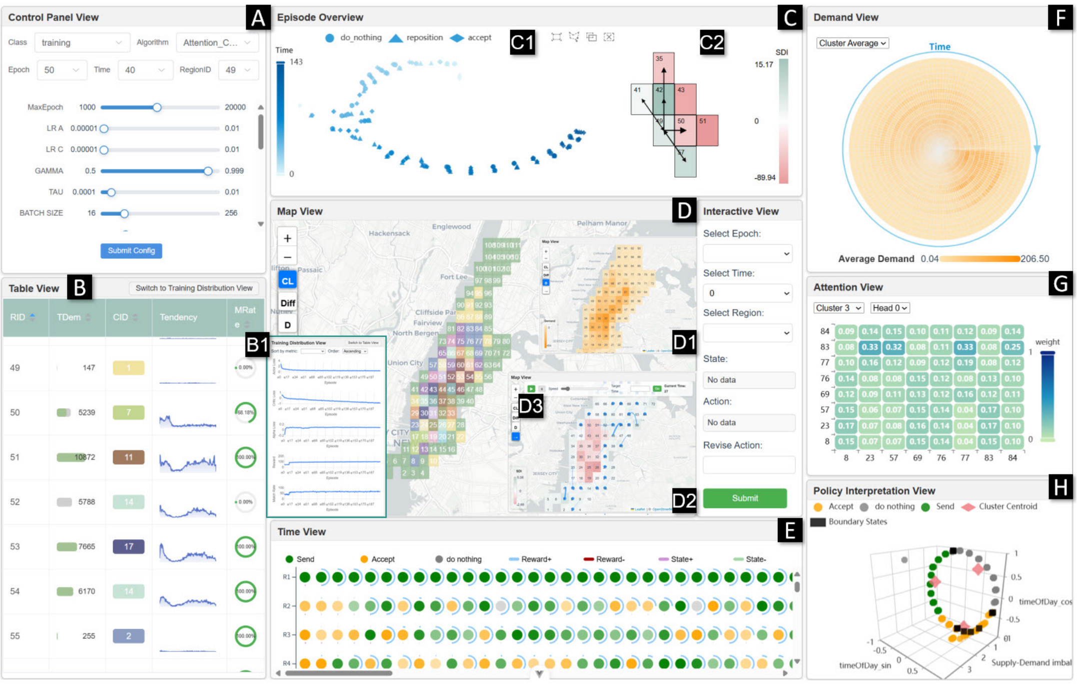
Visual Analytics for Taxi Dispatching Based on Multi-Agent Reinforcement Learning

Qiushi Xia Xinru Wang Huijie Zhang* Yiming Lin Zhaohan Lv

School of Information Science and Technology, Northeast Normal University



Class: training; Algorithm: Attention_C...; Epoch: 50; Time: 40; RegionID: 49

MaxEpoch 1000 — 20000
LR A 0.00001 — 0.01
LR C 0.00001 — 0.01
GAMMA 0.5 — 0.999
TAU 0.0001 — 0.01
BATCH SIZE 16 — 256

RID	TDem	CID	Tendency	MRate
49	147	1		0.00%
50	5239	7		68.18%
51	10872	11		100.00%
52	5788	14		0.00%
53	7665	17		100.00%
54	6170	14		100.00%
55	255	2		100.00%

	8	23	57	69	76	77	83	84
84	0.09	0.14	0.15	0.10	0.11	0.12	0.09	0.14
83	0.08	0.33	0.32	0.08	0.11	0.33	0.08	0.25
77	0.10	0.16	0.12	0.09	0.12	0.19	0.08	0.12
76	0.14	0.08	0.08	0.15	0.12	0.08	0.13	0.10
69	0.12	0.09	0.11	0.13	0.12	0.16	0.12	0.11
57	0.15	0.06	0.07	0.15	0.14	0.04	0.17	0.10
23	0.17	0.07	0.08	0.16	0.14	0.04	0.17	0.10
8	0.15	0.07	0.07	0.15	0.14	0.04	0.15	0.10

Figure 1: Overview of the DpLens visual analytics system, including (A) Control Panel for configuring training parameters, environment settings, and models; (B) Training Distribution View and Table View for monitoring performance metrics and multi-scale regional data; (C) Episode Overview combining dimensionality-reduced state layouts and Supply-Demand Imbalance Matrix for behavior analysis; (D) Map View visualizing spatial demand, supply-demand imbalance, and dispatch flows with time playback; (E) Time View displaying spatiotemporal action patterns and feedback; (F) Demand View illustrating temporal demand variations across clusters via radial bar charts; (G) Attention View revealing spatial dependency modeling through Critic attention weights; and (H) Policy Interpretation View showing 3D embeddings of state-action mappings, policy clusters, and decision boundaries.

Abstract

Taxi dispatching, as a critical task in urban transportation systems, aims to optimize order matching and reduces empty mileage by reallocating idle vehicles from surplus supply areas to high-demand regions, thereby improving system efficiency and economic benefits. In recent years, Multi-Agent Reinforcement Learning (MARL), with its ability to model regional collaborative behaviors and dynamic optimization, has become a prominent technical approach to solving this problem. However, existing methods still face challenges in practice. On one hand, the heterogeneity and non-stationarity between regions at the urban scale increase the complexity of policy learning, making it difficult to achieve efficient coordination of dispatching behaviors. On the other hand, the dispatching process highly depends on temporal context, and the lack of model interpretability restricts practical applications and policy tuning. To address these issues, we propose a region-level MARL dispatch framework, where regions are treated as agents and the action space is heterogeneous. This framework models regional state perception and cross-region vehicle migration behaviors to achieve joint optimization among multiple agents. Building on this, we design and implement a visual analytics system, *DpLens*, which combines multi-view strategy analysis and key state identification to support users in exploring the evolution of dispatch strategies and the collaboration mechanisms among agents, from both macro and micro perspectives and across multiple dimensions. Through case studies of typical urban dispatch tasks and user studies, we demonstrate the effectiveness of our approach in enhancing model interpretability, assisting strategy optimization, and improving system reliability.

Index Terms: Visual Analytics, Taxi Dispatching, Reinforcement Learning, Agent.

1 Introduction

Efficient taxi dispatching is a cornerstone of modern urban transportation systems, playing a pivotal role in balancing vehicle supply with passenger demand.

Its core objective is to strategically redeploy idle vehicles from areas of oversupply to regions with concentrated demand, thereby enhancing order matching rates, minimizing

*Corresponding author. E-mail: zhanghj167@nenu.edu.cn.

unproductive empty cruising (deadheading), and ultimately optimizing the overall operational efficiency and economic viability of the taxi fleet [1, 39]. As cities continue to expand and traffic dynamics become increasingly complex, traditional dispatching methods often struggle to adapt effectively, necessitating more intelligent and responsive solutions.

In recent years, MARL has emerged as a powerful paradigm for addressing complex decision-making problems in dynamic, multi-entity environments, making it particularly well-suited for the taxi dispatching domain [10, 36]. MARL frameworks enable the modeling of cooperative and competitive behaviors among different entities—such as individual taxis or more abstract geographic regions—and learn dispatch strategies through interaction with the environment. This capability to model regional collaborative behavior and perform dynamic optimization has shown great potential in improving dispatching performance. However, applying MARL to large-scale urban taxi dispatching in practice still faces significant challenges. Firstly, due to spatial heterogeneity in demand patterns across different regions at a given time—and the temporal dynamics of these patterns—optimal dispatch behavior between regions becomes highly unstable. This increases the difficulty of policy learning and cooperation in distributed multi-agent dispatching systems. Secondly, the dispatching process is closely intertwined with temporal context; decisions made at one point in time can have cascading effects on future system states, requiring models to capture and leverage such temporal dependencies. Compounding these issues is the ”black box” nature of many reinforcement learning models. The lack of interpretability in the learned policies restricts their practical adoption, as operators and stakeholders often require an understanding of why certain dispatch decisions are made to build trust, facilitate debugging, and enable effective policy refinement.

To address these limitations, this paper proposes a novel regional-level MARL dispatching framework, termed Clustered Attention-based Multi-Agent Soft Actor-Critic (CL-ATT-MASAC). In this approach, we employ clustering techniques to identify regions with similar demand patterns, allowing agents within the same cluster to share a policy network and thus learn more efficient and consistent cooperative strategies. Furthermore, we introduce a multi-head attention mechanism into the critic network, enabling agents to selectively focus on critical agent information when evaluating decision value, thereby achieving more effective and scalable learning in complex multi-agent environments. By modeling regions as agents with heterogeneous action spaces and incorporating the aforementioned mechanisms, the framework achieves system-level joint optimization objectives. Building upon this MARL framework and to specifically tackle the challenges of interpretability and policy optimization, we design and implement a visual analytics system named *DpLens*, which integrates multi-view policy analysis, critical state identification, and interactive exploration functionalities. The system supports users in exploring the evolution of dispatching strategies and complex inter-agent cooperation mechanisms across multiple analytical dimensions, from the macro level (city-wide) to the micro level (individual regions). Through comprehensive case studies on representative urban taxi dispatching tasks, supported by user studies, we demonstrate the effectiveness of the proposed MARL framework and the *DpLens* system in enhancing model interpretability, assisting policy refinement, and ultimately improving the reliability and trustworthiness of intelligent dispatching systems.

- We propose a novel region-level MARL taxi dispatching framework that integrates clustering and attention mechanisms. The framework enables agents in regions with similar patterns to share policies via clustering to improve learning efficiency and enhances selective information exchange among agents through attention mechanisms, thereby achieving more efficient vehicle reallocation coordination and joint optimization.

- We design and implement *DpLens*, a visual analytics system that enables users to explore, understand, and refine learned policies, thereby further enhancing the interpretability of MARL models and supporting effective policy optimization.

- Through case studies and user evaluations, we demonstrate that the synergy between our MARL framework and the *DpLens* system significantly improves the interpretability of vehicle dispatching strategies and enhances the reliability of the dispatch system in applications simulating real-world scenarios.

2 Related Work

In this section, we summarizes the research related to this study, focusing on three main areas: the development of taxi dispatching, MARL, and the associated visual analytics research.

2.1 Method of Taxi Dispatching

Taxi dispatching has long been a critical research topic in intelligent transportation systems. Early approaches predominantly relied on rule-based or heuristic strategies, employing simple principles such as nearest vehicle assignment and queuing theory. While these methods are easy to implement, they struggle to cope with the dynamic and complex nature of urban environments [11, 12]. Subsequently, researchers began formulating the dispatching problem as an optimization task, leading to the development of variants of the Vehicle Routing Problem (VRP), network flow models, and dynamic matching algorithms [7, 16]. For instance, the work by Yan et al. [35] integrates dynamic pricing and adaptive waiting mechanisms to enhance ride-hailing platforms. Although such optimization-based methods offer theoretical advantages, they often depend on strong assumptions about system behavior, limiting their adaptability in real-world applications.

In recent years, with the increasing availability of large-scale trajectory data and advancements in computational power, data-driven methods have garnered significant attention. Researchers have employed time series models, machine learning, and deep learning techniques to forecast passenger demand, identify hotspots, and predict order information [19, 21, 32], which serve as the basis for designing more efficient dispatching strategies. For example, Si et al. [26] proposed a multi-agent hierarchical reinforcement learning framework to optimize vehicle dispatch and route planning in intercity ride-sharing, thereby improving system profitability and order fulfillment rates. These methods have also been extended to incorporate more complex factors and application scenarios [3, 18, 33], addressing increasingly sophisticated dispatching tasks. For instance, Li et al. [13] addressed the scheduling and routing problem of a heterogeneous fleet consisting of vehicles and drones for same-day delivery, proposing a novel hierarchical decision-making approach based on deep reinforcement learning.

Although multi-agent methods are receiving growing attention [24], enabling agents to effectively filter and utilize peer information in complex interactions for improved decision-making, current research and analytical tools remain limited in their ability to intuitively understand these intricate interaction processes and to reveal the emergent mechanisms of collective intelligence.

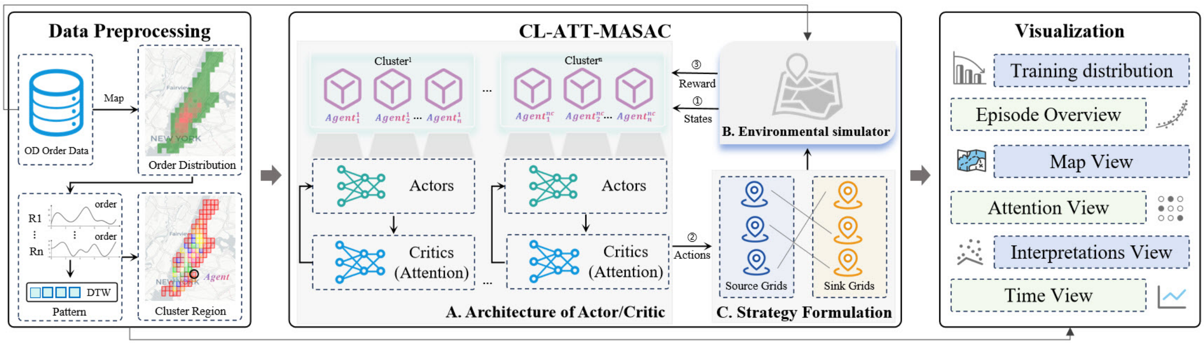


Figure 2: Overview of the framework for this work consists of three modules: (1) Data Preprocessing, performing regional order matching based on grid partitioning results, and conducting clustering based on the DTW results of time series from each region to obtain clusters with similar patterns; (2) CL-ATT-MASAC, regional multi-agent reinforcement learning process, including Architecture of Actor/Critic, Environmental Simulator, and Strategy Formulation; and (3) Visualization, interactive visual analytics focusing on model training performance and policy learning trajectories, aiming to enhance model interpretability.

2.2 Application of Reinforcement Learning

In recent years, reinforcement learning (RL) has gained significant attention as a learning paradigm capable of autonomous decision-making and policy optimization in dynamic environments. With the introduction of deep neural networks, deep reinforcement learning (DRL) has demonstrated remarkable performance in high-dimensional state spaces and has been widely applied in areas such as resource scheduling, traffic signal control, and robotic navigation [30, 37]. To address the challenges posed by complex tasks and large-scale environments, researchers have proposed various extensions, including MARL, hierarchical reinforcement learning, and model-based RL, to enhance scalability, stability, and generalization [8, 20].

In the transportation domain, RL has been extensively used to tackle problems such as dynamic taxi dispatching [4], vehicle routing [22], electric vehicle charging scheduling [34], and optimization of ride-sharing platforms [2], demonstrating superior adaptability and performance over traditional methods. For instance, Shi et al. [4] developed a multi-agent platform, MARL4T, for modeling large-scale urban vehicle dispatching. In addition, recent studies have begun to focus on multi-objective optimization and uncertainty modeling, exploring how to balance dispatching efficiency with user satisfaction, energy consumption, and system robustness [5, 40].

Despite the growing application of RL in transportation and urban computing, its “black-box” nature remains one of the major obstacles to real-world deployment. Since agent behaviors rely on the complex learning of state-action mappings and long-term reward functions, the resulting policies are often difficult to interpret or validate, posing challenges to system trustworthiness and human-AI collaboration [17, 23]. This issue is further exacerbated in multi-agent scenarios, where interactions among individual policies increase the overall model opacity [14]. Therefore, leveraging visual analytics or interpretability mechanisms to help users understand the policy learning process, environmental perception, and behavioral evolution of agents has become a critical direction in current RL research.

2.3 Visual Analytics for Reinforcement Learning

In RL tasks, visual analytics serves as a vital tool for understanding policy evolution, state space exploration, and reward dynamics [29]. Early efforts primarily focused on visualizing fundamental RL elements, such as agent learning trajectories, heatmaps of value functions (e.g., Q-values), and probability distributions of policy choices. These visualizations help users gain insights into agent behavior patterns and learning processes [27, 28]. For deep reinforcement learning models, researchers have also employed techniques to visualize neural network activations, attention mechanisms, and saliency maps to provide preliminary explanations for agent decision-making [6, 31].

As MARL scenarios grow increasingly complex, some studies have begun leveraging visual analytics to analyze agent interactions, cooperation and conflict, and the emergence of group strategies [38]. These systems typically offer interactive interfaces that enable users to track learning dynamics, compare different strategies, and begin to interpret the decision logic of individual or collective agents. For instance, DRLViz [9] integrates interactive memory dimension reduction and error analysis tools to assist experts in understanding the complex internal memory and decision-making mechanisms of DRL agents, thereby enhancing overall model interpretability. Similarly, the system developed by Shi et al. [25] helps users gain deeper insight into the training processes and learned cooperative strategies of multi-agent Deep Deterministic Policy Gradient (DDPG) models. Therefore, developing interactive visual analytics platforms holds promise for improving the interpretability of RL-based dispatching strategies. This study is the first to introduce visual analytics into the vehicle dispatching problem, advancing explainability research in complex multi-agent decision-making tasks.

3 Overview

3.1 Data Description

In this study, the experimental data are sourced from the publicly available Taxi and Limousine Commission (TLC) Trip Record Data released by the New York City TLC. This dataset has been continuously published since 2013 and includes extensive trip records from yellow taxis, green taxis (which serve different city zones based on vehicle color), and for-hire vehicles such as Uber and Lyft. We primarily use the yellow taxi trip data, as these vehicles mainly operate in Midtown Manhattan and other major commercial areas, constituting the core of New York City’s street-hail taxi services.

Each trip record includes detailed information such as pickup and drop-off timestamps, pickup and drop-off locations, trip distance, itemized fare components, rate code, payment method, and the number of passengers reported by the driver. In our experiments, we use a total of 269,154 trip records from January 1, 2016, for model training, and 240,872 trip records from January 3, 2016, for model testing.

3.2 Design Goals

The primary goal of this visual analytics system is to facilitate comprehensive analysis of complex dynamics in MARL-based taxi dispatch frameworks. *DpLens* is designed to improve the interpretability of learned dispatching policies, assist in diagnosing emergent agent behaviors, and support iterative policy refinement to enhance scheduling efficiency and system reliability. To ensure practical relevance, the system was developed through an iterative co-design process with experts in MARL and intelligent transportation. Discussions focused on key challenges such as understanding policy evolution, analyzing coordination mechanisms among agents, and validating model-driven decision processes. These efforts led to the formulation of four design goals:

G1: Support comprehensive monitoring and diagnosis of MARL model performance and policy learning. Provide an overview of model effectiveness and learning progress, enabling detection of abnormal behaviors at both global and regional scales.

G2: Enable multi-level exploration of learned policies and agent behaviors. Support spatiotemporal analysis of agent decisions, state transitions, and rewards from both macro and micro perspectives.

G3: Reveal coordination mechanisms and emergent system dynamics. Analyze inter-agent cooperation and vehicle flow to understand how local actions lead to system-level behaviors and efficiency.

G4: Assist in interpreting key decisions and support interactive policy optimization. Identify critical states and events affecting performance, and enable comparison of different policy versions to enhance model reliability.

3.3 Analysis Framework

Based on the aforementioned objectives, we have designed and implemented a visual analytics framework tailored for taxi dispatching, as illustrated in the Figure.2. This framework consists of three main modules: data preprocessing, CL-ATT-MASAC, and visual analytics. In the data preprocessing module, we first match taxi order data to spatially partitioned urban grids, generating region-level demand time series. To identify areas exhibiting similar temporal demand dynamics, we employ Dynamic Time Warping (DTW) to compute pairwise distances between regional time series, followed by clustering on the resulting distance matrix. The derived clusters serve as the foundation for defining and modeling agents in the subsequent reinforcement learning module. The CL-ATT-MASAC module features a region-level MARL framework. Each agent is modeled using an Actor/Critic architecture and is situated in a simulated environment replicating real-world dispatch scenarios. Building upon the Soft Actor-Critic algorithm, we integrate an attention mechanism to enhance inter-agent communication, thereby improving the effectiveness of policy learning and coordinated decision-making in complex urban traffic contexts. In the visualization module, we integrate the analytical and training outputs from the previous modules into an interactive visual interface. This system supports the tracking of the reinforcement learning training process, enables exploration of agent behavior trajectories, and facilitates comparison of learned strategies. By incorporating human-in-the-loop analysis, the visualization module enhances the interpretability of learned policies, assists domain experts in understanding the underlying decision logic, and ultimately contributes to optimizing dispatch strategies and increasing the trustworthiness of model deployment.

4 Methodology

Our proposed method aims to address the dynamic nature of taxi dispatching in complex urban environments and mitigate instability during the agent policy learning process. It comprises three main components: travel pattern-based regional clustering, an CL-ATT-MASAC MARL module, and a scheduling strategy execution module based on bipartite graph matching. The overall workflow of the method is illustrated in Figure.3.

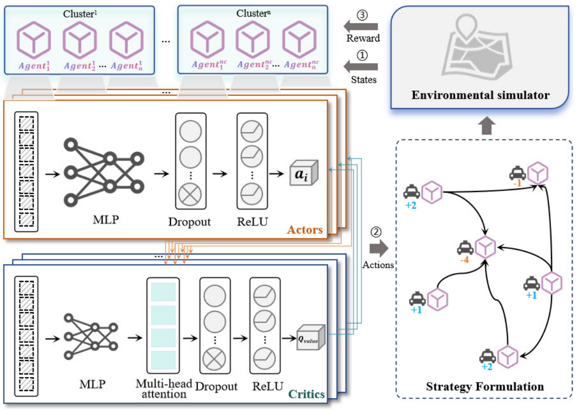


Figure 3: Detailed architecture of CL-ATT-MASAC. Clustered agents share parameterized Actor networks. The Critic employs multi-head attention to capture inter-agent interactions. Output actions are passed to the scheduling module and environment simulator.

4.1 Regional Clustering and Agent Construction

To mitigate the learning interference caused by regional heterogeneity in large-scale multi-agent systems, we adopt a demand-driven clustering approach before training. By analyzing travel demand time series, regions exhibiting similar temporal patterns are grouped into clusters. Agents within the same cluster share a common strategy network, promoting coordinated learning and reducing redundancy. This cluster-based design not only enhances cooperation among agents but also improves the overall learning efficiency and policy stability. The clustering process involves the following key steps:

Regional Time Series Construction: We divide the research city into $n = 111$ non-overlapping spatial grid regions and the day into $m = 144$ 10-minute time slots. The order volume in each time slot is counted for each region, forming the daily order demand sequence of that region. The demand sequence for region i is denoted as:

$$d_i = [d_i^1, d_i^2, \dots, d_i^{144}] \tag{1}$$

where d_i^t represents the number of orders in region i at time slot t.

Inter-Region Similarity Calculation: We use the DTW method to calculate the similarity between the time series of any two regions. The DTW distance between regions i and j is defined as:

$$\mathrm{DTW}(i, j) = \min_{P \in P} \sum_{(u,v) \in P} |d_i^u - d_j^v| \tag{2}$$

where P represents all valid alignment path sets. DTW is particularly suitable for modeling temporal similarities between sequences with time offsets or rate differences.

Regional Clustering: After obtaining the DTW distance matrix between regions, we perform agglomerative hierarchical clustering. The optimal number of clusters K is determined using the elbow method, which evaluates the within-cluster sum of squares (WCSS). Once clustering is completed, the regions are grouped into clusters

$C = \{c_1, c_2, \ldots, c_K\}$. Each cluster shares the same strategy network. The input state tensor of cluster k includes n_k agents, with each agent having a state representation of dimension $\dim_s$.

4.2 Clustered Attention-based Multi-Agent Soft Actor-Critic

Capturing the dynamic interactions among multiple taxis and the stochastic nature of the urban environment requires a principled decision-making framework. We therefore formulate the scheduling task as a Markov game, which enables each agent to make sequential decisions based on its local observation while accounting for the influence of other agents' actions over time. Specifically, the problem is modeled as a Markov game $G = \langle n, S, A, P, R, \gamma \rangle$, where n is the number of agents, S is the state space, A is the action space, P is the state transition probabilities, R is the reward function, and γ is the discount factor. Therefore, the state vector of region i at time slot t is defined as:

$$o_i(t) = \left[\frac{S_i(t) - D_i(t) - \mu}{\sigma}, \sin\left(\frac{2\pi \cdot h}{24}\right), \cos\left(\frac{2\pi \cdot h}{24}\right) \right] \tag{3}$$

where $D_i(t)$ and $S_i(t)$ represent the predicted demand and supply of region i at time t, respectively; μ and σ denote the mean and standard deviation of the training data used for normalization; and $h = \lfloor (t \bmod 144)/6 \rfloor$ denotes the hour (0–23) corresponding to time slot t.

The agent selects actions based on the strategy network μ_{θ_i}:

$$a_i(t) = \mu_{\theta_i}(o_i(t)), a_i(t) \in [-1, 1] \tag{4}$$

To further enhance this modeling capacity, we introduce a multi-head attention mechanism into the Critic. This design enables the network to attend to different aspects of inter-agent dependencies in parallel. Each head independently learns a unique representation of interaction patterns, allowing the Critic to integrate diverse relational information. This multi-head formulation significantly boosts the flexibility and robustness of strategy evaluation, especially in highly dynamic and heterogeneous environments.

The learning process in our CL-ATT-MASAC framework involves both Actor and Critic networks. The Actor network for each agent i is optimized via policy gradient, which encourages actions that lead to higher estimated Q-values:

$$\nabla_{\theta_i} J(\mu_i) = \mathbb{E}_{X,A} \left[\nabla_{\theta_i} \mu_i(o_i(t)) \cdot \nabla_{a_i} Q_{\phi_i}(X_t^c, A_t^c) \right] \tag{5}$$

Meanwhile, the Critic network is trained by minimizing the mean squared error between the predicted Q-value and the target value computed using the next-step Q-value and reward:

$$\begin{gathered} L(\phi_i) = \mathbb{E}_{X,A,r,X'} \left[\left(Q_{\phi_i}(X_t^c, A_t^c) - y \right)^2 \right], \\ y = r_i(t) + \gamma Q_{\phi_i'}(X_{t'}^c, A_{t'}^c) \end{gathered} \tag{6}$$

To enhance the critic network's capability in modeling key interaction patterns within the joint state-action space, this work extends the critic architecture in the MASAC framework by incorporating a multi-head attention mechanism. This mechanism dynamically captures important inter-agent dependencies and enables weighted information aggregation. Specifically, each agent's state is used as the query, while the concatenated states and actions of all agents are embedded as keys and values. Attention weights are computed based on query-key similarity, and a weighted sum over the values produces the interaction embedding. Multiple attention heads operate in parallel to extract diverse interaction perspectives, which are then concatenated and fed into the Q-network for value estimation.

$$\text{head}_h = \text{Softmax}\left(\frac{(o_i W_h^Q)(o_j W_h^K)^T}{\sqrt{d_k}} \right)(o_j W_h^V) \tag{7}$$

where o_i represents the state of the current agent, o_j is the embedding of the concatenated states and actions of other agents, and W_h^Q, W_h^K, W_h^V are the parameters for the h-th attention head.

The immediate reward $r_i(t)$ is determined by both the demand-supply ratio in region i at time t and the agent's action, where the $\rho_i(t)$ reflects the local imbalance between customer requests and available vehicles in that region.

$$\rho_i(t) = \frac{d_i(t)}{s_i(t)} \tag{8}$$

4.3 Scheduling Strategy Execution through Bipartite Graph Matching

The reinforcement learning model outputs continuous actions $a_i \in [-1, 1]$, which we discretize using a threshold ξ:

$$\begin{cases} \text{If } a_i > \xi, & \text{region is a source(send);} \\ \text{If } a_i < -\xi, & \text{region is a sink(accept);} \\ \text{If } |a_i| \le \xi, & \text{region is idle(do\_nothing).} \end{cases} \tag{9}$$

The number of vehicles scheduled from the sending region is $x_i = |a_i \cdot s_i|$, and the expected dispatch amount for the receiving region is $y_j = |a_j \cdot d_j|$.

We model the scheduling problem as a bipartite graph $G = (V_{\text{src}}, V_{\text{sink}}, E)$, where the edge weight d_{ij} represents the Manhattan distance. The goal is to maximize the supply-demand matching while minimizing the total scheduling distance, with the optimization objective:

$$\max_{(i,j)\in M} \sum -d_{ij} \cdot |x_i - y_j| \tag{10}$$

To handle differing numbers of source and sink nodes, we introduce virtual nodes, transforming the problem into a standard assignment problem. This is solved using the Hungarian algorithm (Kuhn-Munkres Algorithm) to find the optimal scheduling pairs, which are then used to update vehicle states in the environment simulator, completing the training-execution closed loop.

5 Visual Design

5.1 Control Panel

To support analysts in flexibly configuring training parameters, environmental settings, and algorithmic models, we designed a control panel view (as shown in Figure.1A). This view enables users to select training- and environment-related parameters, switch between different MARL strategy models, and specify the Epoch, Time, and RegionID for the current system analysis. In addition, users can adjust key hyperparameters, including the learning rate (LR), discount factor (Gamma), soft update coefficient (Tau), and batch size, other views will be updated accordingly with these parameters. By clicking the "Submit Config" button, users can submit the configured parameters to initiate the training process. The intermediate results are saved to the system's data stream to support subsequent multi-perspective visual analysis.

5.2 Training Distribution View

We designed the Training Distribution View (as shown in Figure.1B1) to support analysts in comprehensively monitoring the training performance and policy learning effectiveness of MARL models (G1). This view dynamically presents the distribution trends of key performance metrics—such as Actor Loss, Critic Loss, Alpha Loss, Reward, and Match Rate—through a set of multi-dimensional metric curves. It enables users to grasp the model's convergence process from a global perspective and promptly identify performance fluctuations and training anomalies. The horizontal axis represents the training iterations, while the vertical axis in each sub-view reflects the value changes of the corresponding metric, intuitively illustrating the dynamic adjustment and optimization of the model throughout training. To enhance analytical flexibility, the view also provides interactive controls including "Sort by metric" and "Order," allowing users to rearrange the display order and prioritize metrics of interest based on specific analytical goals. This facilitates focused tracking of critical indicators and the early detection of anomalies (G2).

5.3 Table View

Aiming to facilitate rapid access to and comparison of key scheduling system data across different scales, we designed the Table View (as shown in Figure.1B), which provides a multi-dimensional, region-centric (Region ID, or RID) information display. It supports fine-grained data exploration and interactive analysis based on specific epochs and time steps. At the foundational level, the view presents core metrics for each region within the current epoch, including Total Demand (TDem), Cluster ID (CID), Match Rate (MRate), and Demand Tendency. These indicators help users gain a global understanding of model performance at the regional level, enabling timely identification of potential issues such as demand surges or low match rates (G1).

To support more detailed analyses at the epoch-time granularity, the view further includes additional columns such as Action Value, Reward, Supply-Demand Ratio, and Imbalance (supply-demand gap). These metrics are enriched with lightweight visual elements such as bar indicators, ring charts, and micro trend plots, allowing users to intuitively compare policy performance and agent behavior differences across regions and time periods (G2). For enhanced analytical flexibility, the Tabular View supports customizable column ordering, enabling users to rearrange fields based on specific task priorities. This empowers targeted exploration of key metrics and improves insight efficiency (G1, G2). Overall, the view enables seamless navigation between macro-level patterns and micro-level details, supporting comprehensive understanding and anomaly diagnosis from a global-to-local perspective (G1).

5.4 Episode Overview

To help analysts gain a comprehensive understanding of the policy behavior distribution and decision evolution within a specific region for each epoch, we designed the Episode Overview View (as shown in Figure.1C). This view focuses on the target analysis region (RegionID) across all time steps within the current epoch. By integrating dimensionality-reduced state layouts, action type encodings, and time-series mappings, it supports interactive analysis from global states to localized behaviors (G1, G2).

The scatterplot (Figure.1C1) visualizes the dimensionality-reduced regional state vectors(Equation 3) using t-SNE, where each point represents the RegionID's state at a specific time step, RegionID can be specified through the dropdown menu in the control panel. The shape of each point encodes the agent's action decision under that state, intuitively revealing the distribution patterns of policy behaviors in the state space and facilitating the identification of behavioral evolution paths and localized strategy variations. The color intensity reflects the temporal order, with darker colors indicating later time steps, allowing analysts to observe the phase-wise changes and transitions of agent behavior over time (G2).

Interactive rectangular brushing is supported, enabling users to select subsets of interest within the state space and filter specific temporal segments, thereby enhancing hierarchical exploration from a global to a local level (G2). The filtering results will be linked with the Supply-Demand Imbalance (SDI) matrix view on the right (Figure.1C2). By default, the right panel displays the average actions and SDI status across all time slices for the region specified by the selected RegionID in the control panel, along with its neighboring regions(G1). If the user selects a specific time range in the scatter plot on the left, the matrix on the right will be updated accordingly to reflect the average actions and SDI distribution for the selected time period, facilitating time-specific analysis and comparison. In the matrix, the fill color of each cell encodes the SDI value—green indicates sufficient supply, red denotes excessive demand, and color intensity reflects the magnitude of the imbalance—helping analysts diagnose the relationship between local environmental states and agent strategies (G1). Arrows within the matrix indicate inter-regional dispatch flows, pointing toward the dispatched regions. The thickness of each arrow represents the number of vehicles dispatched, providing a clear depiction of the agent's spatial decision-making and resource movement strategies.

5.5 Map View

The Map View (Figure.1D) helps analysts explore the operational status and policy effects of the scheduling system across time and space, enhancing strategy interpretability and situational awareness. To reveal spatial similarities and differences among regions, a cluster-based color layer is overlaid on the map. The Demand Layer (Figure.1D1) visualizes regional demand (Demand) using a heat-grid, where color depth in an orange gradient encodes demand intensity, making it easy to identify high-demand hubs and low-demand peripheral areas. Users can switch to the Supply-Demand Imbalance (SDI) Layer (Figure.1D2), where red-green colors represent SDI values (as in Figure.1C2), and saturation indicates the magnitude of imbalance. This helps quickly locate areas with mismatched supply and demand (G1, G3).

To support the analysis of spatiotemporal scheduling behavior, the view includes a time playback feature (Figure.1D3). Users can navigate through time using a slider or playback controls to observe how agent behavior and system states evolve over an epoch. Vehicle dispatches between regions are visualized using streamlines: the origin region is marked with a dot, and the thickness of the streamline encodes the number of dispatched vehicles. This design facilitates understanding of agents' decision-making and coordination mechanisms under varying conditions (G2, G3).

5.6 Time View

The Time View (Figure.1E) enables analysts to explore the micro-level behavioral evolution of scheduling strategies from a temporal perspective (G2). The vertical axis represents different regions, while the horizontal axis denotes time steps. Each cell corresponds to a specific region at a particular time, showing the agent's action and the resulting reward. This design balances fine-grained local insights with global time-series readability, supporting multi-scale analysis from detailed spatiotemporal behavior to system-wide trends (G1, G4).

Within each cell, a circular marker encodes the agent's action type using color: green for Send, orange for Accept, and gray for Do Nothing, clearly illustrating spatiotemporal patterns and agent preferences (G2). The transparency of the color reflects the action value, which indicates the proportion of vehicles dispatched or received. Surrounding each circle, semi-arc indicators further encode feedback signals: the right arc shows positive (blue) and negative (red) rewards at that time step, while the left arc represents positive (purple) and negative (green) state changes. These visual cues reveal the causal relationship between agent decisions and environmental feedback (G3).

5.7 Demand View

Demand View (Figure.1F) is designed to support analysts in understanding demand pattern differences and their evolution across multiple regions from both temporal and clustering perspectives. The view adopts a radial multi-layer bar chart layout, unfolding average demand over time for each cluster during the scheduling cycle. This design enhances intuitive perception of global dynamics and key time periods (G1). Each ring represents a cluster, with the circular direction encoding time, advancing clockwise to reflect temporal progression. Segments within each ring represent demand intensity at specific time steps, filled with an orange gradient—darker shades indicate higher demand, making it easier to spot peaks and troughs. Rings are arranged from inner to outer based on total demand per cluster, highlighting the "core–periphery" structure of regional demand. A cluster selection panel at the top (e.g., "Cluster Average") allows users to focus on individual clusters or compare multiple clusters, supporting layered analysis between local and global demand patterns (G2).

5.8 Attention View

Designed to help analysts understand the spatial dependency modeling of the CL-ATT-MASAC model and reveal the Critic's attention patterns and key decision features during policy learning, the Attention View (Figure.1G) visualizes how the model captures spatial correlations between regional states across different epochs, enhancing the interpretability of the policy learning mechanism (G4). A matrix layout displays region IDs on both axes, where each cell represents the attention weight assigned by the Critic under specific clusters and attention heads. Deeper green shades indicate higher importance, revealing the strength of spatial dependencies (G1). Interactive controls located at the top-right allow switching between cluster IDs and attention heads, facilitating multi-scale and multi-path analysis of the model's attention differences (G2). Throughout the training process, analysts can track changes in the model's focus on core, peripheral, and special-state regions. Interactive exploration supports investigating coordination tendencies and spatial decision-making mechanisms during policy evolution (G3).

5.9 Policy Interpretation View

To help analysts understand agent policies in high-dimensional state spaces and their relationships with state features, we designed a policy interpretation view (Figure.1H). This view visualizes the distribution of agent action preferences across different states using a 3D embedding approach, supporting interpretability analysis and key behavior identification (G4).

Since the internal structure of traditional Actor networks is invisible and difficult to interpret, this view offers a visual approximation of the Actor's decision-making structure, enabling researchers to infer the Actor's action selection patterns and preference structures under various state conditions, thereby aiding in the analysis of the learning process. Specifically, the view maps state vectors to three dimensions: normalized supply-demand difference (Equation 3) and hourly cyclic features (sine and cosine values), to capture key state characteristics. Action types are color-coded to intuitively represent the distribution of actions across different states. Cluster centers (pink polyhedra) represent stable and typical action patterns output by the Actor within a state cluster, reflecting the main trends of the policy. Boundary points (black) indicate transition regions between different policy patterns, revealing strategy shifts and boundary structures that help explain the continuous variation of policies in the state space. Through this interpretability view, analysts can assess the behavioral stability and policy reliability of the Actor during the multi-agent reinforcement learning process.

6 Evaluation

6.1 Quantitative Analysis

To further validate the effectiveness of our proposed method in practical scheduling tasks, we conducted comparative experiments between CL-ATT-MASAC and the existing state-of-the-art method META [15], the results are shown in Table.1.

Table 1: Comparative Results with Baseline Method.

Model	Order Matching Rate (%)	Avg. Relocation Count
META	62.50	10.97
Ours	**72.16**	**6.53**

Table 2: Ablation Study Results on Attention and Cluster Modules.

Model	Order Matching Rate (%)	Avg. Relocation Count
MASAC	56.89	10.97
ATT-MASAC	64.56	10.23
CL-MASAC	58.47	9.88
Ours	**72.16**	**6.53**

It can be seen that CL-ATT-MASAC significantly outperforms META in order match rate, with an improvement of 9.66% (from 62.50% to 72.16%), indicating that our method can allocate orders more efficiently and effectively alleviate resource idleness. Additionally, the average number of relocations in CL-ATT-MASAC decreased from 10.97 to 6.53 (a reduction of 4.44), greatly enhancing scheduling efficiency and reducing system operating costs. These results demonstrate that our method not only improves service quality but also achieves better resource utilization efficiency.

In the ablation study, we analyzed the impact of introducing the attention mechanism (ATT) and the clustering module (CL) within the multi-agent reinforcement learning framework on model performance, focusing on their effects on order match rate and average relocation times. The results are shown in Table.2. It can be observed that when both the attention mechanism and clustering module are incorporated, the model achieves the best performance in terms of order match rate and scheduling efficiency. The match rate reaches 72.16%, an increase of 15.27% compared to the baseline MASAC, while the average number of relocations decreases to 6.53, a reduction of 4.44 times. This indicates that the synergy between the attention mechanism and contrastive learning significantly optimizes scheduling strategies.

Further analyzing the individual effect of the attention mechanism, ATT-MASAC achieves a match rate of 64.56%, improving by 7.67% over MASAC, with average relocations reduced to 10.23. This shows that the attention mechanism helps more effectively capture task-relevant information and improve matching efficiency. When only the contrastive learning module is added (CL-MASAC), the match rate slightly increases to 58.47%, but the average relocations significantly decrease to 9.88, indicating that contrastive

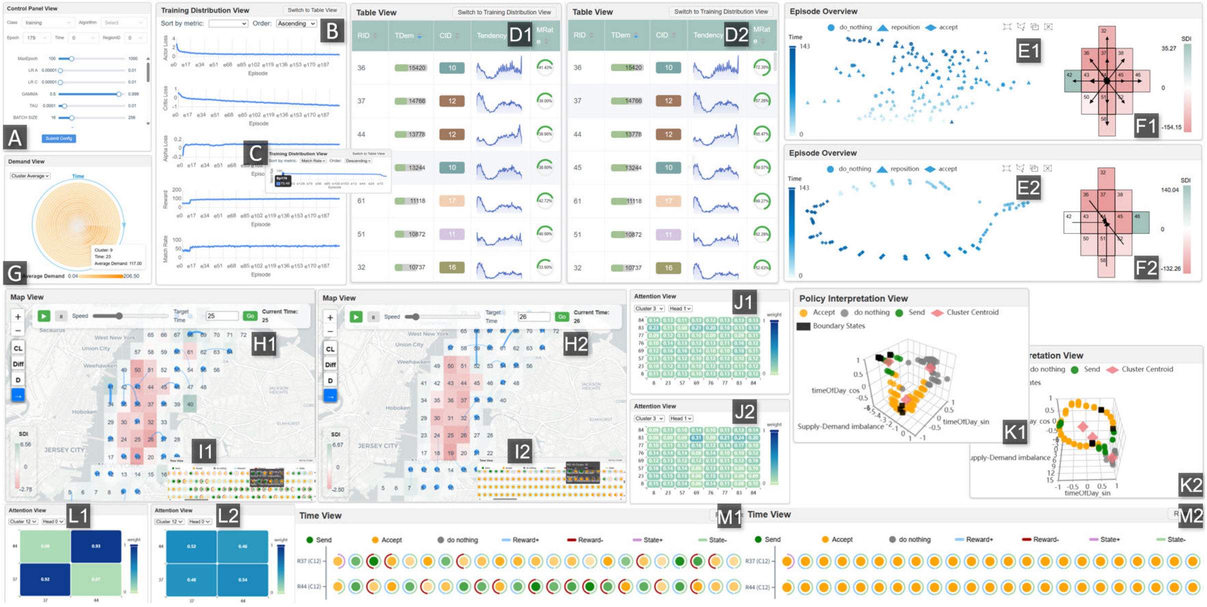


Figure 4: This figure illustrates how *DpLens* facilitates multi-level interpretation of scheduling behavior evolution in a reinforcement learning model across training epochs. (A) Users configure training parameters, enabling PER for enhanced sample efficiency. (B–C) Training Distribution View shows improvement in matching rates, reaching 72.45%. (D) Table View reveals increased dispatch effectiveness in high-demand regions. (E–F) Episode Overview and Time View demonstrate a shift from erratic to stable agent actions. (G) Demand View highlights peak-period demand pressure. (H–I) Map and Reward Views show strategy optimization from misaligned to coordinated spatial decisions. (J) Attention View indicates enhanced global reasoning via focused inter-agent attention. (K) Policy Interpretation View reveals the emergence of structured, interpretable action clusters over time.

learning helps enhance the model's ability to distinguish states and optimize scheduling decisions.

In summary, the attention mechanism contributes more to improving the match rate, while contrastive learning tends to optimize relocation efficiency. Their combination maximizes performance gains, verifying the effectiveness of the module design.

6.2 Case Study

We conducted three case studies to evaluate the effectiveness of our system. The first focuses on the interpretable analysis of model training mechanisms, while the second explores region-level scheduling optimization through interactive feedback. The third case compares the adaptability and regional cooperation capabilities of different methods in dynamic environments.

6.2.1 Interpretable Analysis of Model Training Mechanisms

In this case study, model developers focused on the evolution of scheduling behaviors in a RL model, aiming to trace the policy adaptation process across different training epochs using *DpLens*.

Users begin by configuring training parameters (Figure.4A), enabling Prioritized Experience Replay (PER) for improved sample efficiency. Upon submission, the system initiates training and logs key metrics such as loss, supply, dropped orders, and matching records. In the Training Distribution View, early training stages (Figure.4B) show low and unstable matching rates, which later converge to 72.45% (Figure.4C), indicating growing strategy effectiveness.

To analyze temporal and spatial changes, users compare model performance at epoch 0 and epoch 179. In the Table View, high-demand regions (e.g., 36, 37, 44) show poor early performance (40%, Figure.4D1), improving significantly by epoch 179 (to 72.30%, 57.26%, and 65.47%, Figure.4D2). The Episode Overview and Time View reveal initially erratic agent behavior (Figure.4E1), which becomes consistent and policy-aligned in later epochs (Figure.4E2), with smoother temporal patterns (Figure.4F2) indicating stable strategies.

During peak demand (time slices 0–20), the Demand View highlights system pressure (Figure.4G). In the Map View, early epochs show agents wrongly dispatching from high-demand areas (Figure.4H1), leading to negative rewards (Figure.4I1). By epoch 179, agents instead guide vehicles into these regions (Figure.4H2), earning positive rewards (Figure.4I2), demonstrating improved spatial coordination.

The Attention View shows attention weights dispersed in early stages (Figure.4J1), but later epochs reveal focused attention on key agents (e.g., 83, 69) and critical areas (Figure.4J2), enhancing global decision-making. In the Policy Interpretation View, initial policy distributions are scattered and unclear(Figure.4K1), while later strategies form distinct, interpretable clusters with clearer action boundaries in high-imbalance areas(Figure.4K2).

Moreover, as shown in the attention matrices of epoch 0 and epoch 179 (Figure.4L1, L2), agents initially focus predominantly on their own states (e.g., self-attention scores ¿ 0.9 for R37 and R44), exhibiting limited attention to other agents within the same cluster. In contrast, by epoch 179, cross-agent attention significantly increases-for instance, R37's attention to R44 rises to 0.54-indicating that agents progressively learn to attend to their cluster peers, thereby enabling coordinated policy optimization. This collaborative behavior is further supported by the Time View results. In the early training phase (Figure.4M1), R37 and R44 exhibit independent behaviors, while in the later phase (Figure Figure.4M2), their actions become highly synchronized, demonstrating the effect of cluster-based cooperative learning. Specifically, as R37 adjusts its dispatching strategy, R44 dynamically adapts its own decisions in response, forming a collaborative relationship. Guided by the attention

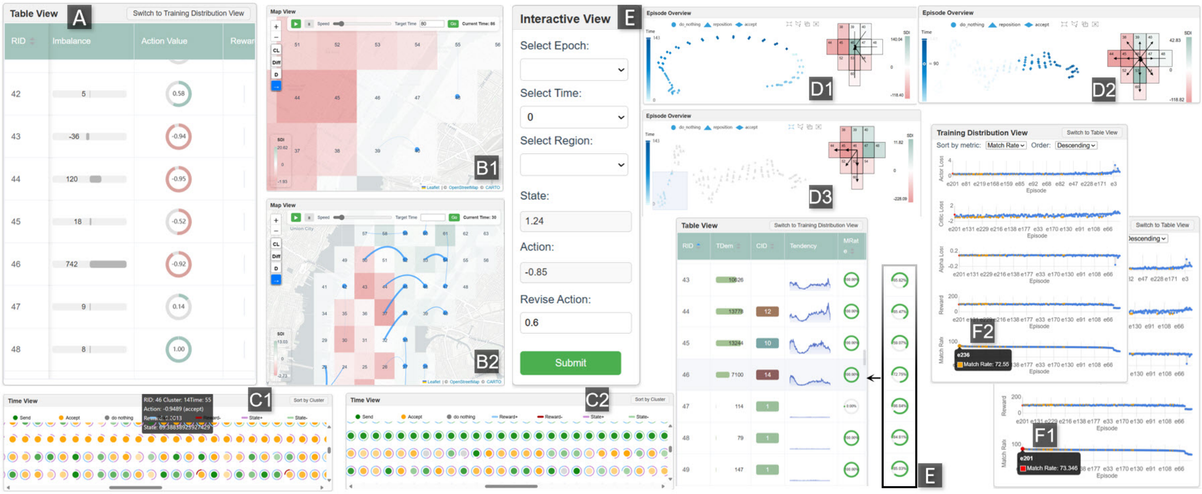


Table View

RID	Imbalance	Action Value
42	5	0.58
43	-36	-0.94
44	120	-0.95
45	18	-0.52
46	742	-0.92
47	9	0.14
48	8	1.00

Interactive View

Select Epoch:

Select Time: 0

Select Region:

State: 1.24

Action: -0.85

Revise Action: 0.6

Submit

Table View

RID	TDem	CID	Tendency	MRate	
43	10826			100.00%	95.82%
44	13778	12		100.00%	95.47%
45	13244	10		100.00%	59.07%
46	7100	14		100.00%	72.75%
47	114	1		0.00%	86.84%
48	79	1		100.00%	84.81%
49	147	1		100.00%	85.03%

Figure 5: Interactive policy refinement based on human feedback. (A) The Table View reveals severe oversupply and negative action values in high-imbalance periods. (B1) The Map View shows poor spatial coordination, with surrounding high-demand regions receiving no support. (C1) The Time View indicates persistent "Accept" actions and zero rewards. (D1–D2) The Episode Overview confirms ineffective routing; surrounding areas remain demand-heavy. (E) The expert adjusts action values via the Interactive View for time slices 0–90, guiding the system to perform "Send" actions. (B2, D3) At epoch 201, the model exhibits spatially coherent dispatch behavior. (C2) The Time View shows positive rewards and adaptive action changes after slice 90, reflecting improved policy responsiveness.

mechanism, this joint policy optimization helps alleviate local congestion and enhances overall reward and system efficiency.

Overall, the RL model evolves from local, chaotic decisions to globally coordinated, policy-consistent behaviors. Experts indicated that the multi-level visual analysis provided by DpLens enhances model interpretability and provides strong support for scheduling strategy optimization.

6.2.2 Dispatching Model Optimization Based on Interactive Feedback

This case focuses on a human dispatcher (E3) with extensive experience in urban mobility scheduling. E3's primary responsibility is to identify suboptimal strategy regions based on model training outcomes and intervene through interactive operations to improve overall system efficiency.

At the beginning of the analysis, E3 loads the results from training epoch 179, where the matching rate has nearly converged to its upper bound. However, he quickly notices significant strategic flaws persisting in certain local areas. In the Table View (Figure .5), E3 examines data from time slice 44 and observes that Region 46 has a severe supply-demand imbalance value of 742, indicating an extreme oversupply of vehicles. Surprisingly, the action value assigned to this region at that time is -0.92, corresponding to the "Accept" action, meaning the system continues to dispatch additional resources into the area.

E3 cross-validates this observation in the Map View (Figure.5B1) and confirms that Region 46 not only accumulates excessive idle vehicles but also borders several high-demand regions such as 44, 45, and 52. Despite this, the system fails to guide surplus vehicles outward effectively. The Time View (Figure.5C1) further reveals that Region 46 consistently performs the "Accept" action across multiple high-imbalance moments, while its reward signal remains at zero for an extended period, reflecting inefficient decision-making without positive environmental feedback. The Episode Overview (Figure.5D1) shows that Region 46 maintains the "Accept" action across all time slices, with no signs of effective dispatching behavior. As illustrated in Figure.5D2, surrounding regions remain demand-heavy throughout, yet vehicles are still routed into Region 46. Based on these findings, E3 concludes that Region 46 should instead perform a "Send" action and decides to intervene manually. Using the Interactive View provided by the system (Figure.5E), E3 iteratively modifies the dispatch policy of Region 46 from epoch 179 onwards. For time slices 0–90 (after which the region becomes supply-deficient), he uniformly sets the action value to 0.6 to explicitly indicate proactive vehicle outflow. The system incorporates these interventions, retrains the model, and generates intermediate results at epoch 201 for performance validation.

In epoch 201, E3 observes a significant shift in the model's behavior. In the Episode Overview (Figure.5D2), Region 46 consistently performs the "Send" action (denoted by triangles) during the first 90 time slices, with vehicle outflows directed toward high-demand areas such as regions 44, 45, 52, 53, and 60, forming a reasonable spatial resource flow. In the Map View (Figure.5B2), Region 46 exhibits prominent outward blue arrows during several time slices, indicating high-volume dispatches that alleviate regional demand pressure. Additional inspection of early time slices confirms that the manual intervention led to vehicle redirection toward demand-heavy neighboring regions (Figure.5D3).

Temporally, the Time View (Figure.5C2) shows that Region 46 performs the "Send" action during most of the first 90 time slices and frequently receives clear positive reward signals (blue arcs), suggesting environmental validation of the revised strategy. More importantly, after time slice 90—when the region becomes supply-constrained—the model adaptively shifts its behavior from "Send" to "Accept", demonstrating its responsiveness to evolving supply-demand dynamics.

Finally, a comparison in the Table View (Figure.5E) reveals that the matching rate of Region 46 improved from 72.75% at epoch 179 to 100%. More importantly, this intervention triggered collaborative optimization in surrounding areas—for example, Region 45's matching rate increased from 59.07% to 100%, and regions 44 and 43 also experienced significant gains, indicating a spillover effect of the optimized policy. As a result, the overall system scheduling efficiency improved substantially.

It is worth noting that while the matching rate at epoch 201 increased significantly to 73.346% after expert intervention, this modification only altered the action outputs in response to environmental states and did not directly update the model parameters. Hence, its impact was limited to the current episode's behavior, without deeper adjustment to the policy network. To address this limitation, the system writes the high-value experience samples derived from human intervention into the PER buffer, assigning them a priority higher than any existing samples. This ensures their higher sampling probability in subsequent training. PER increases the reuse of experiences with large TD errors, allowing the model to revisit critical decisions more frequently, thereby accelerating policy convergence. In this case, PER effectively guided the model to repeatedly learn from the dispatcher's demonstration behavior, gradually internalizing it into the policy network. The model was then trained for an additional 50 epochs. As training progressed, performance continued to fluctuate upward, reaching a new peak matching rate of 72.55% at epoch 236 (Figure.5F2). Although the improvement margin was limited, it demonstrated the model's capacity to learn from human intervention. The Training Distribution View further confirms this: epoch 201 (Figure.5F1) shows the highest recorded performance, with epoch 236 closely following, reflecting successful strategy transfer and generalization.

This process illustrates a mechanism of human-in-the-loop policy fusion: expert interventions act as behavioral guidance, PER provides experience reinforcement, and continued training enables policy learning. The visual analytics system played a crucial role throughout, offering transparency and traceability to support deeper understanding and optimization of the policy evolution.

6.2.3 Comparative Analysis

To compare the scheduling adaptability and regional collaboration capabilities of different algorithms in highly dynamic environments, experts conducted 500 training episodes for MASAC, CL-MASAC, ATT-MASAC, and CL-ATT-MASAC with a fixed fleet size of 10,000. Match rate curves were generated, and all methods were tested in the same scenario. The results were loaded into a visual analytics system for comparison and analysis.

First, the training results (Figure. 6) show that MASAC performed the weakest, with match rates hovering around 50% for a prolonged period, indicating insufficient adaptability and collaboration. CL-MASAC showed rapid improvement early in training and stabilized around 60%. ATT-MASAC started slower but, leveraging the attention mechanism, improved over time to surpass CL-MASAC. CL-ATT-MASAC, which combines clustering and attention mechanisms, converged faster with smaller fluctuations and achieved significantly higher match rates, demonstrating the effectiveness of the collaborative optimization of both mechanisms.

Furthermore, experts selected Region 19 in New York City as a case study within the visual analytics system to examine algorithmic performance under dynamic demand conditions. As shown in Figure. 7A, this region experienced a sharp increase in demand during time steps 19–24, followed by a rapid decline in time steps 25–30, making it an ideal setting for evaluating the adaptability and coordination capabilities of different methods in realistic dispatch scenarios.

As illustrated in Figure.7B, during the demand surge (time steps 19–24), MASAC received only limited support from Region 10, reflecting weak regional collaboration. More critically, during the demand drop (time steps 25–30), MASAC continued to receive a large number of vehicles from Regions 10, 14, 15, and 21, demonstrating delayed response and inefficient dispatch that contradict the actual demand trend. In contrast, as shown in Figure.7C, CL-MASAC was able to dispatch vehicles to Region 19 from Regions 10, 14, and 21 in a timely manner during the rising demand phase. Notably, Region 14 contributed the most support, and the other two regions, which belong to the same cluster, exhibited relatively consistent dispatch behavior. However, the method still showed a certain degree of lag during the demand decline. Figure.7D shows that ATT-MASAC produced overall weaker responses. Although the attention mechanism is capable of capturing dynamic dependencies, it struggled to establish efficient cross-regional coordination in the presence of short-term, high-variance demand fluctuations, indicating limited effectiveness in such scenarios.

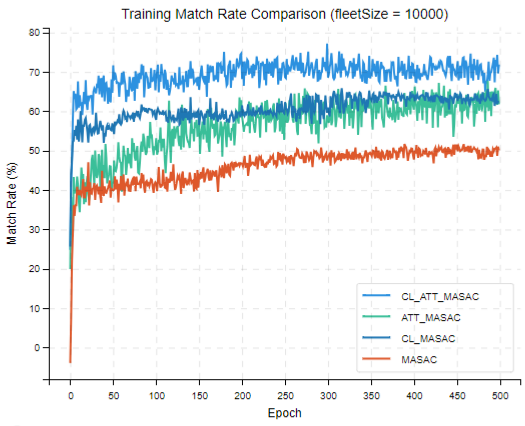


Figure 6: Training match rate curves of MASAC, CL-MASAC, ATT-MASAC, and CL-ATT-MASAC under a fixed fleet size of 10,000.

Among all methods, CL-ATT-MASAC demonstrated the best overall performance. As shown in Figure.7E, during the demand surge, it promptly dispatched vehicles from Regions 10, 15, and 21 to Region 19. Region 15, from a different cluster, provided flexible cross-cluster support, while Region 21, within the same cluster, offered stable and timely assistance—together forming an efficient collaborative dispatch pattern. During the demand drop, the method rapidly ceased unnecessary dispatches and retained only minimal, flexible support from Region 21, achieving fine-grained and low-redundancy resource allocation.

In summary, this case study reveals substantial differences among the four methods in terms of regional coordination, response efficiency, and adaptability under dynamic scheduling conditions. CL-ATT-MASAC exhibited superior responsiveness and collaboration in the face of sharp demand changes. In contrast, MASAC suffered from significant delays and inefficient dispatch. CL-MASAC and ATT-MASAC each showed advantages in early-stage responsiveness or long-term optimization but also demonstrated notable limitations.

6.3 User Study

A user study was conducted to evaluate the workflow, visual design, and usability of *DpLens*. It aimed to validate its effectiveness and gather feedback from target users for iterative refinement.

Participants. A total of 20 participants were recruited, including 5 transportation domain model developers (2 PhD holders and 3 senior engineers), 12 graduate students with backgrounds in visual analytics (9 master's and 3 doctoral students), and 3 professionals from urban traffic management departments (including expert E3 involved in the case study of this research). This diverse group ensured representation of both technical users and real-world domain experts, en-

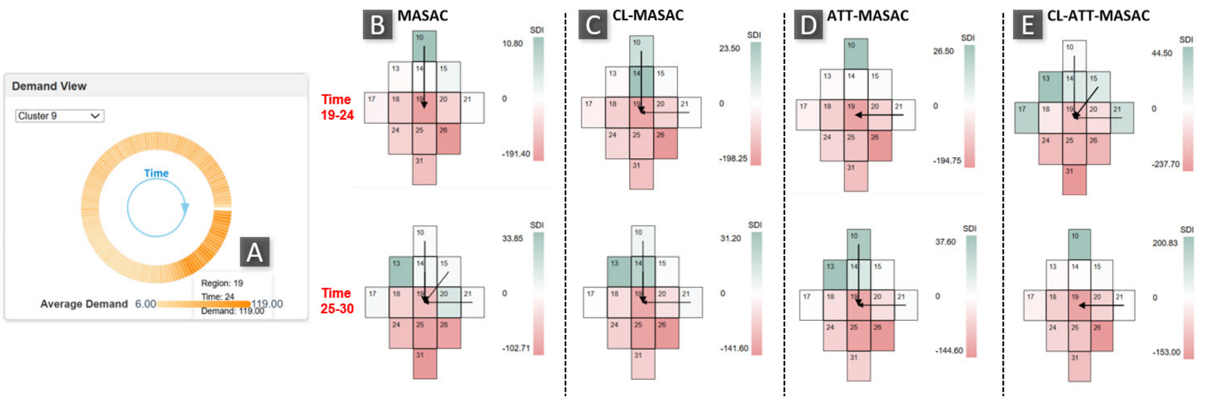


Figure 7: Comparison of dispatch strategies under dynamic demand scenarios. (A) Demand variation in Region 19 during time steps 15–35, with a sharp increase in steps 19–24 and a decline in steps 25–30. (B–E) Vehicle dispatch patterns of MASAC, CL-MASAC, ATT-MASAC, and CL-ATT-MASAC during this period. The results highlight differences in regional coordination, response timeliness, and adaptability, with CL-ATT-MASAC demonstrating the most effective coordination under fluctuating demand.

abling the collection of multifaceted feedback to validate the system's domain applicability and usability.

Evaluation Tasks. Participants used *DpLens* for three core tasks:

(1) Model Training Analysis: Participants were required to describe the training trends of a reinforcement learning model and evaluate its overall performance based on the training metrics (e.g., reward curves, loss functions) and relevant visualizations presented by the system.

(2) Episode Exploration and Agent Behavior Analysis: This task involved participants selecting specific training episodes (e.g., an episode from the later stages of training) to conduct in-depth analyses of agent action sequences, responses to different traffic flow/environmental states, and the relationships or interactions among multiple agents. They were also encouraged to compare different episodes to summarize the evolutionary characteristics of agents throughout the training process.

(3) Decision Process Understanding: Participants needed to choose a specific state during the model's operation and, by utilizing information provided by the system such as feature importance and state feature values, explain the basis and process of the reinforcement learning model's decision-making in that state.

Study Procedure The in-person study began with an introduction to *DpLens*, its objectives, the RL model, visual components, and workflow. A 20-minute system demonstration followed, explaining view functionalities and interaction for data exploration. Participants then had ample time for hands-on familiarization before completing the evaluation tasks. Researchers were available for assistance. Finally, participants completed an anonymous questionnaire with 7-point Likert scale items (1="strongly disagree", 7="strongly agree") to provide systematic feedback.

We conducted significance testing on the questionnaire results. As shown in Table.3, the scores for all Likert scale items were significantly higher than the neutral midpoint, with item means ranging from 6.40 to 6.80 and standard deviations between 0.41 and 0.76. One-sample t-tests indicated that all items yielded t-values greater than 14 and p-values less than 0.001 (e.g., Q3: mean = 6.80, SD = 0.41, t = 30.51, p ¡ 0.001), demonstrating that the ratings for all items were significantly above the neutral level with a high degree of statistical significance.

Results and Discussion Questionnaire results (Figure.8) showed a positive overall perception of *DpLens*, with mean scores significantly above neutral, indicating strengths in design and functionality.

Table 3: Descriptive Statistics and t-test Results

Item	Mean	Std. Dev.	*t*-value	*p*-value
Q1	6.60	0.60	19.44	< 0.001
Q2	6.55	0.60	18.86	< 0.001
Q3	6.80	0.41	30.51	< 0.001
Q4	6.40	0.68	15.77	< 0.001
Q5	6.50	0.69	16.24	< 0.001
Q6	6.60	0.50	23.13	< 0.001
Q7	6.50	0.76	14.69	< 0.001
Q8	6.65	0.49	24.22	< 0.001
Q9	6.60	0.68	17.09	< 0.001
Q10	6.60	0.60	19.44	< 0.001
Q11	6.75	0.44	27.68	< 0.001
Q12	6.65	0.49	24.22	< 0.001
Q13	6.65	0.59	20.18	< 0.001

Completeness & Usability Interaction: For Q1, participants showed strong consensus, with 13 out of 20 giving the highest score of 7. Q2 and Q3 also received high evaluations, with 95% and 100% of participants rating them 6 or 7, respectively. For Q4, 90% of participants provided positive feedback with scores of 6 or above.

Functionality: For Q6, all participants rated the item 6 or 7. Q5 and Q7 received the highest score of 7 from 60% and 65% of participants, respectively, underscoring the system's effectiveness in revealing internal model mechanisms. Q8 and Q9 were also widely endorsed, with 90% of users rating Q9 a 6 or 7.

Visual Design: For Q10, participants expressed strong agreement, with 95% assigning scores of 6 or 7. Q11 and Q12 were highly praised, with 100% of participants rating them 6 or higher, indicating that visual elements played a supportive role in enhancing understanding and exploration. After reevaluation from a visual design perspective, Q13 again received strong approval, with 14 participants assigning the highest score of 7.

In summary, the study affirmed *DpLens*'s capability in

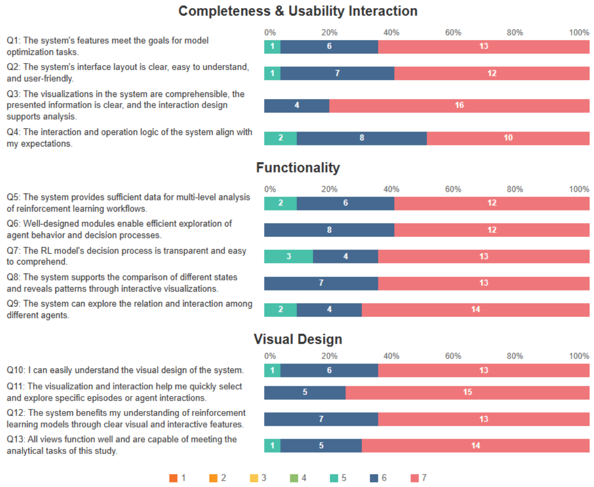


Figure 8: User feedback on *DpLens* from the questionnaire (N=20). The chart displays 7-point Likert scale responses (1=strongly disagree, 7=strongly agree) to 13 questions covering the system's Completeness & Usability Interaction, Functionality, and Visual Design.

assisting users with RL model understanding, analysis, and evaluation. Participants found it functionally complete, interactively fluid, and visually intuitive.

7 Discussion and Future Work

This section discusses the broader applicability and scalability of *DpLens*, acknowledges its current limitations, and outlines promising avenues for future research.

Generalization. The core visual analytics framework of *DpLens* is designed to be adaptable beyond the current taxi dispatching task. Its analytical principles—particularly for dissecting reinforcement learning (RL) training dynamics, regional decision behaviors, and agent interactions—can generalize to other multi-agent RL (MARL) applications such as logistics, mobility-on-demand, and urban resource allocation. Furthermore, the visualization pipeline is algorithm-agnostic, allowing potential adaptation to a range of RL paradigms (value-based, policy-based, actor-critic) and domains such as robotics, game AI, or finance. By providing interpretable and interactive representations, *DpLens* supports broader RL literacy for both novices and domain experts.

Scalability. The proposed method exhibits good scalability and transferability. Although the current experiments are conducted based on the New York taxi dataset, the core of the method lies in travel pattern driven regional clustering and cross region dispatch optimization. This allows the framework to be transferred to other transportation scenarios such as ride hailing dispatch in cities like Chengdu and Beijing without modifying the overall structure, simply by adapting to localized travel data. Meanwhile, the CLATTMASAC framework supports shared policy networks and integrates a multihead attention mechanism to enable information exchange among agents. As a result, it can be applied to traffic dispatch scenarios with varying scales, numbers of agents, and agent types, demonstrating strong scalability.

Limitations and Future Work. *DpLens* currently focuses on post-hoc analysis and human-in-the-loop policy refinement, without supporting real-time training monitoring or integration with deployment pipelines. In this study, region-level taxi dispatch is approximated using grid-based modeling, where vehicle movement costs are estimated via Manhattan distances rather than real road networks and dynamic traffic conditions. While this improves trainability and computational efficiency, it may limit realism and applicability in complex urban settings. Future work includes: (1) Incorporating real-world road topology and dynamic traffic states to better estimate dispatch costs based on actual OD distances and travel times; (2) Supporting real-time inspection of training progress and anomalies; (3) Enabling comparative analysis across multiple model variants or training runs; and (4) Integrating with mainstream RL libraries and MLOps tools to enhance reproducibility and real-world deployment.

8 Conclusion

In this work, we proposes a taxi dispatch framework based on MARL, integrated with the visual analytics system DpLens to enhance the effectiveness and interpretability of dispatch strategies. By modeling urban regions as agents with heterogeneous action spaces, the framework enables joint optimization of vehicle relocation behaviors across regions, significantly improving order matching rates while reducing the number of vehicle relocations. The DpLens system supports multi-view policy analysis and critical state identification, allowing users to explore the evolution of dispatch strategies and inter-agent collaboration mechanisms from both macro and micro perspectives. Case studies and user evaluations demonstrate that the approach performs excellently in terms of model interpretability, policy optimization, and system robustness. Future work will focus on extending real-time monitoring capabilities, strengthening model optimization support, and further validating the system's long-term practicality.

Acknowledgments

This work was supported by National Natural Science Foundation of China under Grant 62377008 and 42171450.

References

[1] A. Aalipour and A. Khani. Modeling, analysis, and control of autonomous mobility-on-demand systems: A discrete-time linear dynamical system and a model predictive control approach. *IEEE Transactions on Intelligent Transportation Systems*, 25(8):8615–8628, 2024. doi: 10.1109/TITS.2024.3398562 2

[2] A. O. Al-Abbasi, A. Ghosh, and V. Aggarwal. Deeppool: Distributed model-free algorithm for ride-sharing using deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 20(12):4714–4727, 2019. doi: 10.1109/TITS.2019.2931830 3

[3] N. Alisoltani, M. Zargayouna, and M. Ameli. Optimising ride-sharing efficiency: innovative shareability-focused pricing strategies. *Transportmetrica B: Transport Dynamics*, 12(1):2417252, 2024. doi: 10.1080/21680566.2024.2417252 2

[4] O. De Lima, H. Shah, T.-S. Chu, and B. Fogelson. Efficient ridesharing dispatch using multi-agent reinforcement learning. *arXiv preprint arXiv:2006.10897*, abs/2006.10897, 2020. doi: 10.48550/arXiv.2006.10897 3

[5] Y. Deng, H. Chen, S. Shao, J. Tang, J. Pi, and A. Gupta. Multi-objective vehicle rebalancing for ridehailing system using a reinforcement learning approach. *Journal of Management Science and Engineering*, 7(2):346–364, 2022. doi: 10.1016/j.jmse.2021.12.004 3

[6] S. Greydanus, A. Koul, J. Dodge, and A. Fern. Visualizing and understanding atari agents. In *International conference on machine learning*, pp. 1792–1801, 2018. doi: 10.48550/arXiv.1711.00138 3

[7] M. Hu and Y. Zhou. Dynamic type matching. *Manufacturing & Service Operations Management*, 24(1):125–142, 2022. doi: 10.1287/msom.2020.0952 2

[8] M. Hutsebaut-Buysse, K. Mets, and S. Latré. Hierarchical reinforcement learning: A survey and open research challenges.

Machine Learning and Knowledge Extraction, 4(1):172–221, 2022. doi: 10.3390/make4010009 3

[9] T. Jaunet, R. Vuillemot, and C. Wolf. Drlviz: Understanding decisions and memory in deep reinforcement learning. In *Computer Graphics Forum*, pp. 49–61, 2020. doi: 10.1111/cgf.13962 3

[10] Y. Jing, B. Guo, N. Li, Y. Ding, Y. Liu, and Z. Yu. Scalable order dispatching through federated multi-agent deep reinforcement learning. *Expert Systems with Applications*, 264:125792, 2025. doi: 10.1016/j.eswa.2024.125792 2

[11] D.-H. Lee, H. Wang, R. L. Cheu, and S. H. Teo. Taxi dispatch system based on current demands and real-time traffic conditions. *Transportation Research Record*, 1882(1):193–200, 2004. doi: 10.3141/1882-23 2

[12] J. Lee, G.-L. Park, H. Kim, Y.-K. Yang, P. Kim, and S.-W. Kim. A telematics service system based on the linux cluster. In *Computational Science–ICCS 2007: 7th International Conference, Beijing, China, May 27-30, 2007, Proceedings, Part IV 7*, pp. 660–667, 2007. doi: 10.1007/978-3-540-72590-9_96 2

[13] M. Li, K. Cai, and P. Zhao. Optimizing same-day delivery with vehicles and drones: A hierarchical deep reinforcement learning approach. *Transportation Research Part E: Logistics and Transportation Review*, 193:103878, 2025. doi: 10.1016/j.tre.2024.103878 2

[14] P. Li, Q. Fei, and Z. Chen. Interpretable multi-agent reinforcement learning via multi-head variational autoencoders. *Applied Intelligence*, 55(7):577, 2025. doi: 10.1007/s10489-025-06473-7 3

[15] C. Liu, C.-X. Chen, and C. Chen. Meta: A city-wide taxi repositioning framework based on multi-agent reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):13890–13895, 2021. 7

[16] M. Lowalekar, P. Varakantham, and P. Jaillet. Online spatio-temporal matching in stochastic and dynamic domains. *Artificial Intelligence*, 261:71–112, 2018. doi: 10.1016/j.artint.2018.04.005 2

[17] Z. Luo, J. Xu, and F. Chen. Multi-agent reinforcement traffic signal control based on interpretable influence mechanism and biased relu approximation. *arXiv preprint arXiv:2403.13639*, abs/2403.13639, 2024. doi: 10.48550/arXiv.2403.13639 3

[18] G. Ma, W. Wang, B. Sun, W. Wu, and Y. Zhou. Crowd-sourced task dispatching for the shared electric vehicle relocation problem: a hybrid variable neighbourhood search and genetic algorithm. *Transportmetrica B: Transport Dynamics*, 13(1):2490511, 2025. doi: 10.1080/21680566.2025.2490511 2

[19] C. Mi, S. Cheng, and F. Lu. Predicting taxi-calling demands using multi-feature and residual attention graph convolutional long short-term memory networks. *ISPRS International Journal of Geo-Information*, 11(3):185, 2022. doi: 10.3390/ijgi11030185 2

[20] T. M. Moerland, J. Broekens, A. Plaat, C. M. Jonker, et al. Model-based reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 16(1):1–118, 2023. doi: 10.1561/2200000086 3

[21] A. Munawar and M. Piantanakulchai. Machine learning-driven passenger demand forecasting for autonomous taxi transportation systems in smart cities. *Expert Systems*, 42(3):e70014, 2025. doi: 10.1111/exsy.70014 2

[22] N. Peng, Y. Xi, J. Rao, X. Ma, and F. Ren. Urban multiple route planning model using dynamic programming in reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):8037–8047, 2021. doi: 10.1109/TITS.2021.3075221 3

[23] E. Puiutta and E. M. Veith. Explainable reinforcement learning: A survey. In *International cross-domain conference for machine learning and knowledge extraction*, pp. 77–95, 2020. doi: 10.1007/978-3-030-57321-8_5 3

[24] T. M. Rajeh, Z. Luo, M. H. Javed, F. Alhaek, and T. Li. A clustering-based multi-agent reinforcement learning framework for finer-grained taxi dispatching. *IEEE Transactions on Intelligent Transportation Systems*, 25(9):11269–11281, 2024. doi: 10.1109/TITS.2024.3370820 2

[25] X. Shi, J. Zhang, Z. Liang, and D. Seng. Maddpgviz: a visual analytics approach to understand multi-agent deep reinforcement learning. *Journal of Visualization*, 26(5):1189–1205, 2023. doi: 10.1007/s12650-023-00928-0 3

[26] J. Si, F. He, X. Lin, and X. Tang. Vehicle dispatching and routing of on-demand intercity ride-pooling services: A multi-agent hierarchical reinforcement learning approach. *Transportation Research Part E: Logistics and Transportation Review*, 186:103551, 2024. doi: 10.1016/j.tre.2024.103551 2

[27] G. A. Vouros. Explainable deep reinforcement learning: state of the art and challenges. *ACM Computing Surveys*, 55(5):1–39, 2022. doi: 10.1145/352744 3

[28] J. Wang, L. Gou, H.-W. Shen, and H. Yang. Dqnviz: A visual analytics approach to understand deep q-networks. *IEEE transactions on visualization and computer graphics*, 25(1):288–298, 2018. doi: 10.1109/TVCG.2018.2864504 3

[29] J. Wang, W. Zhang, H. Yang, C.-C. M. Yeh, and L. Wang. Visual analytics for rnn-based deep reinforcement learning. *IEEE Transactions on Visualization and Computer Graphics*, 28(12):4141–4155, 2021. doi: 10.1109/TVCG.2021.3076749 3

[30] X. Wang, S. Wang, X. Liang, D. Zhao, J. Huang, X. Xu, B. Dai, and Q. Miao. Deep reinforcement learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(4):5064–5078, 2022. doi: 10.1109/TNNLS.2022.3207346 3

[31] L. Wells and T. Bednarz. Explainable ai and reinforcement learning—a systematic review of current approaches and trends. *Frontiers in artificial intelligence*, 4:550030, 2021. doi: 10.3389/frai.2021.550030 3

[32] Q. Xia, H. Zhang, D. Qu, J. Bai, X. Wang, and W. Gong. Tpmvis: visual analytics of taxi dispatching based on travel pattern mining and demand prediction. *Journal of Visualization*, pp. 1–19, 2025. doi: 10.1007/s12650-025-01064-7 2

[33] Y. Xia, W. Qian, and C. Ji. Research on order allocation strategies for ride-hailing platforms considering passenger order cancellations during order overflow. *Applied Sciences*, 15(6):3243, 2025. doi: 10.3390/app15063243 2

[34] B. Xu, A. Rubenis, and C. Long. Reinforcement learning based smart charging for electric vehicle fleet. In *International Symposium on Intelligent Technology for Future Transportation*, pp. 375–383, 2024. doi: 10.1007/978-3-031-84148-4_28 3

[35] C. Yan, H. Zhu, N. Korolko, and D. Woodard. Dynamic pricing and matching in ride-hailing platforms. *Naval Research Logistics (NRL)*, 67(8):705–724, 2020. doi: 10.1002/nav.21872 2

[36] H. Yu, X. Guo, X. Luo, Z. Wu, and J. Zhao. Rstr: A two-layer taxi repositioning strategy using multi-agent reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 26(3):3619–3628, 2025. doi: 10.1109/TITS.2024.3516944 2

[37] K. Zhang, Z. Yang, and T. Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control*, 325:321–384, 2021. doi: 10.1007/978-3-030-60990-0_12 3

[38] Y. Zhang, G. Zheng, Z. Liu, Q. Li, and H. Zeng. Marlens: understanding multi-agent reinforcement learning for traffic signal control via visual analytics. *IEEE transactions on visualization and computer graphics*, pp. 1–16, 2024. doi: 10.1109/TVCG.2024.3392587 3

[39] B. Zheng, Q. Hu, L. Ming, J. Hu, L. Chen, K. Zheng, and C. S. Jensen. Soup: Spatial-temporal demand forecasting and competitive supply in transportation. *IEEE Transactions on Knowledge and Data Engineering*, 35(2):2034–2047, 2023. doi: 10.1109/TKDE.2021.3110778 2

[40] X. Zhou, L. Wu, Y. Zhang, Z.-S. Chen, and S. Jiang. A robust deep reinforcement learning approach to driverless taxi dispatching under uncertain demand. *Information Sciences*, 646(1):119401, 2023. doi: 10.1016/j.ins.2023.119401 3