
Travel Time Cost (h)	0.46	0.22	0.22	0.66	0.27	0.28	0.70	0.21	0.23
Charging Waiting Time Cost (h)	0.68	10.63	7.95	1.43	8.45	9.38	0.87	12.95	12.66
Pure Charging Time Cost (h)	2.35	2.53	3.19	2.22	2.43	3.11	1.80	2.19	3.07
Total Charging Cost (h)	3.49	13.38	11.36	4.29	11.15	12.77	3.38	15.35	15.96

The heat map in Figure 8 summarizes the comparison between PROLIFIC and the baseline methods. Each node in the heat map shows the change rate of PROLIFIC relative to the baseline for a given metric. As shown in Figure 8, PROLIFIC achieved superior performance compared to the two baseline algorithms in each metric except for the travel time cost. It is important to note that baselines always select the nearest charging station when deciding to charge. However, the PROLIFIC algorithm may opt for a charging station that is farther away, but boasts a higher power outlet or shorter queue waiting time, with the goal of reducing the overall cost of charging time. In contrast, the RDGLP algorithm only charges when the SOC is less than φ . This lower frequency of charging results in a reduced total charging travel time within an episode, which may explain why RDGLP outperforms other algorithms in terms of travel time cost. Meanwhile, the RLOAR algorithm often triggers charging behavior when the ride-hailing vehicle is close to a charging station. Although the frequency of charging is high, the distance traveled for each charge is relatively short. Consequently, the travel time cost for RLOAR is also relatively low.

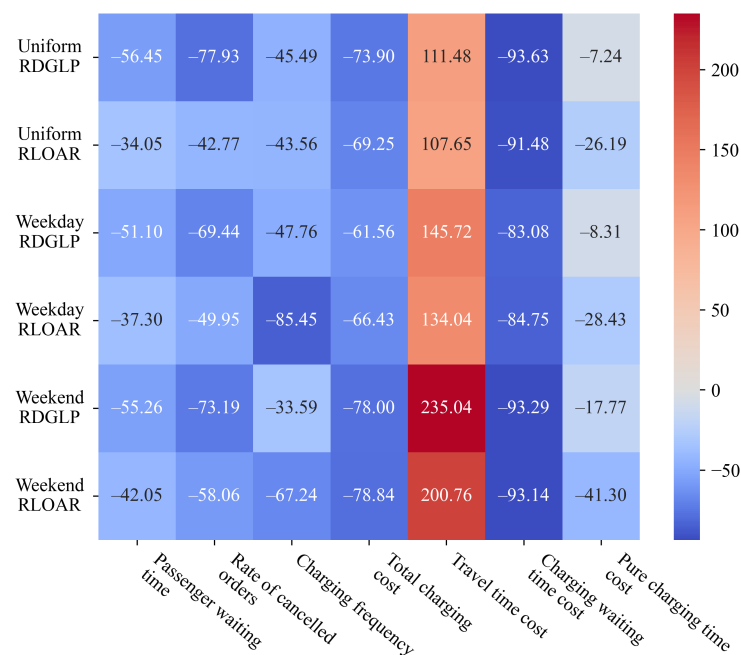


Figure 8. Heat map of PROLIFIC performance optimisation ratios.

Compared to other algorithms, PROLIFIC can still ensure low passenger wait times and charging time costs even in more complex scenarios such as *weekday* and *weekend* scenarios. This demonstrated the effectiveness of PROLIFIC in selecting appropriate charging stations and scheduling charging times to handle the non-uniform spatial distribution of charging stations and the peak–valley characteristics of order quantity.

In these three scenarios, PROLIFIC used the same set of parameters, indicating that the algorithm has a certain generalization ability and can learn deeper semantic information through continuous interaction with the environment, thereby reducing the time cost of charging and passengers' waiting time. The simulation process included randomly generated orders and fleet initialization each time rather than a fixed random seed, reflecting the algorithm's robustness.

7. Conclusions

In this study, we propose PROLIFIC, a novel deep reinforcement learning-based strategy for EVs scheduling and charging. By fostering the use of EVs, PROLIFIC contributes to sustainable urban mobility and a reduced carbon footprint, ultimately preserving the environment for future generations.

The primary innovation of PROLIFIC lies in its fusion of centralized order allocation with distributed charging decisions, which are facilitated through distributed deep Q-networks. Contrasting prior work, which often separates charging from dispatching decisions or defaults to the nearest charging station, PROLIFIC enables each EV to make autonomous decisions based on its unique state and environment. This methodology recognizes the intricate interdependencies between charging and dispatching decisions, effectively addressing the 'curse of dimensionality' encountered in deep learning, and produces optimal solutions that minimize both passenger and charging wait times. Consequently, PROLIFIC provides an advanced optimization strategy for networked car services, proficiently balancing these decisions within the realm of EV ride-hailing services.

Furthermore, we have designed a unique reward function for PROLIFIC that not only quantifies the contribution of actions toward the overarching objective but also guides the agent toward optimal behaviors through a progressive reinforcement mechanism. By considering factors such as the EV's remaining battery power, passenger demand, and the charging station's occupancy, our reward function aids the EV in striking an optimal balance between order acceptance and charging. This approach also contributes to the stability of the training process. The progressive reinforcement mechanism accelerates the agent's discovery of efficient strategies, mitigating inherent uncertainty and volatility during training. Simultaneously, by reflecting the complexity of real-world scenarios, this reward function enhances the adaptability of the trained strategies in uncertain and dynamic environments.

To evaluate PROLIFIC's performance, we created a simulation system that emulates real-world conditions, leveraging actual spatial distribution data for orders, charging stations, and other relevant road network information. The system considers the diversity of charging stations, wait times, and EV accessibility. We compared PROLIFIC's performance with two baseline algorithms across varied scenarios, including *weekend*, *weekday*, and *uniform* conditions. PROLIFIC consistently outperformed the baseline algorithms, significantly reducing charging wait times by 61.56% to 78.84% and passenger wait times by 34.05% to 56.45%. These results highlight PROLIFIC's capability to identify patterns in order distribution and charging station availability, which assists in making informed decisions about optimal charging times and locations, thereby improving the overall service efficiency. Despite a slightly higher travel time cost compared to other algorithms, PROLIFIC exhibits notable advantages in reducing the charging wait time cost and total charging cost, emphasizing its effectiveness in minimizing overall charging expenses and passenger wait time. Additionally, PROLIFIC sustains low passenger wait times and charging costs, even in complex scenarios like *weekday* and *weekend*. This adaptability, which responds to

changes in charging station distribution and order volume, further confirms PROLIFIC's competence in managing these intricate situations.

For future work, we aim to employ multi-intelligence deep reinforcement learning to amalgamate charging, order allocation, and repositioning into a comprehensive scheduling approach to enhance overall fleet coordination and further elevate operational efficacy. We are also planning to incorporate the capacity of the power grid and energy pricing considerations while taking into account the variations in usage patterns between day and night.

Author Contributions: Conceptualization, J.M. and Y.Z.; methodology, Y.Z.; software, Y.Z.; validation, J.M., Y.Z. and Z.D.; formal analysis, Y.Z.; investigation, Y.Z.; resources, J.M. and Z.D.; data curation, Y.Z.; writing—original draft preparation, Y.Z.; writing—review and editing, J.M.; visualization, Y.Z.; supervision, L.T.; project administration, L.T.; funding acquisition, J.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Natural Science Foundation of China (No.62002030).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kumar, R.R.; Alok, K. Adoption of electric vehicle: A literature review and prospects for sustainability. *J. Clean. Prod.* **2020**, *253*, 119911. [\[CrossRef\]](#)
2. Rasbash, D.; Dillman, K.J.; Heinonen, J.; Ásgeirsson, E.I. A National and Regional Greenhouse Gas Breakeven Assessment of EVs across North America. *Sustainability* **2023**, *15*, 2181. [\[CrossRef\]](#)
3. Jenn, A. Emissions benefits of electric vehicles in Uber and Lyft ride-hailing services. *Nat. Energy* **2020**, *5*, 520–525. [\[CrossRef\]](#)
4. Xia, X.; Meng, Z.; Han, X.; Li, H.; Tsukiji, T.; Xu, R.; Zheng, Z.; Ma, J. An automated driving systems data acquisition and analytics platform. *Transp. Res. Part C Emerg. Technol.* **2023**, *151*, 104120. [\[CrossRef\]](#)
5. Xia, X.; Bhatt, N.P.; Khajepour, A.; Hashemi, E. Integrated Inertial-LiDAR-Based Map Matching Localization for Varying Environments. *IEEE Trans. Intell. Veh.* **2023**, 1–12. [\[CrossRef\]](#)
6. Meng, Z.; Xia, X.; Xu, R.; Liu, W.; Ma, J. HYDRO-3D: Hybrid Object Detection and Tracking for Cooperative Perception Using 3D LiDAR. *IEEE Trans. Intell. Veh.* **2023**, 1–13. [\[CrossRef\]](#)
7. Liu, W.; Quijano, K.; Crawford, M.M. YOLOv5-Tassel: Detecting Tassels in RGB UAV Imagery With Improved YOLOv5 Based on Transfer Learning. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 8085–8094. [\[CrossRef\]](#)
8. Pal, A.; Bhattacharya, A.; Chakraborty, A.K. Allocation of electric vehicle charging station considering uncertainties. *Sustain. Energy Grids Netw.* **2021**, *25*, 100422. [\[CrossRef\]](#)
9. Zhang, C.; Liu, Y.; Wu, F.; Tang, B.; Fan, W. Effective Charging Planning Based on Deep Reinforcement Learning for Electric Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 542–554. [\[CrossRef\]](#)
10. Li, C.; Dong, Z.; Fisher, N.; Zhu, D. Coupling User Preference with External Rewards to Enable Driver-centered and Resource-aware EV Charging Recommendation. In Proceedings of the Machine Learning and Knowledge Discovery in Databases, Grenoble, France, 19–23 September 2022; Springer Nature Switzerland: Cham, Switzerland, 2023; pp. 3–19. [\[CrossRef\]](#)
11. Ma, T.Y.; Xie, S. Optimal fast charging station locations for electric ridesharing with vehicle-charging station assignment. *Transp. Res. Part D Transp. Environ.* **2021**, *90*, 102682. [\[CrossRef\]](#)
12. Zhang, L.; Gong, K.; Xu, M. Congestion Control in Charging Stations Allocation with Q-Learning. *Sustainability* **2019**, *11*, 3900. [\[CrossRef\]](#)
13. Suanpang, P.; Jamjuntr, P.; Kaewyong, P.; Niamsorn, C.; Jermisittiparsert, K. An Intelligent Recommendation for Intelligently Accessible Charging Stations: Electronic Vehicle Charging to Support a Sustainable Smart Tourism City. *Sustainability* **2023**, *15*, 455. [\[CrossRef\]](#)
14. Cao, Y.; Wang, H.; Li, D.; Zhang, G. Smart Online Charging Algorithm for Electric Vehicles via Customized Actor–Critic Learning. *IEEE Internet Things J.* **2022**, *9*, 684–694. [\[CrossRef\]](#)
15. Zhu, M.; Liu, X.Y.; Wang, X. Joint Transportation and Charging Scheduling in Public Vehicle Systems—A Game Theoretic Approach. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 2407–2419. [\[CrossRef\]](#)
16. Cao, Y.; Wang, Y. Smart Carbon Emission Scheduling for Electric Vehicles via Reinforcement Learning under Carbon Peak Target. *Sustainability* **2022**, *14*, 12608. [\[CrossRef\]](#)
17. Basso, R.; Kulcsar, B.; Egardt, B.; Lindroth, P.; Sanchez-Diaz, I. Energy consumption estimation integrated into the Electric Vehicle Routing Problem. *Transp. Res. Part D Transp. Environ.* **2019**, *69*, 141–167. [\[CrossRef\]](#)
18. Gao, Y.; Yang, J.; Yang, M.; Li, Z. Deep Reinforcement Learning Based Optimal Schedule for a Battery Swapping Station Considering Uncertainties. *IEEE Trans. Ind. Appl.* **2020**, *56*, 5775–5784. [\[CrossRef\]](#)

19. Bai, J.; Ding, T.; Jia, W.; Zhu, S.; Bai, L.; Li, F. Online Rectangle Packing Algorithm for Swapped Battery Charging Dispatch Model Considering Continuous Charging Power. *IEEE Trans. Autom. Sci. Eng.* **2022**. [CrossRef]
20. Lin, B.; Ghaddar, B.; Nathwani, J. Deep Reinforcement Learning for the Electric Vehicle Routing Problem With Time Windows. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 11528–11538. [CrossRef]
21. Froger, A.; Mendoza, J.E.; Jabali, O.; Laporte, G. Improved formulations and algorithmic components for the electric vehicle routing problem with nonlinear charging functions. *Comput. Oper. Res.* **2019**, *104*, 256–294. [CrossRef]
22. Baum, M.; Dibbelt, J.; Gamsa, A.; Wagner, D.; Zuendorf, T. Shortest Feasible Paths with Charging Stops or Battery Electric Vehicles. *Transp. Sci.* **2019**, *53*, 1627–1655. [CrossRef]
23. Shi, J.; Gao, Y.; Wang, W.; Yu, N.; Ioannou, P.A. Operating Electric Vehicle Fleet for Ride-Hailing Services With Reinforcement Learning. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 4822–4834. [CrossRef]
24. Kullman, N.D.; Cousineau, M.; Goodson, J.C.; Mendoza, J.E. Dynamic Ride-Hailing with Electric Vehicles. *Transp. Sci.* **2022**, *56*, 775–794. [CrossRef]
25. Tang, X.; Li, M.; Lin, X.; He, F. Online operations of automated electric taxi fleets: An advisor-student reinforcement learning framework. *Transp. Res. Part C Emerg. Technol.* **2020**, *121*, 102844. [CrossRef]
26. Liu, S.; Wang, Y.; Chen, X.; Fu, Y.; Di, X. SMART-eFlo: An Integrated SUMO-Gym Framework for Multi-Agent Reinforcement Learning in Electric Fleet Management Problem. In Proceedings of the 2022 IEEE 25th International Conference on Intelligent Transportation Systems, Macau, China, 8–12 October 2022; pp. 3026–3031. [CrossRef]
27. Zhang, H.; Sheppard, C.J.R.; Lipman, T.E.; Moura, S.J. Joint Fleet Sizing and Charging System Planning for Autonomous Electric Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2022**, *21*, 4725–4738. [CrossRef]
28. Zhang, W.; Liu, H.; Xiong, H.; Xu, T.; Wang, F.; Xin, H.; Wu, H. RLCharge: Imitative Multi-Agent Spatiotemporal Reinforcement Learning for Electric Vehicle Charging Station Recommendation. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 6290–6304. [CrossRef]
29. An, Y.; Gao, Y.; Wu, N.; Zhu, J.; Li, H.; Yang, J. Optimal scheduling of electric vehicle charging operations considering real-time traffic condition and travel distance. *Expert Syst. Appl.* **2023**, *213*, 118941. [CrossRef]
30. Liu, W.L.; Gong, Y.J.; Chen, W.N.; Liu, Z.; Wang, H.; Zhang, J. Coordinated Charging Scheduling of Electric Vehicles: A Mixed-Variable Differential Evolution Approach. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 5094–5109. [CrossRef]
31. Yu, G.; Liu, A.; Zhang, J.; Sun, H. Optimal operations planning of electric autonomous vehicles via asynchronous learning in ride-hailing systems. *Omega* **2021**, *103*, 102448. [CrossRef]
32. Zalesak, M.; Samaranayake, S. Real time operation of high-capacity electric vehicle ridesharing fleets. *Transp. Res. Part C Emerg. Technol.* **2021**, *133*, 103413. [CrossRef]
33. Guo, G.; Xu, Y. A Deep Reinforcement Learning Approach to Ride-Sharing Vehicle Dispatching in Autonomous Mobility-on-Demand Systems. *IEEE Intell. Transp. Syst. Mag.* **2022**, *14*, 128–140. [CrossRef]
34. Yi, Z.; Smart, J. A framework for integrated dispatching and charging management of an autonomous electric vehicle ride-hailing fleet. *Transp. Res. Part D Transp. Environ.* **2021**, *95*, 102822. [CrossRef]
35. Yuan, Y.; Zhang, D.; Miao, F.; Chen, J.; He, T.; Lin, S. p²Charging: Proactive Partial Charging for Electric Taxi Systems. In Proceedings of the 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), Dallas, TX, USA, 7–10 July 2019; pp. 688–699. [CrossRef]
36. Bourgeois, F.; Lassalle, J.C. An Extension of the Munkres Algorithm for the Assignment Problem to Rectangular Matrices. *Commun. ACM* **1971**, *14*, 802–804. [CrossRef]
37. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2017**, arXiv:1412.6980. [CrossRef]
38. 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* **2015**, *518*, 529–533. [CrossRef] [PubMed]
39. Lin, L.J. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Mach. Learn.* **1992**, *8*, 293–321. [CrossRef]
40. Hasselt, H.v.; Guez, A.; Silver, D. Deep Reinforcement Learning with Double Q-Learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; Volume 30. [CrossRef]
41. Xia, X.; Hashemi, E.; Xiong, L.; Khajepour, A. Autonomous Vehicle Kinematics and Dynamics Synthesis for Sideslip Angle Estimation Based on Consensus Kalman Filter. *IEEE Trans. Control Syst. Technol.* **2023**, *31*, 179–192. [CrossRef]
42. Xiong, L.; Xia, X.; Lu, Y.; Liu, W.; Gao, L.; Song, S.; Yu, Z. IMU-Based Automated Vehicle Body Sideslip Angle and Attitude Estimation Aided by GNSS Using Parallel Adaptive Kalman Filters. *IEEE Trans. Veh. Technol.* **2020**, *69*, 10668–10680. [CrossRef]
43. Liu, W.; Xia, X.; Xiong, L.; Lu, Y.; Gao, L.; Yu, Z. Automated Vehicle Sideslip Angle Estimation Considering Signal Measurement Characteristic. *IEEE Sens. J.* **2021**, *21*, 21675–21687. [CrossRef]
44. 2020 Xi'an Taxi Operation Development Report. Available online: <http://jtj.xa.gov.cn/jtzc/jtkx/60306640f8fd1c2073f5fa91.html> (accessed on 2 July 2023).
45. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.