HCRide: Harmonizing Passenger Fairness and Driver Preference for Human-Centered Ride-Hailing

Lin Jiang[1], Yu Yang[2], Guang Wang[1]*
[1]Department of Computer Science, Florida State University
[2]Department of Computer Science and Engineering, Lehigh University
lj23d@fsu.edu, yuyang@lehigh.edu, guang@cs.fsu.edu

Abstract

Order dispatch systems play a vital role in ride-hailing services, which directly influence operator revenue, driver profit, and passenger experience. Most existing work focuses on improving system efficiency in terms of operator revenue, which may cause a bad experience for both passengers and drivers. Hence, in this work, we aim to design a human-centered ride-hailing system by considering both passenger fairness and driver preference without compromising the overall system efficiency. However, it is nontrivial to achieve this target due to the potential conflicts between passenger fairness and driver preference since optimizing one may sacrifice the other. To address this challenge, we design HCRide, a **H**uman-**C**entered **Ride**-hailing system based on a novel multi-agent reinforcement learning algorithm called **H**armonization-oriented **A**ctor-**Bi**-**C**ritic (Habic), which includes three major components (i.e., a multi-agent competition mechanism, a dynamic Actor network, and a Bi-Critic network) to optimize system efficiency and passenger fairness with driver preference consideration. We extensively evaluate our HCRide using two real-world ride-hailing datasets from Shenzhen and New York City. Experimental results show our HCRide effectively improves system efficiency by 2.02%, fairness by 5.39%, and driver preference by 10.21% compared to state-of-the-art baselines.

1 Introduction

In recent years, ride-hailing services (e.g., Uber, Lyft, Ola Cabs, and DiDi Chuxing) have become indispensable to our daily transportation needs. By 2023, the global ride-hailing market size was valued at $109.3 billion, and it is expected to expand at a growth rate of 12.70% from 2024 to 2033 [RESEARCH, 2023]. One of the most important components of ride-hailing services is the order dispatch system, which directly impacts the revenue of platforms, the work experience of drivers, and the user experience of passengers.

Due to its importance, order dispatch has attracted significant attention from both industry and academia [Chen *et al.*, 2019; Yuan and Van Hentenryck, 2021; Xu *et al.*, 2018; Wang *et al.*, 2022b]. However, most existing works focus on maximizing system revenue, which can potentially compromise the driver and passenger experience. Although some recent studies consider passenger fairness [Sühr *et al.*, 2019; Zhou *et al.*, 2023; Nanda *et al.*, 2020; Wang *et al.*, 2021; Wang *et al.*, 2023a; Jiang *et al.*, 2023], most adopt an *absolute fairness* setting [Wang *et al.*, 2022a], assuming all passengers should experience equal waiting times regardless of location, which overlooks the dynamic nature of supply and demand across regions. Furthermore, driver preferences are frequently ignored, leading to poor experiences when drivers are dispatched to unfamiliar or undesired areas.

Hence, in this work, we aim to design a human-centered ride-hailing order dispatch system that harmonizes passenger fairness and driver preference. However, it is nontrivial to achieve this due to the following two reasons. (i) It is challenging to formally define passenger fairness and driver preference since they have highly spatial and temporal dynamics. (ii) Harmonizing passenger fairness and driver preference is also challenging due to their potential conflicts since improving passenger fairness may not align with drivers' preferences. For example, to ensure passenger fairness, drivers might be dispatched to high-demand areas, which could be distant from their preferred locations or lead to extended working hours beyond their preferred schedules.

To address the above challenges, we propose a **H**uman-**C**entered **Ride**-hailing framework, called **HCRide**, which aims to minimize total passenger waiting time and enhance fairness without compromising driver preferences. In HCRide, we formulate the order dispatch problem as a Constrained Markov Decision Process (CMDP), where the passenger fairness-aware reward serves as the optimization objective and the accumulated driver preference-based cost is treated as a constraint. Passenger fairness is formally defined based on the divergence of waiting times across different spatial-temporal contexts, considering both inter-region and intra-region levels. Driver preferences are modeled using each driver's historical visitation frequency to various regions, reflecting their working habits and regional familiarity. To solve this CMDP, we develop a novel multi-agent reinforcement learning (RL) algorithm called **H**armonization-oriented **A**ctor-**Bi**-**C**ritic (Habic). Habic consists of three key components. First, a multi-agent competition mechanism transforms the large joint action space into smaller distributed action spaces among a limited number of candidate agents. This enables a micro-level decision process [Qin *et al.*, 2022]

*Corresponding author

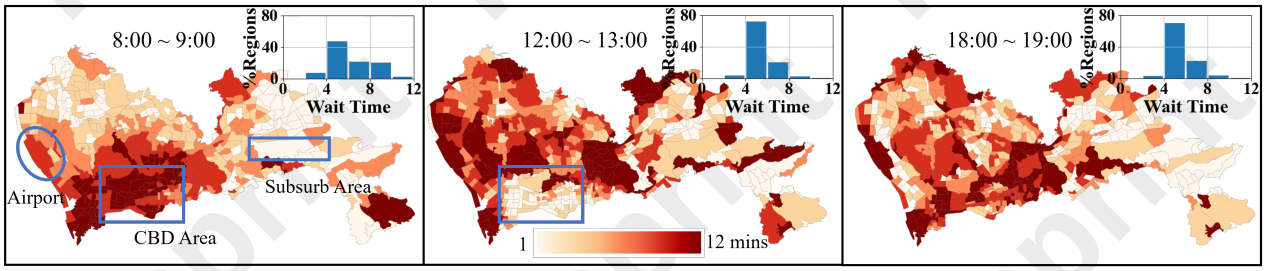

Figure 1: Average Passenger Waiting Time in Different Regions at Different Periods

that helps manage competition among proximate heterogeneous drivers with varying preferences. Second, a Bi-Critic module incorporates two evaluation networks: one estimates the reward value reflecting system efficiency and passenger fairness, while the other assesses the cost related to driver preferences. Third, an Actor module utilizes the outputs of the Bi-Critic networks to generate dispatch decisions that balance both passenger fairness and driver preferences.

The key contributions of this work are as follows:

1. To our knowledge, this is the first study on human-centered ride-hailing order dispatch that considers both passenger fairness and driver preference. Our design is motivated by social studies and data-driven analysis, from which we observe: (i) a notable discrepancy in waiting times among passengers, both within and across regions; (ii) drivers show distinct preferences for operational regions—some favor smaller, confined areas, while others are willing to cover broader locations.
2. Based on the data-driven findings, we design a human-centered ride-hailing order dispatch system called HCRide to improve passenger fairness without compromising driver preferences. Spatio-temporal-aware fairness and preference are defined. The core of HCRide is a novel multi-agent RL algorithm called Habic, which includes a multi-agent competition mechanism and an Actor-Bi-Critic module to harmonize passenger fairness and driver preference.
3. More importantly, we implement and extensively evaluate our HCRide based on two real-world ride-hailing datasets. Experiment results show our HCRide effectively improves system efficiency by 1.77% and 2.02%, inter-region fairness by 5.29% and 5.28%, intra-region fairness by 7.65% and 5.39%, and driver preference by 7.77% and 10.21% compared to baselines on the Shenzhen and NYC datasets, respectively. To verify our work, we have the code available at GitHub [1].

2 Socially Informed Fairness and Preference Formulation

2.1 Data-driven Findings

In our previous project, we conducted qualitative studies to understand people's perceptions of current ride-hailing services, focusing on their views on fairness and their personal preferences [Wang *et al.*, 2025]. In this paper, we also utilize real-world data with over one million ride-hailing orders to verify findings from a quantitative perspective, and finally, we have the following conclusions:

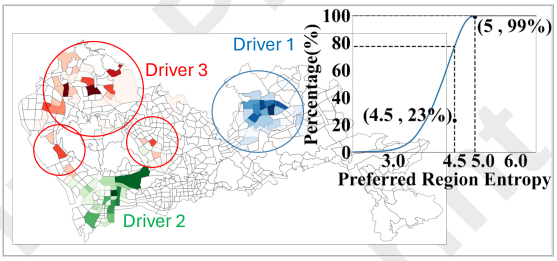

Figure 2: A Visualization of Driver Preference

1. There is strong demand among passengers for equitable waiting times, particularly relative to others in close spatial-temporal proximity. However, substantial disparities in waiting times exist both within and across regions, driven by various spatial-temporal factors. As shown in Fig. 1, we visualize average passenger waiting times across 491 regions in Shenzhen during three periods: morning and evening rush hour, and noon non-rush hour. The results show that (i) **spatially**, significant differences exist between regions, such as between the Central Business District (CBD) and suburban areas; and (ii) **temporally**, even within the same region, average waiting times vary considerably within and across time periods.

2. Drivers have also reported having individual preferences for operating in specific areas at different times, such as locations near their homes, airports, or downtown districts. However, they are often assigned orders outside these preferred regions or in unfamiliar areas, which can negatively affect their satisfaction and operational efficiency. As shown in Fig. 2, darker colors represent areas frequently visited by a given driver. We observe that some drivers (e.g., Driver 1) tend to operate within limited regions, while others (e.g., Driver 3) cover a broader range. The preferred region entropy analysis in the upper right of Fig. 2 further highlights the diversity in drivers' operational areas.

2.2 Design of Fairness and Preference

Passenger Fairness

Motivated by the above findings, we define passenger fairness from both inter-region and intra-region perspectives, incorpo-

[1] https://github.com/LinJiang18/HCRide

rating spatial and temporal patterns. In particular, intra-region fairness is defined as:

$$WT(p_1 \mid u, v) = WT(p_2 \mid u, v) \tag{1}$$

where $WT(\cdot)$ denotes the passenger waiting time, and p_1 and p_2 are two passengers located in the same region u during a specific time period v (e.g., 8:00–9:00). In this study, each time period is set to one hour, implying that passengers within the same region and period should experience comparable waiting times. Depending on the application, the temporal granularity can be adjusted. Under Eq. 1, fairness is achieved when the waiting times for both passengers are equal.

For passengers in different regions, we define a fairness benchmark, denoted as $WT_c(u, v)$, representing the expected average waiting time in region u during time period v. A passenger's waiting time in region u is considered fair if it aligns closely with this benchmark $WT_c(u, v)$. Inspired by the concept of Demographic Parity [Singh and Joachims, 2018], we evaluate fairness across regions by comparing their respective fairness benchmarks as follows:

$$\frac{WT_c(u_1, v)}{|C_1|} = \frac{WT_c(u_2, v)}{|C_2|} \tag{2}$$

where $|C_i| = \beta_i \times \frac{N^i_{\text{passenger}}}{N^i_{\text{driver}}}$; $N_{\text{passenger}}$ is the historical average number of passengers in region u during period v, N_{driver} denotes the number of drivers who prefer to operate in region u, and β_i is an adjustable hyperparameter. Under this setting, the inter-region fairness benchmark $WT_c(u, v)$ is inversely proportional to the supply-demand ratio. The underlying **rationale** is that the expected waiting time in a region should decrease when driver supply exceeds passenger demand.

Driver Preference

To quantify driver preferences, we introduce the following definitions: (i) $\mathcal{U}$: the set of all regions in the city; (ii) $\mathcal{H}^+_k$: the set of regions that provide positive feedback when driver k serves a passenger there; (iii) $\mathcal{H}^0_k$: the set of regions that are neutral—i.e., do not yield positive feedback but are still acceptable to driver k; (iv) $\mathcal{H}^-_k$: the set of regions that result in negative feedback for driver k. The formal representations of these sets are defined as follows:

$$\mathcal{H}^+_k = \{u \in \mathcal{U} \mid V_k(u) > d\} \tag{3}$$

$$\mathcal{H}^0_k = \{u \in \mathcal{U} \mid dis(u, u_1) < \kappa V_k(u_1) \quad \forall u_1 \in \mathcal{H}^+_k\} \tag{4}$$

$$\mathcal{H}^-_k = \{u \in \mathcal{U} \mid u \notin \mathcal{H}^+_k \cup \mathcal{H}^0_k\} \tag{5}$$

where $V_k(u)$ denotes the historical visitation frequency of driver k to region u. A region u is classified into the Positive region set $\mathcal{H}^+_k$ for driver k if the visitation frequency $V_k(u)$ exceeds a threshold d. $dis(u, u_1)$ represents the distance between the region u and u_1, which should be shorter than $\kappa V_k(u_1)$. Here, $\kappa V_k(u_1)$ defines the radius of influence for a positive region u_1. This means that if region u falls within the influence radius of the positive region u_1, it is deemed acceptable for driver k to operate in.

In Fig. 2, we illustrate examples of the influence radius for drivers 1 and 3. The negative region set $\mathcal{H}^-_k$ comprises all other regions that are not included in the positive region set $\mathcal{H}^+_k$ or the neutral region set $\mathcal{H}^0_k$.

3 HCRide System Design

In this part, we introduce the detailed design of the human-centered ride-hailing order dispatch system HCRide, which prioritizes passenger fairness and accommodates driver preferences, considering various spatio-temporal factors.

3.1 Order Dispatch Problem Formulation

Formally, we model the fairness-oriented, preference-aware ride-hailing order dispatch problem as a Constrained Markov Decision Process (CMDP) $\mathcal{G}$, defined as an 8-tuple: $\mathcal{G} = \{\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \mathcal{C}, \mu, \gamma_r, \gamma_c\}$, where $\mathcal{S}$ is the state space; $\mathcal{A}$ the action space; $\mathcal{P}$ the transition probability function, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$; $\mathcal{R}$ the reward function; $\mathcal{C}$ the set of cost functions; $\mu : \mathcal{S} \to [0, 1]$ the initial state distribution; and $\gamma_r, \gamma_c \in (0, 1]$ are discount factors for future rewards and costs. Let Π denote the set of all stationary policies, and let $\pi_\theta(a \mid s) \in \Pi$ be a policy parameterized by θ that maps state s to a distribution over actions a. We divide each day into consecutive time slots (e.g., one minute per slot) and perform state transitions from time slot t to $t + 1$. Dispatch decisions are executed at the beginning of each slot. The detailed formulation of CMDP $\mathcal{G}$ is presented below.

- **Agent**: In our problem, we define each driver as an agent. An agent is marked as inactive while fulfilling an order, which renders it temporarily unable to accept new orders until the current one is completed. As a result, the number of active agents N_t varies across time slots.
- **State** $\mathcal{S}$: To support feasible order dispatch decisions, we define the state S from three dimensions. The state of agent k at time slot t is defined as $s^k_t = \{\mathrm{ST}^k_t, \mathrm{DV}^k_t, \mathrm{CON}^k_t\}$, where ST^k_t is the spatial-temporal state, including the current region r, time period p, and coordinates. DV^k_t represents the driver state, which encodes the driver's preferences. CON^k_t is the context state, capturing global supply-demand conditions, weather, and traffic information.
- **Action** $\mathcal{A}$: Agents in our system can perform one of the three action types: accepting an order a_r, moving to complete an order a_m, or cruising a_c (when the vehicle is unoccupied) based on the driver's preferences.
- **Reward** $\mathcal{R}$: The reward R^i for order i includes both passenger waiting time and fairness, denoted as:

$$\begin{aligned} r^i_t = &-(1-\alpha) WT(i|u, v) \\ &- \alpha \left(\frac{1}{K_{u,v}} \sum_{k=1}^{K_{u,v}} (WT(k|u, v) - WT_c(u, v))^2 \right) \end{aligned} \tag{6}$$

The reward function in our system consists of two components. The first term, $WT(i \mid u, v)$, reflects system efficiency by representing the waiting time of order i in region u during period v. Shorter waiting times yield higher rewards. The second term, $\frac{1}{K_{u,v}} \sum_{k=1}^{K_{u,v}} (WT(k \mid u, v) - WT_c(u, v))^2$, serves as a fairness regularization term. Here, $K_{u,v}$ denotes the total number of passengers

in region u during period v. This term captures the variance between actual waiting times $WT(k \mid u, v)$ and the dynamic fairness benchmark $WT_c(u, v)$, encouraging equitable service across different contexts. To balance efficiency and fairness, we introduce a hyperparameter α to modulate the weight between these two components. In this framework, only the agent fulfilling order i receives the reward r_t^i.

- **Cost** $\mathcal{C}$: In our setting, each agent k will be assigned a positive region set $\mathcal{H}_k^+$, a neutral region set $\mathcal{H}_k^0$, and a negative region set $\mathcal{H}_k^-$ based on its historical operating locations (i.e., preference). Drawing from the negativity effect principle [Rozin and Royzman, 2001], negative experiences often exert a stronger influence on our psychological state than equally significant positive or neutral ones. Hence, we impose a cost c_t^i on agent k when an order dispatch leads it to a destination within its negative region set $\mathcal{H}_k^-$. The magnitude of this cost c_t^i is determined by the distance between the order's destination and the nearest preferred location within $\mathcal{H}_k^+$.

The goal of the defined CMDP $\mathcal{G}$ is to optimize the long-term cumulative reward $J_r(\pi_\theta)$ while ensuring that the cumulative cost $J_c(\pi_\theta)$ remains below a predetermined threshold ξ. Given the centralized nature of the order dispatch system overseeing all agents, we adopt a strategy of centralized training with decentralized execution [Sharma *et al.*, 2021] to reduce computational complexity. This strategy emphasizes the cumulative reward and cost across all agents, rather than focusing on the outcomes of individual agents. Therefore, our long-term cumulative reward $J_r(\pi_\theta)$ and cumulative cost $J_c(\pi_\theta)$ can be represented as:

$$J_r(\pi_\theta) = \underset{\tau\sim\pi_\theta}{E} \sum_{t=0}^{T} \sum_{i=1}^{O_t} \left[(\gamma_r)^t r_t^i\right] \tag{7}$$

$$J_c(\pi_\theta) = \underset{\tau\sim\pi_\theta}{E} \sum_{t=0}^{T} \sum_{i=1}^{O_t} \left[(\gamma_c)^t c_t^i\right] \tag{8}$$

where T denotes the total number of time slots within one episode, and O_t represents the total number of orders in time slot t. During each slot, we dispatch order i to an agent according to the strategy π_θ, and the agent receives a corresponding reward r_t^i and cost c_t^i.

Our objective of maximizing the cumulative reward $J_r(\pi_\theta)$ over an episode and ensuring that the cumulative cost $J_c(\pi_\theta)$ does not exceed a predetermined value ξ can be denoted as Eq. 9. It indicates we aim to enhance system efficiency (i.e., reduce the total waiting time of all passengers) and improve passenger fairness without disproportionately compromising driver preferences.

$$\max_{\pi_\theta \in \Pi} J_r(\pi_\theta) \quad s.t. J_c(\pi_\theta) \leq \xi \tag{9}$$

3.2 Optimization Objective Conversion

Directly solving the constrained problem in an MDP is challenging, so we further convert Eq. 9 into the Lagrangian form:

$$L(\theta, \lambda) = J_r(\theta) - \lambda(J_c(\theta) - \xi) \tag{10}$$

$$\max_{\theta} \min_{\lambda} L(\theta, \lambda) \tag{11}$$

where the $\lambda \in \mathbb{R}^+$ is the Lagrange multiplier, which is a positive real number. The objective of the above Eq. 11 aims to find the global optimal saddle point (θ^*, λ^*). Since θ^* is the optimal value, the θ^* should satisfy $L(\theta^*, \lambda^*) \geq L(\theta, \lambda^*)$, $\forall \theta \in \mathbb{R}$. Similarly, λ^* should satisfy $L(\theta^*, \lambda^*) \leq L(\theta^*, \lambda)$, $\forall \lambda \in \mathbb{R}^+$. Finally, $\forall(\theta, \lambda)$, we obtain:

$$L(\theta^*, \lambda) \geq L(\theta^*, \lambda^*) \geq L(\theta, \lambda^*) \tag{12}$$

However, optimizing the two parameters simultaneously is computationally intractable, especially for θ that is described by a deep neural network. Therefore, we alternatively optimize the two parameters by fixing one and updating the other until convergence. We obtain the final θ^* and λ^* when both of them satisfy that:

$$H = \{(\theta^*, \lambda^*) | \, ||\theta^* - \theta^{*-}|| \leq \epsilon_1, ||\lambda^* - \lambda^{*-}|| \leq \epsilon_2\} \tag{13}$$

Where θ^{*-} and λ^{*-} are the values of the previous values before achieving convergence. In the next part, we will show how we solve this optimization problem with MARL.

3.3 Harmonization-oriented Actor-Bi-Critic

In this section, we design a new MARL algorithm called Habic (i.e., **H**armonization-oriented **A**ctor-**Bi**-**C**ritic) to solve the above-defined problem. There are three key components in the Habic: (i) A multi-agent competition mechanism, which is designed to provide information for decision-making by generating matching features between orders and drivers. (ii) A dynamic Actor network, which is designed to alternately update the policy parameter θ and the Lagrange parameter λ based on the matching features to make decisions in the multi-agent environment. (iii) A Bi-Critic network, which is utilized to evaluate the values of accumulated reward $J_r(\pi_\theta)$ and accumulated cost value $J_c(\pi_\theta)$ simultaneously. An overall framework of Habic is shown in Fig. 3.

Multi-agent Competition Mechanism

We consider drivers within a certain range of an order to compete for it. The driver selected by the Actor will accept the order, and other drivers will keep their original actions, i.e., staying or cruising. Since the number of drivers around an order is dynamic, the candidate agent set and action space are also dynamic. Each agent in the set generates a matching feature $m_i^k = \{s_t^k, o_t^i\}$ to compete for the order, which includes the state s_t^k of agent k and the state o_t^i of order i. The details of $s_t^k = \{ST_t^k, DV_t^k, CON_t^k\}$ can be seen in Sec. 3.1, and $o_t^i = \{OR_i, DE_i, u, v\}$ describes order i's information, including the pickup location OR_i and drop-off location DE_i (represented by longitude and latitude), region u, and period v. As shown in Fig. 3, n matching features are fed into the Actor network to help make dispatch decisions.

Dynamic Actor Network for Decision

In this part, we will first introduce the decision process in Actor, and then show how we update the parameter θ for the policy function $\pi_\theta(a|s)$ and λ for the Lagrange multiplier. We regard the Actor to be equivalent to the policy function $\pi_\theta(a|s)$ to make order dispatch decisions. Assuming there are n candidate agents in the agent set and the matching feature

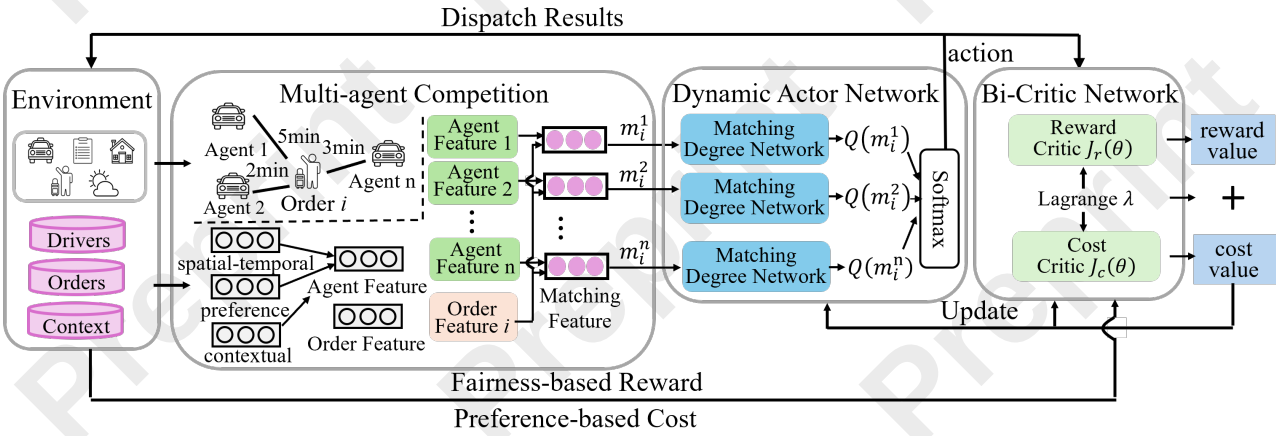

Figure 3: The Overall Framework of the Proposed Habic Method

set is $M = \{m_i^1, m_i^2, \cdots, m_i^n\}$, the probability of choosing the kth agent can be represented as:

$$\pi_\theta(a_k|s) = \frac{exp(Q(m_i^k))}{\sum_{j=1}^{n} exp(Q(m_i^j))} \tag{14}$$

Where $Q(m_i^k)$ is the matching degree neural network to calculate the matching degree between the order i and the agent k. $\pi_\theta(a^k|s)$ means the probability of selecting agent k from the candidate agent set. After calculating the probability of selecting each agent, we can obtain the policy function $\pi_\theta(a|s)$ for the order dispatch decisions.

After executing our order dispatch strategy based on the $\pi_\theta(a|s)$, we can collect the a set of transitions $((s_t^1, \cdots, s_t^{n_t}), a_t^k, r_t^k, c_t^k, s_{t+1}^k)$ and use them as the training data. Based on the gradient search procedure, we can obtain the updating rules for θ and λ as follows:

$$\begin{aligned}\theta_{n+1} &= \theta_n - \eta_\theta \nabla_{\theta_n}(-L(\theta_n, \lambda_n)) \\ &= \theta_n + \eta_\theta[\nabla_{\theta_n} J^R(\pi_\theta) - \lambda_n \nabla_{\theta_n} J^C(\pi_\theta)] \qquad (15) \\ \lambda_{n+1} &= max\,(0, \lambda_n + \eta_\lambda \nabla_{\lambda_n}(-L(\theta_n, \lambda_n))) \\ &= max\,((0, \lambda_n - \eta_\lambda \nabla_{\lambda_n}(J^C(\pi_\theta) - d))) \qquad (16)\end{aligned}$$

Where η_θ and η_λ represent the update step sizes for parameters θ and λ. As we described in Sec. 3.2, the parameters θ and λ will be updated alternately until they reach the optimal values θ^* and λ^*. However, in Eq. 15 and Eq. 16, the values of $J^R(\pi_\theta)$ and $J^C(\pi_\theta)$ still remain unknown. Therefore, we further design the Bi-Critic network to evaluate their values.

Bi-Critic Network for Evaluation

This section introduces how we evaluate the value of $J^R(\pi_\theta)$ and $J^C(\pi_\theta)$. As shown in Eq. 7 and Eq. 8, the $J^R(\pi_\theta)$ and $J^C(\pi_\theta)$ share the same structure, so we will only show the evaluation process for $J^R(\pi_\theta)$, and the evaluation process of $J^C(\pi_\theta)$ is the same. According to [Schulman *et al.*, 2017], $J^R(\pi_\theta)$ can be rewritten as:

$$J^R(\pi_\theta) = E_{s\sim D^\pi(s)} E_{a\sim\pi_{\theta^-}} \left[\frac{\pi_\theta(a|s)}{\pi_{\theta^-}(a|s)} A^R_{\pi_\theta}(s,a)\right] \tag{17}$$

Where $D^\pi(s)$ is the state visitation distribution, which can be described as the average probability of the state s appearing at each moment in the trajectory. π_{θ^-} is the old strategy in the last cycle and π_θ is the new strategy waiting to be updated in this cycle, which means π_{θ^-} and π_θ are equivalent to π_{θ_n} and $\pi_{\theta_{n+1}}$ in Eq. 15. $A^R_{\pi_\theta}(s,a)$ is the Advantage function, which can be considered as another version of Q-value with lower variance by taking the state-value off as the baseline. In Habic, we calculate $A^R_{\pi_\theta}(s,a)$ by utilizing the Generalized Advantage Estimation (GAE) method [Schulman *et al.*, 2015], which can be described as:

$$A^R_{\pi_\theta}(s_t, a_t) = \sum_{l=0}^{\infty} (\gamma_r \psi)^l \left(r_t + \gamma_r V^R_{\pi_\theta}(s_t) - \gamma_r V^R_{\pi_\theta}(s_{t+1})\right) \tag{18}$$

Where $\psi \in [0,1]$ is a hyper-parameter in GAE, and $V^R_{\pi_\theta}(s_t)$ is the value function to describe the value of state s_t when following a policy π_θ.

In Eq. 17, we adopt the off-policy strategy [Brandfonbrener *et al.*, 2021] by using the old strategy π_{θ^-} to collect the data and update the new strategy parameter θ, so the difference between the new strategy and the old strategy will not be too large. We also leverage the KL-divergence [Kullback, 1951] to restrict the updating range of θ, which is shown in Eq. 19:

$$E_{s\sim D^\pi(s)}\left[D_{KL}(\pi_{\theta^-}(\cdot|s), \pi_\theta(\cdot|s))\right] \le \delta \tag{19}$$

By combining Eq. 17 and Eq. 19, we can obtain the final describe of $J^R(\pi_\theta)$ based on the PPO-Clip [Jayant and Bhatnagar, 2022]:

$$\begin{aligned}J^R(\pi_\theta) = E_t \Bigg[min \Bigg(&\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta^-}(a_t|s_t)} A^R_{\pi_\theta}(s_t, a_t), \\ &clip\left(\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta^-}(a_t|s_t)}, 1-\epsilon, 1+\epsilon\right) A^R_{\pi_\theta}(s_t, a_t)\Bigg)\Bigg]\end{aligned} \tag{20}$$

Where $clip(x,a,b) = max(min(x,b),a)$, which means restricting x within the range $[a,b]$. Using the Eq. 20, we can estimate the value of $J^R(\pi_\theta)$ based on the Advantage function $A^R_{\pi_\theta}(s_t,a_t)$, which can be represented by $V^R_{\pi_\theta}(s_t)$ based on Eq. 18. Therefore, we build the Reward Critic network to calculate $V^R_{\pi_\theta}(s_t)$ with the parameter ψ_r. We denote the Reward Critic network as $V^R_{\psi_r}(s_t)$. Similarly, the Cost Critic

network can be denoted as $V^C_{\psi_c}(s_t)$ with the parameter ψ_c. The update rules for parameter ψ_r and ψ_c are:

$$\psi_r \leftarrow \psi_r - \eta_{\psi_r} \nabla \sum [r_t + \gamma_r V^R_{\psi_r}(s_{t+1}) - V^R_{\psi_r}(s_t)] \quad (21)$$

$$\psi_c \leftarrow \psi_c - \eta_{\psi_c} \nabla \sum [r_t + \gamma_c V^C_{\psi_c}(s_{t+1}) - V^C_{\psi_c}(s_t)] \quad (22)$$

To summarize, we build a Reward Critic network $V^R_{\psi_r}(s_t)$ and a Cost Critic network $V^C_{\psi_c}(s_t)$ to estimate $V^R_{\pi_\theta}(s_t)$ and $V^C_{\pi_\theta}(s_t)$, respectively. By updating ψ_r and ψ_c using Eq. 21 and Eq. 22, we obtain the accumulated estimates. The values of $V^R_{\pi_\theta}(s_t)$ and $V^C_{\pi_\theta}(s_t)$ are then used to compute the accumulated reward $J^R(\pi_\theta)$ and cost $J^C(\pi_\theta)$ based on Eq. 18 and Eq. 20. After obtaining these estimates, we use them to update the parameters θ and λ, where the converged θ^* serves as the optimal solution to our optimization objective in Eq. 9, improving passenger fairness without disproportionately compromising driver preferences.

4 Evaluation

4.1 Evaluation Methodology

Data: We evaluate our HCRide on two real-world ride-hailing datasets from the Chinese City Shenzhen and New York City (NYC). The Shenzhen dataset includes 1.07 million orders served by 1,200 ride-hailing vehicles from 03/2021 to 06/2021. The NYC dataset includes 214k orders served by 800 ride-hailing vehicles from 01/2024 to 02/2024.
Baselines: We compare our HCRide with five different categories of baselines: (1) Myopic dispatching method: MD [Zhang *et al.*, 2017]. This method aims to minimize the total waiting time for all the passengers in one slot without future consideration. The method MD will be considered the benchmark to be compared with all other methods in Table 1. (2) Single-agent RL methods: DQND [Mnih *et al.*, 2013], AC-bgm [Wang *et al.*, 2023b]. (3) Multi-agent RL methods: IPPO [De Witt *et al.*, 2020], MAPPO [Yu *et al.*, 2022]. (4) Constrained RL methods: CPO [Achiam *et al.*, 2017], Lag-TRPO [Ray *et al.*, 2019]. Compared to the previous two types of methods, the constrained RL methods introduce the cost and constraint. (5) Variants of our HCRide considering different fairness definitions: HCRide-AF with absolute fairness [Sühr *et al.*, 2019], HCRide-MMF with max-min fairness [Sun *et al.*, 2022].
Metrics: We define three categories of metrics to evaluate the performance of system efficiency (Average Passenger Waiting Time $APWT$), passenger fairness (inter-region fairness PF_{inter} and intra-region fairness PF_{intra} based on the variance of passenger waiting time), and driver preference (Preference Violation Rate PVR), which evaluates the proportion of orders assigned to non-preferred regions of drivers, i.e., the negative region set H_k^-.

4.2 Overall Performance

As shown in Table 1, we compare our HCRide with all eight baselines on the two datasets.

System Efficiency
We evaluate the system efficiency using the Average Passenger Waiting Time $APWT$. As shown in Table 1, our HCRide outperforms all other baselines. Using the Shenzhen dataset as an example, our HCRide reduces the average passenger waiting time by 5.48% compared to the benchmark baseline MD and outperforms the state-of-the-art method Lag-TRPO by $1.77\% = (5.48\% - 3.78\%)/(100\% - 3.78\%)$ in the whole day and 1.53% in the morning rush hour. Compared to Lag-TRPO, HCRide utilizes a more efficient updating method PPO-clip [Jayant and Bhatnagar, 2022], thereby achieving better convergence. HCRide-AF and HCRide-MMF are variants of HCRide with different fairness settings. Since there are no changes to the efficiency settings, their efficiency performances are similar. In particular, single-agent methods DQND and AC-bgm can achieve better performance than multi-agent methods IPPO and MAPPO. One possible reason is that we use the discrete extension for the single-agent RL methods, which incorporates our multi-agent competition mechanism. This allows these methods to learn the value of order-passenger pair from every discrete order dispatch behavior, providing abundant training transactions to guide learning the matching degree between orders and passengers. However, multi-agent RL methods like IPPO and MAPPO can learn from the operation trajectories of each driver. When there are a large number of agents (e.g., over 1,000), it will bring a high variance for training. Additionally, it is challenging for the Actor to learn the competition among different drivers since they are trained independently.

Passenger Fairness
We evaluate passenger fairness from both inter-region and intra-region levels with metrics DPF_{inter} and DPF_{intra}. As shown in Table 1, our HCRide notably improves both metrics. For single-agent RL algorithms such as AC-bgm and constrained RL algorithms like Lag-TRPO, although they utilize the same fairness-based reward function as HCRide, they are less effective due to poor exploration and update ability. We also compare the performance of our spatio-temporal-aware fairness definition with two other widely used fairness definitions: absolute fairness [Zhou and Sethu, 2002] and max-min fairness [Sun *et al.*, 2022]. The results show that our HCRide can achieve better fairness performance compared to HCRide-AF and HCRide-MMF. A possible reason is that our spatio-temporal-aware fairness definition focuses on more fine-grained local information across different spatio-temporal contexts. We also provide the visualization results for inter-region fairness in Fig. 4, which shows the distribution of the average passenger waiting times of all regions on each day during the training process. We find that the variance of average passenger waiting time between different regions decreases during the training process, and the system eventually converges to be close to the fairness benchmark.

Driver Preference
For driver preference, we focus on the percentage of dispatched orders that differ from driver preferences. Using the Shenzhen dataset as an example, the driver preference violation rate reaches 18.21% for the benchmark MD. In this experiment, we set a predetermined violation rate of 15% and expect it to decrease during the training process, eventually converging below this value. The predetermined violation rate can also be set to other values based on operators' goals.

Cities	Shenzhen				NYC			
Methods	Efficiency	Fairness		Preference	Efficiency	Fairness		Preference
	$DAPWT$	DPF_{inter}	DPF_{intra}	$DPVR$	$DAPWT$	DPF_{inter}	DPF_{intra}	$DPVR$
DQND	2.51±0.34	5.26±1.75	6.05±1.57	3.33±3.56	3.43±0.87	7.81±2.23	9.92±2.41	4.25±3.29
AC-bgm	4.03±0.85	6.92±1.17	7.73±1.34	-2.25±3.31	4.77±1.52	10.21±2.37	9.54±2.11	2.53±3.11
IPPO	-0.99±0.78	2.34±1.03	2.55±1.49	3.51±3.25	-3.01±1.94	4.41±3.05	5.87±2.08	-0.68±2.34
MAPPO	1.11±0.94	3.47±1.94	4.12±0.98	3.11±3.37	2.26±1.99	8.28±2.03	7.92±2.56	-1.15±3.48
CPO	4.51±1.12	7.73±1.26	8.52±1.94	7.75±2.95	4.72±1.03	9.88±1.88	12.73±2.47	12.19±5.38
Lag-TRPO	3.78±1.32	8.55±1.46	9.77±1.54	12.74±3.29	5.24±2.17	10.38±2.69	15.42±3.02	15.81±4.42
HCRide-AF	5.31±1.45	7.00±1.82	7.25±1.69	19.23±3.68	7.01±1.98	10.01±2.77	10.86±2.98	22.45±4.01
HCRide-MMF	5.42±1.17	9.98±1.68	11.36±1.92	**19.79±3.03**	7.12±2.08	13.25±3.01	14.34±3.09	23.85±4.21
HCRide	**5.48±1.13**	**13.39±1.71**	**16.67±1.46**	19.52±3.87	**7.15±2.03**	**15.11±2.88**	**19.98±3.03**	**24.41±3.92**

Table 1: Comparison of different methods in terms of various metrics. The symbol **%** is omitted in the table. $DAPWT$, DPF_{inter}, DPF_{intra} and $DPVR$ represent the Decreased ratios in Average Passenger Waiting Time $APWT$, inter-region Passenger Fairness PF_{inter}, intra-region Passenger Fairness PF_{intra} and Preference Violation Rate PVR, respectively. Lower values indicate better performance. All values are based on comparison with the benchmark baseline MD [Zhang *et al.*, 2017]. The best performance is in bold.

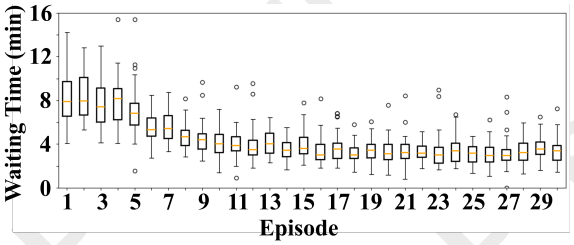

Figure 4: Average Waiting Time Distribution for All Regions

From Table 1, we find that HCRide achieves the best performance on the two datasets. In contrast, baselines such as DQND, AC-bgm, IPPO, and MAPPO, which fail to consider driver performance in order dispatch decisions, show performance similar to the myopic dispatch strategy. Constrained RL baselines such as CPO and Lag-TRPO outperform other non-constrained RL baselines but still fall short compared to our HCRide. A key reason is that our HCRide has a multi-agent competition mechanism to improve the sampling efficiency and a Bi-Critic to provide direct reward value and cost value estimation for the matching degree.

5 Related Work

We divide order dispatch work into efficiency-oriented order dispatch and human-centered order dispatch.

Efficiency-oriented Order Dispatch: Most existing studies prioritize efficiency without considering human factors such as fairness and preference. Xu et al. [Xu *et al.*, 2018] use the DRL to solve sequential dispatch problems by building the global Q-function for orders and passengers. Sadeghi et al. [Sadeghi Eshkevari *et al.*, 2022] propose a scalable RL dispatching algorithm and conduct both offline evaluation and online evaluation. Recently, some works have begun to pay attention to the transferability of algorithm efficiency across various platforms. Wang et al. [Wang *et al.*, 2018] use the transfer learning method to make DRL-based order dispatch algorithms more adaptive in different cities. Wang et al. [Wang *et al.*, 2022b] propose a federated learning algorithm to improve the reliability of dispatching data during cross-platform processes.

Human-centered Order Dispatch: In recent years, human-centered design has attracted much interest, and more and more works focus on fairness. Sühr et al. [Sühr *et al.*, 2019] propose an order dispatch method considering two-sided fairness for both driver and passenger. Lu et al. [Lu *et al.*, 2021] introduce the queueing theory to solve the long waiting time problem for passengers and make a trade-off between efficiency and fairness. There are also some other works focusing on human preference. Carvalho et al. [de Carvalho and Golpayegani, 2022] propose a multi-agent multi-objective optimization approach to satisfy user preferences in ridesharing services. Li et al. [Li *et al.*, 2021] introduce the mutual information-based approach to solve the preference-aware group task assignment in spatial crowdsourcing.

To our knowledge, our HCRide is the first order dispatch system that harmonizes both passenger fairness and driver preference for human-centered ride-hailing services.

6 Conclusion

Motivated by insights from our previous qualitative study and data-driven analysis, in this paper, we design a human-centered ride-hailing order dispatch system called HCRide, which aims to improve both system efficiency and passenger fairness in terms of waiting time without compromising driver preferences. In HCRide, spatio-temporal-aware fairness and preference are formally defined, and we design a novel multi-agent reinforcement learning algorithm called harmonization-oriented Actor-Bi-Critic, which includes a multi-agent competition mechanism, a dynamic Actor network, and a Bi-Critic network to optimize system efficiency and passenger fairness with driver preferences as constraints. Extensive evaluations on two datasets show our HCRide effectively improves system efficiency by 2.02%, inter-region fairness by 5.28%, intra-region fairness by 5.39%, and driver preference by 10.21% compared to baselines.

Acknowledgments

We sincerely thank all anonymous reviewers for their insightful comments and valuable suggestions. This work is partially supported by Florida State University, National Science Foundation under Grant Numbers 2411152, 2427915, and 2318697.

References

[Achiam *et al.*, 2017] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pages 22–31. PMLR, 2017.

[Brandfonbrener *et al.*, 2021] David Brandfonbrener, Will Whitney, Rajesh Ranganath, and Joan Bruna. Offline rl without off-policy evaluation. *Advances in neural information processing systems*, 34:4933–4946, 2021.

[Chen *et al.*, 2019] Mengjing Chen, Weiran Shen, Pingzhong Tang, and Song Zuo. Dispatching through pricing: Modeling ride-sharing and designing dynamic prices. In *IJCAI*, pages 165–171, 2019.

[de Carvalho and Golpayegani, 2022] Vinicius Renan de Carvalho and Fatemeh Golpayegani. Satisfying user preferences in optimised ridesharing services: A multi-agent multi-objective optimisation approach. *Applied Intelligence*, 52(10):11257–11272, 2022.

[De Witt *et al.*, 2020] Christian Schroeder De Witt, Tarun Gupta, Denys Makoviichuk, Viktor Makoviychuk, Philip HS Torr, Mingfei Sun, and Shimon Whiteson. Is independent learning all you need in the starcraft multi-agent challenge? *arXiv preprint arXiv:2011.09533*, 2020.

[Jayant and Bhatnagar, 2022] Ashish K Jayant and Shalabh Bhatnagar. Model-based safe deep reinforcement learning via a constrained proximal policy optimization algorithm. *Advances in Neural Information Processing Systems*, 35:24432–24445, 2022.

[Jiang *et al.*, 2023] Lin Jiang, Shuai Wang, Baoshen Guo, Hai Wang, Desheng Zhang, and Guang Wang. Faircod: A fairness-aware concurrent dispatch system for large-scale instant delivery services. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4229–4238, 2023.

[Kullback, 1951] Solomon Kullback. Kullback-leibler divergence, 1951.

[Li *et al.*, 2021] Yunchuan Li, Yan Zhao, and Kai Zheng. Preference-aware group task assignment in spatial crowdsourcing: A mutual information-based approach. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 350–359. IEEE, 2021.

[Lu *et al.*, 2021] Chenbei Lu, Jiaman Wu, Chenye Wu, Yongli Qin, Qun Li, and Nan Ma. Efficiency or fairness? carpooling design for online ride-hailing platform in transport hubs at midnight. In *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*, pages 244–255, 2021.

[Mnih *et al.*, 2013] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[Nanda *et al.*, 2020] Vedant Nanda, Pan Xu, Karthik Abhinav Sankararaman, John Dickerson, and Aravind Srinivasan. Balancing the tradeoff between profit and fairness in rideshare platforms during high-demand hours. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 2210–2217, 2020.

[Qin *et al.*, 2022] Zhiwei Tony Qin, Hongtu Zhu, and Jieping Ye. Reinforcement learning for ridesharing: An extended survey. *Transportation Research Part C: Emerging Technologies*, 144:103852, 2022.

[Ray *et al.*, 2019] Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 7(1):2, 2019.

[RESEARCH, 2023] GRAND VIEW RESEARCH. Ride-hailing service market. https://www.factmr.com/report/ride-hailing-service-market, 2023.

[Rozin and Royzman, 2001] Paul Rozin and Edward B Royzman. Negativity bias, negativity dominance, and contagion. *Personality and social psychology review*, 5(4):296–320, 2001.

[Sadeghi Eshkevari *et al.*, 2022] Soheil Sadeghi Eshkevari, Xiaocheng Tang, Zhiwei Qin, Jinhan Mei, Cheng Zhang, Qianying Meng, and Jia Xu. Reinforcement learning in the wild: Scalable rl dispatching algorithm deployed in ridehailing marketplace. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3838–3848, 2022.

[Schulman *et al.*, 2015] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

[Schulman *et al.*, 2017] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[Sharma *et al.*, 2021] Piyush K Sharma, Rolando Fernandez, Erin Zaroukian, Michael Dorothy, Anjon Basak, and Derrik E Asher. Survey of recent multi-agent reinforcement learning algorithms utilizing centralized training. In *Artificial intelligence and machine learning for multi-domain operations applications III*, volume 11746, pages 665–676. SPIE, 2021.

[Singh and Joachims, 2018] Ashudeep Singh and Thorsten Joachims. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2219–2228, 2018.

[Sühr *et al.*, 2019] Tom Sühr, Asia J Biega, Meike Zehlike, Krishna P Gummadi, and Abhijnan Chakraborty. Two-

sided fairness for repeated matchings in two-sided markets: A case study of a ride-hailing platform. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3082–3092, 2019.

[Sun *et al.*, 2022] Jiahui Sun, Haiming Jin, Zhaoxing Yang, Lu Su, and Xinbing Wang. Optimizing long-term efficiency and fairness in ride-hailing via joint order dispatching and driver repositioning. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3950–3960, 2022.

[Wang *et al.*, 2018] Zhaodong Wang, Zhiwei Qin, Xiaocheng Tang, Jieping Ye, and Hongtu Zhu. Deep reinforcement learning with knowledge transfer for online rides order dispatching. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 617–626. IEEE, 2018.

[Wang *et al.*, 2021] Guang Wang, Shuxin Zhong, Shuai Wang, Fei Miao, Zheng Dong, and Desheng Zhang. Data-driven fairness-aware vehicle displacement for large-scale electric taxi fleets. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pages 1200–1211. IEEE, 2021.

[Wang *et al.*, 2022a] Xiaomeng Wang, Yishi Zhang, and Ruilin Zhu. A brief review on algorithmic fairness. *Management System Engineering*, 1(1):7, 2022.

[Wang *et al.*, 2022b] Yansheng Wang, Yongxin Tong, Zimu Zhou, Ziyao Ren, Yi Xu, Guobin Wu, and Weifeng Lv. Fed-ltd: Towards cross-platform ride hailing via federated learning to dispatch. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4079–4089, 2022.

[Wang *et al.*, 2023a] Guang Wang, Sihong He, Lin Jiang, Shuai Wang, Fei Miao, Fan Zhang, Zheng Dong, and Desheng Zhang. Fairmove: A data-driven vehicle displacement system for jointly optimizing profit efficiency and fairness of electric for-hire vehicles. *IEEE Transactions on Mobile Computing*, 23(6):6785–6802, 2023.

[Wang *et al.*, 2023b] Shuai Wang, Baoshen Guo, Yi Ding, Guang Wang, Suining He, Desheng Zhang, and Tian He. Time-constrained actor-critic reinforcement learning for concurrent order dispatch in on-demand delivery. *IEEE Transactions on Mobile Computing*, 2023.

[Wang *et al.*, 2025] Guang Wang, Vivek K Singh, and Desheng Zhang. A mixed-methods study of wait time perception and discrepancy in technology-mediated mobility systems. *Proceedings of the ACM on Human-Computer Interaction*, 9(2):1–28, 2025.

[Xu *et al.*, 2018] Zhe Xu, Zhixin Li, Qingwen Guan, Dingshui Zhang, Qiang Li, Junxiao Nan, Chunyang Liu, Wei Bian, and Jieping Ye. Large-scale order dispatch in on-demand ride-hailing platforms: A learning and planning approach. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 905–913, 2018.

[Yu *et al.*, 2022] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of ppo in cooperative multi-agent games. *NeurIPS*, 35:24611–24624, 2022.

[Yuan and Van Hentenryck, 2021] Enpeng Yuan and Pascal Van Hentenryck. Real-time pricing optimization for ride-hailing quality of service. In *30th International Joint Conference on Artificial Intelligence (IJCAI-21*, 2021.

[Zhang *et al.*, 2017] Lingyu Zhang, Tao Hu, Yue Min, Guobin Wu, Junying Zhang, Pengcheng Feng, Pinghua Gong, and Jieping Ye. A taxi order dispatch model based on combinatorial optimization. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 2151–2159, 2017.

[Zhou and Sethu, 2002] Yunkai Zhou and Harish Sethu. On the relationship between absolute and relative fairness bounds. *IEEE Communications Letters*, 6(1):37–39, 2002.

[Zhou *et al.*, 2023] Ze Zhou, Claudio Roncoli, and Charalampos Sipetas. Optimal matching for coexisting ride-hailing and ridesharing services considering pricing fairness and user choices. *Transportation Research Part C: Emerging Technologies*, 156:104326, 2023.