# Dynamic Pricing for Vehicle Dispatching in Mobility-as-a-Service Market via Multi-Agent Deep Reinforcement Learning

Guolin Sun, *Member, IEEE*, Gordon Owusu Boateng, *Member, IEEE*, Kai Liu, Daniel Ayepah-Mensah, and Guisong Liu

***Abstract*—Vehicle dispatching in the mobility-as-a-service (MaaS) market has gradually become a situation of multi-service provider competition and coexistence. However, most existing research on vehicle dispatching with dynamic pricing for the MaaS market is still limited to single-service provider scenarios. In this paper, we propose an economic model that analyzes the vehicle dispatching service pricing and demand interactions between multiple mobility service providers (MSPs) and passengers, respectively. We formulate the vehicle dispatching service pricing and demand problem as a two-stage Stackelberg game under different pricing schemes, namely, *independent pricing scheme (IPS)* and *competitive pricing scheme (CPS)*, considering the vehicle supply-demand relationship and market competition among the MSPs. The MSPs, as leaders, set their service pricing strategies first, and then the passengers, as the followers, determine their service demands. Due to the high-dimensional and complicated nature of the dynamic MaaS market environment, we develop a multi-agent deep reinforcement learning (MADRL) algorithm to achieve the Nash equilibrium (NE) of the formulated game, which indicates the optimal pricing and demand strategies for MSPs and passengers. Simulation results and analysis show that the proposed MADRL-based algorithm converges to the optimal solution and outperforms other benchmark schemes under both IPS and CPS in terms of maximizing MSPs' revenue and protecting passengers' benefits. Furthermore, the proposed MADRL-based algorithm under CPS improves MSPs' market attractiveness and long-term benefits, which encourages MSPs to participate in competitive vehicle dispatching in the MaaS market.**

Manuscript received 16 May 2023; revised 4 February 2024; accepted 13 March 2024. Date of publication 19 March 2024; date of current version 15 August 2024. This work was supported in part by the Natural Science Foundation of China under Grant 61806040 and Grant 61771098, in part by Chengdu Science and Technology Program, under Grant 2023-JB00-00016-GX, in part by the Natural Science Foundation of Sichuan Province, China under Grant 2022YFG0314, in part by the Natural Science Foundation of Guangdong Province, China under Grant 2021A1515011866, in part by Sichuan Science and Technology Program under Grant 2020YFQ0025, in part by Yibin Science and Technology Program under Grant DZKJDX2021020005, and in part by the fund from Intelligent Terminal Key Laboratory of Sichuan Province under Grant SCITLAB-1018 and Grant SCITLAB-20019. The review of this article was coordinated by Dr. Tao Dusit Niyato. *(Corresponding author: Guolin Sun.)*

Guolin Sun, Kai Liu, and Daniel Ayepah-Mensah are with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China, and also with the Intelligent Terminal Key Laboratory of Sichuan Province, Yibin 644005, China (e-mail: guolin.sun@uestc.edu.cn).

Gordon Owusu Boateng is with the School of Information and Communication Engineering, University Electronic Science and Technology of China, Chengdu 611731, China.

Guisong Liu is with the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu 610074, China.

Digital Object Identifier 10.1109/TVT.2024.3378968

***Index Terms*—Dynamic pricing, MaaS, MADRL, Stackelberg game, vehicle dispatching.**

## I. Introduction

FOR most of the last few years, the integration of seamless mobility for commuters has been considered a futuristic concept of urban mobility. This concept is already embodied by the Mobility-as-a-Service (MaaS) market. MaaS is a revolutionary transportation paradigm that merges existing and upcoming mobility services into a single digital platform, enabling customized door-to-door transportation, trip planning, and payment choices tailored for each individual [1]. Rather than purchasing personal modes of transportation or supplementing existing ones, passengers will order mobility service options suited to their specific requirements or pay per trip for tailor-made travel options. Indeed, there is a significant degree of anticipation since it is believed that MaaS will enhance travel experience, cut travel costs, and better manage travel demand while reducing environmental and social impact. The concept of vehicle dispatching is a novel addition to the MaaS framework, where passengers can request vehicles online for short-term use once they subscribe to a mobility service [2]. Thus, it is becoming increasingly common for people to use Internet platforms and personal terminals to access a variety of vehicle dispatching services as a primary mode of transportation. This phenomenon has become significant in optimizing urban transportation resources, alleviating traffic congestion, lowering carbon emissions, and fostering the sustainable growth of an environmentally friendly economy, as the ride-hailing and vehicle dispatching ideas develop [3].

With several anti-monopoly regulatory measures introduced, new developments in vehicle dispatching for the MaaS market have resulted in several mobility service providers (MSPs) emerging and coexisting to provide mobility services to passengers. MSPs compete to gain a reasonable share of the MaaS market, expanding passengers' choice of mobility services. Nonetheless, the competition among MSPs for passengers has become more complex, making it difficult to establish a level playing field for the orderly operation and development of the MaaS market. Some existing works have applied pricing models for optimal vehicle allocation while examining the influence of competition and profit maximization of operators, in an attempt to address issues of operating an orderly MaaS market [4], [5],

[6]. In [4], a surge price predictive model was employed by multiple vehicle ride-sourcing operators to manage vehicles efficiently, save passengers money and time, and provide profitable insight to drivers by encapsulating the complex dynamics of service fleets and demand in real-time. However, the proposed predictive model did not consider the effect of market pricing competition of operators on the market. Therefore, some authors investigated the optimal behavior in monopoly and duopoly market situations and quantified the effects of competition on customers and firms [5]. Practically, an increasing number of operators in the MaaS market allows passengers to have different options to choose from. To create a multi-operator market, the authors in [6] formulated the operator pricing problem in electric vehicles as a competitive market using a two-stage Stackelberg game, where the price of one operator has a linear influence on the number of existing customers of the other operators in the market. However, this work ignored the vehicle supply-demand relationship by focusing solely on operator demand for car-sharing services. We note that most existing works, if not all, focus on one aspect of the problem, e.g., considering network load (supply-demand relationship) or market competition. Moreover, a common limitation of these works is the use of traditional/conventional optimization methods such as model predictive control and backward induction to achieve their respective optimal strategies. In fact, these traditional optimization methods may perform poorly in dynamic environments such as the MaaS market. To this end, it is imperative to study the competitive vehicle dispatching service problem in the MaaS market by jointly considering the supply-demand relationship of MSPs and passengers and the market competition among the MSPs, while intelligently achieving optimal optimization policies in the dynamic environment.

Motivated by the above-mentioned limitations, this paper designs a framework for vehicle dispatching-ride request matching between multiple competing MSPs and passengers in the MaaS market. Due to the conflicting objectives of the stakeholders involved (MSPs and passengers), it is imperative to model the dynamic and strategic interactions between them to capture the real-time pricing and demand strategies. Game theory, e.g., Stackelberg game, has emerged as an analytic tool for modeling interactions among stakeholders with contradicting objectives. Therefore, we model the MSPs' vehicle dispatching service pricing and passengers' service demand problem as a two-stage Stackelberg game while considering the vehicle supply-demand relationship and the market competition between MSPs. We assume that all MSPs offer *Didi Chuxing service*, a taxi dispatching service in China, to passengers, and there is fierce competition among MSPs for a large market share. As a result, we investigate the influence of market competition on MSPs' pricing strategies by analyzing two types of pricing schemes: *independent pricing scheme (IPS)* and *competitive pricing scheme (CPS)*. Unlike traditional model-based approaches that are inherently limited by the pre-specified model, we propose a model-free multi-agent deep Q-network (MADQN) algorithm to achieve the Nash equilibrium (NE) for the formulated game by adapting to dynamics in the MaaS environment. In summary, the main contributions of our work are outlined as follows:

- We propose an intelligent framework for competitive vehicle dispatching service pricing and passenger service demand problem between multiple MSPs and passengers in the MaaS market.
- We formulate the interaction between MSPs and passengers as a two-stage multi-leader multi-follower Stackelberg game, where the MSPs act as leaders by setting their vehicle dispatching service prices first, and the passengers act as followers to determine their service demands.
- To achieve a NE, we reformulate the game-based optimization problem as a Markov decision process (MDP) and propose a MADQN-based scheme that achieves the joint optimal service pricing and demand strategies of maximizing both MSPs and passengers' benefits.
- Finally, we conduct comprehensive simulations to evaluate the performance of the proposed algorithm under IPS and CPS. The simulation results and analysis prove that our proposed scheme outperforms other baseline schemes.

The rest of the paper is organized as follows: Section II presents related works, and Section III presents the system model. Section IV presents the problem formulation and algorithm. Section V discusses the simulation results and analysis, and finally, Section VI concludes this paper.

## II. Related Works

The rapid development of vehicle dispatching or ride-hailing services in the MaaS market has attracted extensive attention from researchers in industry and academia. In the following subsections, we discuss related works on pricing models, market competition, and reinforcement learning (RL)-based solutions in ride-hailing or vehicle dispatching services.

### A. Ride-Hailing Pricing Models

Some authors have proposed several pricing schemes to improve operator revenue and user interests. Wang et al. proposed an equilibrium model with a single taxi-hailing service, and a partial-derivative-based sensitivity analysis was conducted to quantitatively evaluate the impact of the platform's pricing strategies on the taxi market performance [7]. Based on the relationship between user demand and vehicle supply, Castillo et al. studied how surge pricing and other market design interventions can prevent the wild goose chase problem from crippling the ride-hailing market by establishing a theoretical model to analyze surge pricing [8].

Unlike the works focusing on the theoretical pricing strategy models, other existing works have proposed dynamic pricing solutions to balance vehicle supply and demand. Battifarano et al. constructed a data-driven L1-regularized log-linear model as a surge multiplier model [4]. Compared to the pricing mechanisms using single-source data to determine surge multipliers, the model combined multi-source data such as traffic speeds, events, road closures, weather conditions, and real-time supply and demand of vehicles across regional networks to determine regional-level surge multipliers. Some authors proposed a user request-specific pricing model similar to Didi Chuxing's travel

distance and time-based pricing structure to determine the optimal pricing strategy based on regional differences [9]. To promote traffic efficiency and reduce the dispatch cost of inconvenient demand, Iacobucci et al. determined the optimal pricing strategy based on regional-level user demand to balance supply and demand, which assumed that users could choose other alternative transportation modes [10]. Furthermore, He et al. incorporated order cancellation behavior or choice intention into a market with street-hailing and e-hailing taxi services and designed pricing and penalty/compensation strategies based on the proposed equilibrium model to maximize platform revenue or social welfare [11]. To evaluate the performance and application scenarios of the multiplier-based pricing schemes, the authors in [12] examined the effects of two pricing schemes–uniform pricing and multiplier-based pricing on operator profits and participant payoff by formulating game-based models between operator and passenger.

The works discussed so far considered factors affecting ride-hailing pricing and supply-demand relationships. However, they did not capture the competition among service operators; meanwhile, vehicle dispatching or ride-hailing markets are rather oligopolies in many cities.

### B. Market Competition in Ride-Hailing Scenarios

Turan et al. formulated the platform operator's profit maximization problem by adopting a network-flow-based model and analyzing the operator's pricing strategies based on user choice intentions in a duopoly scenario [5]. Several prior works have applied game-theoretic methods to decide the optimal pricing strategy in the vehicle ride-hailing MaaS market. Amar et al. formulated the territory sharing problem as a cooperative game and developed a bargaining-based solution model for cooperative territory allocation [13]. In [14], a review of the application of Stackelberg game approaches in energy trading among electric vehicles was conducted. Some authors used a Stackelberg game to model the interaction between a municipality and MSPs to derive an optimal pricing strategy [15]. Cheng et al. assumed a joint venture of all operators to provide shared electric cars for different competing operators [6]. Operators shared the dividends from the joint venture profits based on their market share. These competitors provided differentiated services for their customers through their electric mobile Apps, where the vehicle allocation problem between the joint venture and car-sharing operators was modeled as a two-stage Stackelberg game to analyze the operator's optimal pricing to users. In [16], the authors investigated the pricing competition among multiple charging station operators in the mobility-on-demand market. Charging station operators determined their charging pricing strategies to maximize their profits based on the power procurement cost, historical charging prices of all station operators, etc.

In summary, most of the aforementioned related works have common features of either of the following; theoretical and dynamic pricing strategy models to balance vehicle supply and demand, and competition among vehicle service operators in oligopoly and duopoly markets. Moreover, these works proposed conventional model-based optimization approaches to make pricing and profit maximization decisions. These approaches are problematic in adapting to dynamic environments since they have little insight into the real-time operation of the MaaS market and require accurate vehicle and passenger data to make well-informed decisions.

### C. Ride-Hailing Policy Optimization With RL

DRL has been applied to resource optimization and scheduling problems for ride-hailing scenarios, including taxi repositioning [17] and taxi dispatching services [2]. DRL methods learn a policy by interacting with the complex environment to capture and model real-time changes in dispatching policies. Some authors presented a context-aware taxi dispatching approach that incorporates rich contexts into DRL modeling for efficient taxi allocations [2]. In [18], Shi et al. proposed a DQN-based transportation network company (TNC) route scheduling algorithm to allow the TNC service center to schedule routes to vacant TNC vehicles. However, using a centralized agent to model the allocation policy increases the action state to unbearable levels. Therefore, Li et al. proposed a distributed MARL technique to address the order dispatching problem in large ridesharing scenarios [19]. In [17], a MARL-based framework was proposed to mitigate the disequilibrium of supply and demand by repositioning taxis at a city scale.

To the best of our knowledge, this is the first paper that jointly considers the supply-demand relationship and market competition among multiple MSPs while intelligently achieving optimal optimization policies for vehicle dispatching in the MaaS market. Specifically, we propose a competitive pricing framework based on MADQN in a multi-MSP vehicle dispatching scenario, considering the supply-demand relationship and market competitiveness of the MSPs.

## III. System Model

### A. System Architecture

We consider an urban settlement area where multiple MSPs coexist, and commuters prefer vehicle dispatching services for ease of mobility and cost minimization. We design a vehicle dispatching service framework for a typical MaaS scenario, which comprises the following entities as depicted in Fig. 1: MSPs who own a set of vehicles, passengers (commuters), and an authorized market agency (MA). The MSPs are vehicle dispatching service providers who dispatch vehicles to the passengers' locations to serve their ride requests for revenue in return. It is assumed that all the MSPs offer Didi Chuxing vehicle dispatching service to prospective passengers. The passengers negotiate service prices with multiple MSPs and select their preferred MSP for ride-hailing based on service reliability and cost. The authorized MA is a centralized vehicle dispatcher on the Didi Chuxing service platform (dispatching center) that acts as an intermediary between the MSPs and the passengers and is tasked with matching passengers' requests to the appropriate MSPs considering their service demands and vehicle dispatching service pricing strategies, respectively.

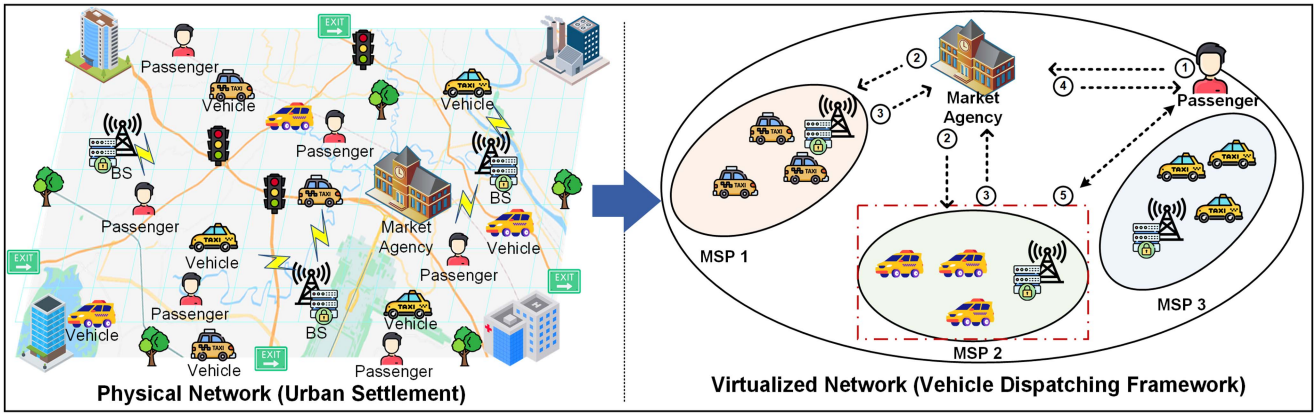



Fig. 1. System architecture.

TABLE I
KEY SYMBOLS AND DEFINITIONS

| Symbol | Definition | Symbol | Definition |
|---|---|---|---|
| $\mathcal{N}$ | Set of MSPs | $q_i$ | Service quality of MSP $i$ |
| $\mathcal{M}$ | Set of passengers | $x_j$ | Passenger $j$'s service demand for MSP $i$ |
| $N$ | Number of MSPs | $c_i$ | Cost of vehicle fuel consumption per unit mileage of MSP $i$ |
| $M$ | Number of passengers | $p_i$ | Unit price charged by MSP $i$ for service |
| $SU_i$ | Utility of MSP $i$, $\forall i \in \mathcal{N}$ | $\omega_i$ | Cost efficiency of MSP $i$ |
| $BU_j$ | Utility of passenger $j$, $\forall j \in \mathcal{M}$ | $\varpi_i$ | Market competitiveness of MSP $i$ |
| $T_{max}$ | Maximum pickup time | $\vartheta$ | Load in the market scenario |
| $T_i$ | Pickup time of MSP $i$ | $d_{ij}$ | Scheduled distance traveled by a dispatched vehicle of MSP $i$ to complete passenger $j$'s service request |

A detailed mode of operation of the vehicle dispatching service framework (right side of Fig. 1) is described as follows:

① The MA receives a passenger's ride request information.

② Then, the MA broadcasts the passenger's ride request to the available MSPs in close proximity to the passenger's location (MSP 1 and MSP 2 in our scenario).

③ Each MSP responds to the ride request by sending its vehicle dispatching service-related information, e.g., service response time, the service unit price, vehicle availability, etc., to the MA. Each MSP seeks to carefully adjust its pricing strategy to maximize its revenue and improve its market competitiveness. Setting an appropriate service price will attract more passengers to prefer a vehicle from the specific MSP, resulting in higher revenue and a large market share.

④ Then, the MA relays the vehicles' dispatching service information to the passenger.

⑤ Next, the passenger decides its service demand strategy following the service prices of the competing MSPs. Finally, the MA selects the winner-MSP (MSP 2) and matches it with the passenger by dispatching a vehicle from the MSP to the passenger's location to serve the ride request.

We assume that a successful vehicle dispatching-ride request matching between the MSP and the passenger is recorded at the dispatching center as a transaction in a secure and transparent manner using blockchain technology [20]. To achieve this, the vehicle dispatching service framework is decentralized by linking the MA (centralized dispatcher) to all the other stakeholders to annul its centrality [21]. In this way, the MA can broadcast all transactions on the blockchain platform, and all entities can store copies of such transactions in their databases as backups. However, the security aspect of the vehicle dispatching transactions is out of the scope of this research. To better understand the notations in the subsequent models presented below, we summarize the key symbols and their definitions in Table I.

### B. Network Model

We consider a business scenario where MSPs coexist and compete to provide vehicle dispatching services to passengers (commuters) for their mobility needs. Let $i$ denote an MSP that provides vehicle dispatching service to a passenger $j$, so that the sets of MSPs and passengers are defined as $\mathcal{N} = \{1, 2, \ldots, i, \ldots, N\}$ and $\mathcal{M} = \{1, 2, \ldots, j, \ldots, M\}$, respectively. Each MSP $i$ owns a total number of $V_i$ vehicles, which can be dispatched to the passengers' locations to provide ride-hailing services upon request. We indicate $[V_1, V_2, \ldots, V_i]$ and $[F_1, F_2, \ldots, F_i]$, $\forall i \in \mathcal{N}$ as the total number of vehicles and the number of idle vehicles owned by MSP $i$, respectively. We define the service demand of passengers at different periods as $[\mu^1, \mu^2, \ldots, \mu^t]$, which includes the service demands during light, medium, and heavy load periods. We define the load $\vartheta^t$ at period $t$ as the ratio of passengers' service demands $\mu^t$ to the total number of vehicles in the market as;

$$\vartheta^t = \frac{\mu^t}{\sum_{i \in \mathcal{N}} V_i}, \tag{1}$$

where $\vartheta^t$ is a global state quantity, which reflects the supply and demand relationship of vehicles in the market.

It is obvious that compared to the off-peak period of passenger travels, the load $\vartheta$ during peak periods of passenger travels is

higher. Thus, during periods of high load, the supply and demand of vehicles vary greatly, which may cause passengers to wait longer to be matched with vehicles. Also, since the total number of vehicles owned by MSPs and their spatial distribution are different, the travel distance $d_{ij}$ and pickup time $T_i$ of each MSP $i$ when dispatching a vehicle to the pickup location of passenger $j$ will vary. The pickup time $T_i$ is a direct reflection of the service quality $q_i$ of MSPs, which directly affects the passenger's satisfaction towards vehicle dispatching service provisioning. Hence, MSPs with high service quality dispatch vehicles to the passenger's pick-up location faster. Thus, we model the relationship between the service quality $q_i$ of an MSP $i$ and the pickup time $T_i$ as;

$$q_i = \log\left(1 + \frac{T_{\max}}{T_i}\right), \tag{2}$$

where $T_{\max}$ represents the maximum pickup time defined by the market framework. An MSP $i$ is qualified to conduct vehicle dispatching transactions with passengers, for any ride request only if its pickup time $T_i$ is less than the maximum pickup time $T_{\max}$, i.e., $T_i < T_{\max}$.

Generally, passengers who are sensitive to service quality are more willing to choose an MSP with higher service quality to complete their ride requests and in turn, achieve higher service satisfaction. Therefore, we model passenger $j$'s service satisfaction with respect to MSP $i$'s service quality $q_i$ as follows [22];

$$Sat_j = \alpha \ln(1 + x_j \cdot q_i), \tag{3}$$

where $\alpha$ is the service quality satisfaction coefficient, which is a curve fitting parameter of (3) modeled to the real-world experiments and indicates the sensitivity of passenger $j$ towards the service quality $q_i$ provided by MSP $i$, and $x_j$ indicates passenger $j$'s service demand to MSP $i \in \mathcal{N}$. We use a log function to model passenger service satisfaction because it is monotonically increasing and reflects the law of diminishing returns [22]. We define a common value of $\alpha$ for all passengers associated with one MSP, which is why it is unassociated with a specific passenger $j$. The optimal $\alpha$ is determined by nonlinear least squares fitting [23].

### C. Business Model

In the vehicle dispatching service market framework, passenger $j$ issues a ride request through its terminal with service request information, including its personal information, pick-up location, drop-off location (destination), etc., to trigger a vehicle dispatching transaction. Then, MSP $i$ owning vehicles can dispatch a vehicle to complete the passenger's ride request if selected. The dispatched vehicle travels a distance $d_{ij}$ to transport passenger $j$ from its pick-up location to the drop-off location (destination), and the passenger pays a fee $d_{ij} \cdot p_i$ to MSP $i$, where $p_i$ is the price charged by the MSP $i$ per unit mileage. The cost of vehicle fuel consumption per unit mileage incurred by the MSP $i$ to transport the passenger to its destination is denoted as $c_i$. Thus, MSP $i$ earns revenue $d_{ij}(p_i - c_i)$ by completing passenger $j$'s ride request for its mobility requirements.

Several existing works have investigated the supply and demand relationship of vehicles in the MaaS market to obtain closed-form models with dynamic pricing [9], [12], [24]. In the dynamic price model proposed in [9], the user's vehicle (taxi) cost increases with the increase of the operator's service response time, which means that in the marginal area where the supply of vehicles is insufficient, or the difference between supply and demand is large in a period, the operator charges a higher fee. However, in [24], the authors calculate the dynamic price directly based on the difference between supply and demand, and the price set by the operator increases with the increase in passenger demand. Similarly, the authors in [12] obtained dynamic prices based on differences in user demand during high and low peak periods. Thus, motivated by the above works, we can obtain a dynamic price related to the load $\vartheta^t$ in the market. However, we assume that there are multiple MSPs in the vehicle dispatching service framework that specifically provide Didi Chuxing vehicle dispatching service to passengers. This results in competition among MSPs to maximize their operating revenue and obtain a reasonable market share. Thus, due to competition, each MSP needs to continuously adjust its service unit price in order to adapt to the passengers' service demand strategy, which will improve their revenue and enable them to gain a sizable market share. Consequently, we define two pricing schemes as follows:

*Independent pricing scheme (IPS):* In IPS, each MSP only considers its overall vehicle load (supply and demand relationship) in the market scenario to determine its unit price independently, without considering other MSPs' information (without considering competition with other MSPs), such as service quality and historical vehicle unit price. That is, under IPS, MSPs behave as if they are the only SPs in the network; they set their unit prices independently and not based on the unit prices set by other MSPs. Therefore, the pricing strategy under IPS is not influenced by competitors.

*Competitive pricing scheme (CPS):* In CPS, the MSP not only considers the load of vehicles in the market but also considers its own pricing strategy and other competitors' historical unit pricing strategies, service quality, and other information for dynamic pricing. That is, under CPS, the pricing strategy of an MSP is influenced by its competing MSPs. In this case, market competitiveness reflects the total profit since each MSP seeks to set reasonable unit prices to achieve reasonable profits and large market share.

From the perspective of vehicle dispatching service provisioning, MSPs deliver differentiated service quality and unit price for ride requests, resulting in differences in the service cost efficiency of different MSPs. We model the service cost efficiency $\omega_i$ of MSP $i$ as follows;

$$\omega_i = \frac{q_i}{p_i} \log\left(1 + \frac{F_i}{V_i}\right), \tag{4}$$

where $p_i$, $q_i$, and $\frac{F_i}{V_i}$ denote the unit price, the service quality, and the vehicle availability rate of MSP $i$. The higher the vehicle availability rate, the more likely MSP $i$ can provide more passengers with high-quality vehicle dispatching services. From (4), it

can be seen that MSPs can improve their service cost efficiency $\omega_i$ by reducing the unit price $p_i$ for vehicle dispatching service and improving service quality $q_i$. The more cost-efficient the service of MSP $i$ is, the more likely it is to have higher market competitiveness and a larger market share. We model MSP $i's$ competitiveness $\varpi_i$ as the ratio of its service cost efficiency to the sum of service cost efficiencies of all MSPs, and is given by;

$$\varpi_i = \frac{\omega_i}{\sum_{i\in\mathcal{N}} \omega_i}. \tag{5}$$

From (5), it can be found that as the service cost efficiency of MSP $i$ increases, its market competitiveness $\varpi_i$ increases. This enables the MSP to attract passengers to subscribe to its service.

### D. Utility Model

Let $BU_j$ denote the benefit of passenger $j$ (buyer), and $SU_i$ denote the benefit of an MSP $i$ (seller). For vehicle dispatching service transactions, passengers usually want to obtain high-quality services at a lower cost. Hence, service satisfaction and vehicle dispatching service cost are the main concerns for passengers. Thus, we define the utility $BU_j$ of passenger $j$ as the difference between the passenger service satisfaction $Sat_j$ experienced and the cost of enjoying vehicle dispatching service $p_i \cdot x_j$ from MSP $i$, which is expressed as;

$$BU_j(x_j, p_i) = Sat_j - (x_j \cdot p_i). \tag{6}$$

*Utility of MSP in IPS:* In IPS, each MSP $i$ dynamically adjusts its unit price to balance the vehicle supply and demand based on the vehicle load in the market. Furthermore, the higher MSP $i's$ service quality in the market, the higher a passenger has a strong intention to choose it for its mobility service needs. Therefore, we model MSP $i's$ utility $SU_i$ under IPS as follows;

$$SU_i(x_j, p_i) = x_j(p_i - c_i)d_{ij}, \tag{7}$$

where $p_i$ represents the unit price of MSP $i$ relative to the load in the market, $c_i$ represents vehicle fuel consumption cost, and $d_{ij}$ indicates the distance traveled. The value of $p_i$ changes as the load or supply-demand relationship in the market varies.

*Utility of MSP in CPS:* In CPS, each MSP $i$ dynamically adjusts its unit price, on the one hand, to balance the supply and demand of vehicles and on the other hand, to enhance the competitiveness $\varpi_i$, expanding its market share and making reasonable profits. Therefore, based on the competitiveness $\varpi_i$ of MSP $i$ and the vehicle load in the market, we model the utility $SU_i$ of MSP $i$ under CPS as follows;

$$SU_i(x_j, p_i, \mathbf{P}_{-i}) = \varpi_i \cdot x_j(p_i - c_i)d_{ij}, \tag{8}$$

where $\mathbf{P}_{-i}$ is the pricing strategy set of all other MSPs except MSP $i$.

## IV. Problem Formulation and Algorithm

Several MSPs coexist and compete in the network to provide passengers with vehicle dispatching services, and passengers can choose any MSP to meet their service requests. For vehicle dispatching transactions between passengers and MSPs in the MaaS market, a passenger decides its service demand by comparing the mobility service quality and the unit price of the vehicle dispatching service of the MSPs. The MSPs, on the other hand, dynamically adjust their pricing strategies based on the supply and demand of vehicles and the competition amongst them to improve their operating revenue and enhance market competitiveness. Therefore, in this section, we model the interaction between passenger $j$ and MSP $i$ for vehicle dispatching service transactions using the Stackelberg game model. Finally, we formulate the decision-making problem in the Stackelberg game as an MDP to solve the NE that achieves optimal service pricing and demand strategies based on a MADQN algorithm.

### A. Two-Stage Stackelberg Game Formulation

In the vehicle dispatching service transaction, MSP $i$ as a seller provides MaaS by dispatching its vehicles to serve ride requests, and passenger $j$ as a buyer requests the service based on its demand. In other words, MSP $i$ trades its vehicle with passenger $j$ for an incentive (revenue) in return. Therefore, MSPs and passengers decide their vehicle dispatching service strategies to maximize their respective utilities. In this sequel, we refer to the interactions between MSP $i$ and passenger $j$ for vehicle dispatching-ride request matching as a trading game. We mention that we formulate the trading game problem for only MSPs under CPS since it is similar to the game formulation for MSPs under IPS. The only difference is that the MSPs under IPS do not consider competition with the other MSPs in the MaaS market. Hence, the utility function in (8) is substituted with (7) when dealing with IPS.

Based on non-cooperative game theory, we model the MSP and passenger vehicle dispatching problem as a trading-based two-stage multi-leader multi-follower Stackelberg game to obtain optimal utility strategies [25]. In stage I of the game, MSP $i$ acts as a leader and determines its service unit price considering the overall supply and demand relationship of vehicles and the competitive unit prices set by all other MSPs in the MaaS market. Then in stage II, passenger $j$ as a follower decides its service demand to MSP $i$. We assume that both MSPs (sellers) and passengers (buyers) are rational in the game, deciding their optimal utility strategies to maximize their interests based on each player's strategies.

We define the unit price set by MSP $i$ using the vector $\mathbf{P} = \{p_1, p_2, \ldots, p_i, \ldots, p_N\}$, $i \in \mathcal{N}$ and the service demand of passenger $j$ to MSP $i$ using the vector $\mathbf{X} = \{x_1, x_2, \ldots, x_j, \ldots, x_M\}$, $j \in \mathcal{M}$. Furthermore, we indicate the optimal unit price and optimal service demand strategies of MSP $i$ and passenger $j$ at NE as $\mathbf{P}^* = \{p_1^*, p_2^*, \ldots, p_i^*, \ldots, p_N^*\}$ and $\mathbf{X}^* = \{x_1^*, x_2^*, \ldots, x_j^*, \ldots, x_M^*\}$, respectively. In the two-stage Stackelberg game, passenger $j$ responds to the unit price set by each MSP by making a decision on its optimal service demand that maximizes its utility. Similarly, each MSP seeks to choose an optimal pricing strategy that maximizes its utility. The optimal service price of an MSP should be moderate while improving its cost efficiency in order to be chosen as the MSP-winner to provide the passenger with mobility service. Both MSP (leader) and passenger (follower) keep adjusting their trading game strategies to maximize their utilities. Thus, we represent the utility optimization problem of the game by dividing it into two sub-problems as follows:

**Algorithm 1:** Stackelberg Game-based Trading Procedure.

**Initialize:** Set $N$ MSPs, $M$ passenger service requests
**for** each episode **do**
  **for** each passenger request $j$ **do**
    MA receives passenger $j's$ service request and broadcasts the request information to MSP $i$
    **Stage I:** MSP $i$ decides its pricing strategy $\mathbf{P} = \{p_1, p_2, \ldots, p_i, \ldots, p_N\}$ via solving Eq. (9) as Algorithm 2
    MA sends pricing strategy $\mathbf{P}$ to passenger $j$
    **Stage II:** Passenger $j$ receives the pricing strategy $\mathbf{P}$ of MSP $i$ and decides its service demand $\mathbf{X} = \{x_1, x_2, \ldots, x_j, \ldots, x_M\}$ via solving Eq. (10)
    MA chooses the MSP-winner $i$ who gets the maximum utility value
    MSP-winner $i$ dispatches a vehicle to passenger $j$ to provide its mobility service needs
  **end for**
**end for**

*Stage I (Leader's pricing strategy):* Each MSP anticipates a passenger's service demand by receiving information from the MA. The MA determines the MSP-winner after providing the passenger with the pricing strategy of the said MSP. The optimal pricing strategy of MSP $i$ is the strategy that can maximize its utility with a given unit price $p_i$. Thus, we formulate the unit price optimization problem of MSP $i$ as;

$$\max_{p_i} \quad [SU_i\left(x_j, p_i, \mathbf{P}_{-i}\right)]$$
$$\text{s.t.} \quad c_i \leq p_i \leq p_{\max}, \quad \forall i \in \mathcal{N}, \tag{9}$$

where $\mathbf{P}_{-i}$ is the pricing strategy set of all other MSPs except MSP $i$, and $p_{\max}$ is the reasonable maximum unit price a passenger is willing to pay. The constraint in (9) indicates that the unit price $p_i$ set by MSP $i$ should not be below the cost of fuel consumption $c_i$ and not exceed the maximum unit price $p_{\max}$.

*Stage II (Follower's service demand strategy):* Based on the unit price $p_i$ set by MSP $i$, passenger $j$ decides its service demand strategy $x_j$ to select an MSP to provide it with mobility service. The service demand optimization problem of passenger $j$ is formulated as;

$$\max_{x_j} \quad [BU_j\left(x_j, p_i\right)]$$
$$\text{s.t.} \quad x_j \geq 0, \quad \forall i \in \mathcal{N}, \forall j \in \mathcal{M}. \tag{10}$$

The optimal demand strategy for vehicle dispatching service transactions in the passenger sub-game problem is to solve (6) so that the passenger utility $BU_j$ is maximized by $x_j$, where the service demand $x_j \geq 0$.

*Definition 1(Nash Equilibrium):* The points $x_j^*$ and $p_i^*$ in $(x_j^*, p_i^*)$ are defined as the NE if $SU_i(x_j^*, p_i^*, p_{-i}^*) \geq SU_i(x_j^*, p_i, p_{-i}^*)$ and $BU_j(x_j^*, p_i^*) \geq BU_j(x_j, p_i^*), \forall j \in \mathcal{M}, i \in \mathcal{N}$.

To verify the existence and uniqueness of the NE, we take the first-order and second-order derivatives of $SU_i(x_j, p_i, \mathbf{P}_{-i})$ and $BU_j(x_j, p_i)$ [26].

*Theorem 1:* There is a unique NE at the point $(x_j^*, p_i^*)$.

*Proof:* Given the unit price $p_i$ of MSP $i$, passenger $j$ maximizes its utility by deciding its optimal service demand $x_j^*$. We take the first-order and second-order derivatives of the passenger utility $BU_j(x_j, p_i)$ in (6) as;

$$\frac{\partial BU_j}{\partial x_j} = \frac{q_i}{1 + x_j \cdot q_i} - p_i \tag{11}$$

$$\frac{\partial^2 BU_j}{\partial {x_j}^2} = \frac{-{q_i}^2}{(1 + x_j \cdot q_i)^2} \tag{12}$$

We can get $\frac{\partial^2 BU_j}{\partial {x_j}^2} < 0$, since $x_j \geq 0$ and $q_i \geq 0$. Therefore, $BU_j(x_j, p_i)$ is strictly concave and the optimal service demand $x_j^*$ of passenger $j$ can be obtained as;

$$x_j^* = \frac{1}{p_i} - \frac{1}{q_i} \tag{13}$$

Next, we find the optimal unit price $p_i^*$ of MSP $i$ that maximizes its utility. We recall that competitiveness has been defined in (5). We first define the auxiliary pricing strategies as $\{h_i = (\frac{1}{p_i})\}_{i \in \mathcal{I}}$. Let $h_{-i}$ denote the auxiliary pricing strategy of another MSP other than MSP $i$. For simplicity of notations and derivation, we rewrite $\varpi_i$ as $\frac{g_i}{\frac{1}{I}\sum_{l \in \mathcal{I}} g_l h_l}$, where $g_i = \omega_i = \frac{q_i}{p_i} \log(1 + \frac{F_i}{W_i})$. Based on $x_j^*$, MSP $i$ finds the optimal price $p_i^*$ to maximize its utility. We substitute $x_j^*$ into (8) as;

$$SU_i\left(h_i, h_{-i}, x_j^*\right) = \frac{g_i h_i}{\frac{1}{I}\sum_{l \in \mathcal{I}} g_l h_l} \cdot \left(h_i - \frac{1}{q_i}\right) \cdot \left(\frac{1}{h_i} - c_i\right) \tag{14}$$

$$= \frac{I}{\sum_{l \in \mathcal{I}} g_l h_l} \left(g_i h_i - c_i g_i {h_i}^2 - \frac{g_i}{q_i} + \frac{c_i g_i}{q_i} h_i\right) \tag{15}$$

The auxiliary pricing strategy $h_i = (\frac{1}{p_i})$ is a continuous monotonic function of $p_i$. Hence, the noncooperative game of the MSPs is equivalent to $\{I, \{h_i\}_{i \in \mathcal{I}}, \{SU_i(h_i, h_{-i}, x_j^*)\}_{i \in \mathcal{I}}\}$. The auxiliary pricing strategy $h_i$ of MSP $i$ can be defined as $[(\frac{1}{p_{\max}}), (\frac{1}{c_i})]$. From (11), $SU_i(h_i, h_{-i}, x_j^*)$ is continuous in $[(\frac{1}{p_{\max}}), (\frac{1}{c_i})]$. We take the first-order and second-order derivatives of the MSP utility $SU_i(x_j, p_i, \mathbf{P}_{-i})$ in (8) as;

$$\frac{\partial SU_i}{\partial h_i} = -\frac{1}{(\sum_{l \in \mathcal{I}} g_l h_l)^2} \left(g_i{}^2 h_i - c_i g_i{}^2 h_i{}^2 - \frac{g_i{}^2}{q_i} + \frac{c_i g_i{}^2}{q_i} h_i\right) + \frac{1}{\sum_{l \neq i} g_l h_l} \left(g_i - 2 c_i g_i h_i + \frac{c_i g_i}{q_i}\right) \tag{16}$$

$$= -I \frac{\left(g_i - 2 c_i g_i h_i + \frac{c_i g_i}{q_i}\right) \sum_{l \neq i} g_l h_l + \frac{g_i{}^2}{q_i} - c_i g_i{}^2 h_i{}^2}{\left(\sum_{l \in \mathcal{I}} g_l h_l\right)^2} \tag{17}$$

$$\frac{\partial^2 SU_i}{\partial {h_i}^2} = - I \frac{2 c_i g_i \left(\sum_{l \neq I} g_l h_l\right)^2 + \left(2 g_i{}^2 + \frac{c_i g_i{}^2}{q_i}\right) \sum l \neq i g_l h_l + \frac{2 g_i{}^2}{q_i}}{\left(\sum_{l \in \mathcal{I}} g_l h_l\right)^3} \tag{18}$$

We can easily prove $\frac{\partial^2 SU_i}{\partial {h_i}^2} < 0$. This shows $SU_i(h_i, h_{-i}, x_j^*)$ is strictly concave with respect to $h_i$. To prove the uniqueness of the NE, we express $\frac{\partial SU_i}{\partial h_i} = 0$ as;

$$I\left(g_i - 2c_i g_i h_i + \frac{c_i g_i}{q_i}\right)\sum_{l\neq i} g_l h_l + \frac{{g_i}^2}{q_i} - c_i {g_i}^2 {h_i}^2 = 0 \tag{19}$$

$$h_i^* = \sqrt{\sum_{l\neq i} 2 g_l h_l c_i g_i - c_i g_i + \frac{g_i}{q_i}} + I\left(\sum_{l\neq i} g_l h_l g_i + \sum_{l\neq i} g_l h_l \frac{x_i g_i}{q_i}\right) \tag{20}$$

Finally, $p_i^* = (\frac{1}{h_i^*})$.

The specific procedure to achieve the NE of the Stackelberg game of the trading problem is shown in Algorithm 1.

### B. MADRL-Based Algorithm for Utility Optimization

Traditional/conventional methods such as backward induction [27] have been applied to achieve the NE of Stackelberg games, which confirms that the optimal solution of the formulated Stackelberg game exists and is unique. However, the main challenge of these methods is that they rely on historical data for selecting the optimal strategy, which cannot be adapted to dynamic environment scenarios. These methods also have little insight into the real-time operation of the MaaS market and are inherently limited by the pre-specified model. Because MSPs and passengers are non-cooperative, they both do not have access to complete information in the market environment for vehicle dispatching. DRL-based methods can solve dynamic environment problems as they can learn the optimal strategy by interacting with the complex environment without prior knowledge. However, deploying a single-agent DRL method only maximizes a single cumulative reward (utility). This implies that a single agent DRL can only learn the optimal pricing strategies to be the same for all MSPs, which nullifies the essence of analyzing the pricing competition among them. Based on this, we prefer the MADRL method, which deploys an agent on each MSP to select its unique optimal pricing strategy while considering the other agents' strategies in the competitive trading market. Thus, we model the vehicle dispatching problem as an MDP and propose a MADRL method to obtain the optimal strategies. Each agent collects information from the MaaS market environment, trains on the information, and makes optimal decisions when dispatching conditions change.

To customize the generic MADRL framework for our defined vehicle dispatching problem, we design our proposed algorithm considering dynamic pricing of the multiple MSPs (sellers) who compete among themselves to provide a common service (Didi Chuxing service) to their passengers (buyers), i.e., the MADRL-based pricing algorithm. The main challenge in solving the dynamic pricing problem emanates from the fact that the game players (MSPs and passengers) are rational and seek to maximize their individual utilities at the expense of the other players. Therefore, the main goal of our proposed MADRL-based pricing algorithm is to find the NE at which none of the players has an incentive to deviate from the optimal solution. The multiple MSPs adjust their service prices based on supply-demand fluctuations and market competitiveness. Thus, each MSP has its own pricing strategy, which is distinct from the other competing MSPs, to enhance its competitiveness. For the DRL model, we use a common dataset to pre-train all the agents on the MSPs. This will provide a common starting point for all agents, allowing them to learn the underlying patterns in the data. After pre-training, each agent can then continue learning and adapting their actions based on the market competitiveness of the MSP, to select actions that achieve the maximum utility.

We define the state, action, and reward of the MDP with the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}(s'|s,a), \mathcal{R}, \mathcal{S}')$, where $\mathcal{S}$ indicates the state set, $\mathcal{A}$ is the action set, $\mathcal{R}$ denotes the reward function, and $\mathcal{S}'$ indicates next state. $\mathcal{P}(s'|s,a)$ represents the state transition probability, where an action $a$ taken at state $s$ at time step $t$ leads to a new state $\mathcal{S}'$ at next time step $t+1$. The function for the transition probability leading $s^t$ to $s^{t+1}$ when an action $a^t$ is taken is defined as;

$$\mathcal{P}(s'|s^t, a^t) = \begin{cases} 1, & s' = s^{t+1} \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

With $\mathcal{V}^\pi([s^0, s^1, \ldots, s^t], a^t)$ as the Markov chain utility of given policy $\pi$, we define the long-term reward of a given state $s^t$ as the sum of discounted rewards, which can be expressed as;

$$r(s^t) + \sum_{t=1}^{\infty} \gamma^t r\left(s^{t+1}\right), \tag{22}$$

where $\gamma$ indicates the discount factor, which maps the future reward to the current state and is defined within the range 0 to 1. To balance immediate and future rewards, $\gamma$ should be initialized close to 1. The state-value function of a random policy $\mathcal{V}^\pi$ at time step $t$ is expressed as;

$$\mathcal{V}^\pi(s^t) = \mathbb{E}\left\{\sum_{t=1}^{\infty} \gamma^t r(s^t)\right\}. \tag{23}$$

The main goal of MDP is to find the optimal policy $\pi^*$, which maximizes the future reward. Using the Markov property, the policy $\pi$ can be defined as;

$$\mathcal{V}^\pi(s^t) = \mathbb{E}\left\{r\left(s^t, a^t\right) + \gamma \sum_{s'} \mathcal{P}\left(s'|s^t, a^t\right) \mathcal{V}^\pi\left(s'\right)\right\}, \tag{24}$$

where $r(s^t, a^t)$ is the current reward, $\mathcal{V}^\pi(s^t)$ is the current utility, and $\mathcal{V}^\pi(s')$ is the future utility. The state-value function for an optimal policy using the Bellman equation is expressed as;

$$\mathcal{V}^{\pi^*}(s^t) = \underset{a^t \in \mathcal{A}}{\text{argmax}} \left\{\mathcal{V}^\pi(s^t)\right\}. \tag{25}$$

*State-Action-Reward Mapping:* For an agent on MSP $i \in \mathcal{N}$, the state of the environment consists of its market competitiveness $\varpi_i$, quality of service $q_i$, and the vehicle load $\vartheta$. Therefore, the state of the environment for MSP $i$ in the $t-th$ trading game is defined as $s_i^t = (\varpi_i, q_i, \vartheta)$. Based on state $s_i^t$, MSP $i$ selects

the optimal action $a_i^t \in A_i$. We define MSP $i's$ action space as $A_i = \{0.6, 0.7, \ldots, 2.9, 3.0\}$, based on which actions are chosen by the DRL agent on MSP $i$ to determine its unit price. The agent performs action $a_i^t$ to update the pricing policy $p_i$ to enhance its utility $SU_i(x_j, p_i)$. Therefore, the agent's reward $r_i^t$ is defined as;

$$r_i^t = SU_i\left(x_j, p_i, \mathbf{P}_{-i}\right). \tag{26}$$

For the agent on the passenger side, the state of the environment consists of the service quality $q_i$, the pricing policy $p_i$ of MSP $i$, and the vehicle load $\vartheta$ for MSP $i$. Therefore, the state of the environment in the $t-th$ trading game is defined as ${s_j}^t = (q_i, p_i, \vartheta)$. The action definition of the passenger side's agent is the same as that of the MSP side's agent. The agent updates the service demand strategy to improve $BU_j(x_j, p_i)$. Thus, the agent's reward $r_j$ is defined as;

$$r_j^t = BU_j\left(x_j, p_i\right). \tag{27}$$

To arrive at the NE of the formulated game, (26) achieves the optimal unit price ${p_i}^*$ of MSP $i$ and (27) achieves the optimal service demand ${x_j}^*$ of passenger $j$. The optimal rewards for the MSP and passenger achieve the NE point $({p_i}^*, {x_j}^*)$ of the game. Therefore, (26) and (27) jointly determine the final output of the MADRL by calculating the system reward as $r^t = r_i^t + r_j^t$.

*Q-learning (QL):* QL is based on the concept of state-action value function $\mathcal{Q}^\pi(s, a)$ of policy $\pi$, which evaluates the total discounted rewards derived from state $s$ by first taking action $a$ and then following the policy $\pi$. The optimal Q-function $\mathcal{Q}^*(s, a)$ is defined as the maximum reward that can be obtained from QL. The optimal Q-function follows the Bellman optimality equation [28], and is given by;

$$\mathcal{Q}^*(s, a) = \left[r + \gamma \max_{a'} \mathcal{Q}^*\left(s', a'\right)\right]. \tag{28}$$

This indicates that the optimal Q-value from state $s$ and action $a$ is the sum of the immediate reward $r$ and future discounted rewards with discount factor $\gamma$. The fundamental idea underlying QL is to utilize the Bellman optimality equation as an iterative update $\mathcal{Q}^{t+1}(s, a)$. It can be proven that this converges to the optimal Q-function, i.e., $\mathcal{Q}^t \to \mathcal{Q}^*$ as $t \to \infty$ [29]. If an agent's strategy is $\pi$, then the multiple agents' strategy is defined as $\pi = [\pi_1, \pi_2, \ldots, \pi_N]$. Given the strategy $\pi$, we define the state-action value pair as $\mathcal{Q}^\pi(s^t, a^t) = \mathbb{E}_\pi\{r(s^t, a^t), \pi\}$ and the state value as $\mathcal{V}^\pi(s) = \mathbb{E}_{a\sim\pi(s)}[\mathcal{Q}^\pi(s^t, a^t]$.

*Native Deep Q-Network (DQN):* A Q-table with values for every possible combination of $s$ and $a$ is impractical for our problem scenario. This is because, as the state and action spaces increase, the Q-table is overwhelmed and complexity is very high. The DQN approach combines deep neural networks (DNNs) and experience replay to improve on the QL algorithm. Thus, a neural network (NN) with parameter $\theta$ can be trained to estimate the Q-values $\mathcal{Q}(s, a; \theta) \approx \mathcal{Q}^*(s, a)$. The minimization of the following loss $\mathcal{L}^t(\theta^t)$ accomplishes this at each time step $t$ as;

$$\mathcal{L}_t(\theta_t) = \mathbb{E}_{s,a,r,s'\sim\rho(\cdot)}\left[\left(y^t - \mathcal{Q}\left(s, a; \theta^t\right)\right)^2\right], \tag{29}$$

where $y^t = r + \gamma \max_{a'} \mathcal{Q}(s', a'; \theta^{t-1})$ is the temporal difference (TD) target, $y^t - \mathcal{Q}$ is the TD error, and $\rho$ indicates the behaviour distribution, which is over the transition $\{s, a, r, s'\}$ collected from the market environment. Transitions are added to the replay memory at each time step. Then, we compute the loss and gradient during training using a mini-batch of transitions sampled from the replay memory rather than simply the most recent transition. The learning updates become more stable due to taking minibatch samples from the experience replay memory.

**Algorithm 2:** MADQN-based Pricing Algorithm for Vehicle Dispatching.

**Input:** Replay memory capacity $C$, discount factor $\gamma$, epsilon greedy $\epsilon$, exploration increment $\delta$, learning rate $\tau$
**Initialize:** Set replay memory $D$ with capacity $C$, two Q-value NNs with random weights $\theta$ and $\theta^-$
**for** MSP $i$, $i \in \mathcal{N}$ **do**
  MSP $i$ receives $\mathbf{X} = \{x_1, x_2, \ldots, x_j, \ldots, x_M\}$ broadcast by MA
  Agent $i$ observes the current state $s_i^t = (\varpi_i, q_i, \vartheta)$ and randomly selects an action $a_i^t$ with $\epsilon$
  Otherwise select $a_i^t = \text{argmax}_{a_i \in A_i} \mathcal{Q}(s_i^t, a_i^t; \theta)$
  MSP $i$ reports $p_i^t$ to the MA and waits for the MA to return the service demand strategy $x_j^t$
  Compute the utility $SU_i(x_j, p_i, \mathbf{P}_{-i})$ and reward $r_i^t$ using Eq. (16) and observe the new state $s_i^{t+1}$
  Store transition $(s_i^t, a_i^t, r_i^t, s_i^{t+1})$ in memory $D$
  Sample random minibatch $D' = (s_i^k, a_i^k, r_i^k, s_i^{k+1})$ from $D$
  Set: $y_i^k = r_i^{k+1} + \gamma \text{argmax}_{a^{t+1}} \mathcal{Q}'(s_i^{k+1}, a_i^{k+1}; \theta^-)$
  Update $\theta$ by performing a gradient descent step: $Loss = \frac{1}{|D'|}\sum_k^{|D'|}(y_i^k - \mathcal{Q}(s_i^{t+1}, a_i^t; \theta))^2$
  Every $L$ steps replace $\theta^-$ with $\theta$: $\theta^- \leftarrow \tau\theta + (1-\tau)\theta^-$
**end for**

Each agent aims to optimize its value function through a collective policy $\pi$ and Nash Q-values are utilized to represent the Nash Equilibrium in our MADRL framework [30]. For MSP $i$ (same applies to passenger $j$), it is characterized by a specific joint policy $\pi = [\pi_1, \pi_2, \ldots, \pi_N]$, where, for all $s \in \mathcal{S}$ and $i \in \{1, \ldots, N\}$, it holds that:

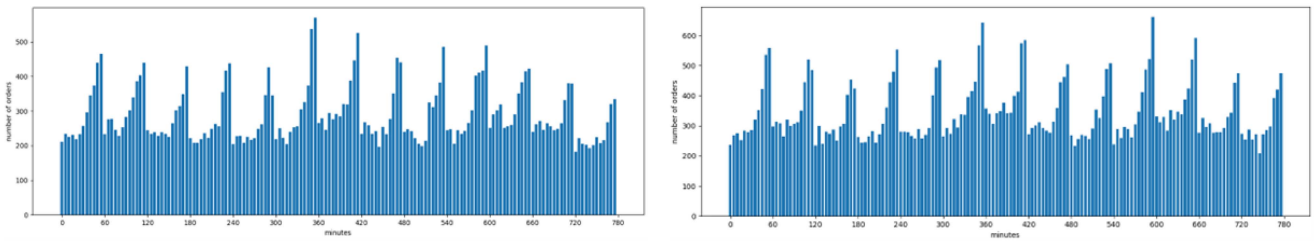
$$\mathcal{V}\left(s, \pi, \pi_{-i}^*\right) \geqslant \mathcal{V}\left(s, \pi_i, \pi^*\right), \tag{30}$$

Based on (30), there is always a set of Nash Q-values $\mathcal{Q} = \{\mathcal{Q}_1, \ldots, \mathcal{Q}_N^*\}$ for each time step $t$ [31]. The Nash $\mathcal{Q}$ value for the trading-based MARL can be defined as [30]:

$$\text{Nash}\ \mathcal{Q}(s, a) = r(s, a) + \gamma \mathbb{E}_{s'}\left[\text{Nash}\ \mathcal{V}_{\pi_*}\left(s'\right)\right]. \tag{31}$$

The NE can be obtained from the current stage game for all agents and the $Q$-value updated accordingly. By repeatedly iterating this process, the Q-value eventually converges to the optimal $Q$-value based on the NE in the game.

We provide Algorithm 2, which is based on native DQN to implement the MADQN algorithm. Due to the limitations of backward induction, Algorithm 2 is used to find the optimal unit price ${p_i}^*$ of each MSP, which is *Stage I* of the Stackelberg game. Then, we use Algorithm 1 to find the optimal service demand, ${x_j}^*$, which is *Stage II* of the Stackelberg game, and select the




(a) (b)

Fig. 2. Didi Order Data Distribution. (a) Data distribution as of November 1, 2016. (b) Data distribution as of November 2, 2016.

winner-MVNO. Since the point $(p_i{}^*, x_j{}^*)$ is defined as the NE, the combined result of Algorithms 1 and 2 achieves the NE (optimal solution) of the formulated Stackelberg game.

The computational complexity of the MADQN algorithm depends on the trainable parameters (state and action sets) and the NN configuration (hidden layers and output). Therefore, the computational complexity of our proposed MADQN-based pricing algorithm is expressed as $\mathcal{O}(\mathcal{G} \times |\mathcal{S}| \times |\mathcal{A}| \times \Gamma)$, where $\mathcal{G}$ is the number of agents, $|\mathcal{S}|$ is the state space, and $|\mathcal{A}|$ is the action space. The term $\Gamma = (\mathcal{O}(|\mathcal{L}|^2\mathcal{H})$ defines the complexity of each evaluation and target NN, where $\mathcal{H}$ represents the number of hidden layers, and $\mathcal{L}$ is the dimension of the output. It is noteworthy that the number of episodes and minibatch size affect the complexity of the algorithm.

### C. Theoretical Convergence Analysis

In this subsection, we delve into the theoretical convergence analysis of the MADQN-based pricing algorithm, adopting a similar approach in [32]. Our primary focus is to demonstrate that (31) represents a contraction mapping, with its fixed point residing at a Nash Q-value denoted as $Q^* = [Q_1^*, \ldots, Q_N^*]$. To establish the convergence of the proposed algorithm, we consider the following assumptions:

*Assumption 1:* Each action-value pair is visited infinitely often, and the reward is bounded by some constant $K$.

*Assumption 2:* The learning rate $\alpha$ should satisfy the following conditions for all $s, a_1 \ldots a_n$.

1) $0 \leq \alpha < 1, \sum_{t=0}^{\infty} \alpha^t = \infty, \sum_{t=0}^{\infty} (\alpha^t)^2 < \infty$.
2) $\alpha_t(s, a_1 \ldots a_n) = 0, if (s, a_1 \ldots a_n) \neq (s^t, a_1^t \ldots a_n^t)$.

Our proof is also built upon the two lemmas as follows:

*Lemma 1:* Assume that $\alpha^t$ satisfies Assumption 2 and the mapping $P^t : \mathbb{Q} \to \mathbb{Q}$ satisfies the following condition: there exists a number $\gamma \in [0, 1]$ and a sequence $\gamma^t$ converging to 0 with probability 1 such that $\|P^tQ - P^tQ^*\| \leqslant \gamma \|Q - Q^*\| + \lambda^t$ for all $Q \in \mathbb{Q}$ and $Q^* = E[P^tQ^*]$, then the $Q$-value updated by;

$$Q^{t+1} = \left(1 - \alpha^t\right) Q^t + \alpha^t \left[P^t Q^t\right], \tag{32}$$

convergences to $Q^*$ with probability 1.

*Proof:* See Theorem 1 in [33] and Corollary 5 in [34] for detailed derivations.

*Definition 1:* Let $Q$ be the set of all agents' value, as $Q = (Q_1, \ldots, Q_N)$, where $Q_i \in \mathbb{Q}_i$ for all $i \in \mathcal{N}$ and $\mathbb{Q} = \mathbb{Q}_1, , , \times \mathbb{Q}_N$. So $P^t : \mathbb{Q} \to \mathbb{Q}$ is a mapping on complete metric space $\mathbb{Q} \to \mathbb{Q}$ as $P^tQ = (P^tQ_1, \ldots, P^tQ_N)$, where;

$$P^tQ_i(s, a) = r_i(s, a)(t) + \gamma \pi^* (s') Q_i (s'), \tag{33}$$

where $s'$ is the state of the next time step and $\pi^*(s')$ is the NE at the state $s'$, which can also be described as the NE for the stage game $(Q_1(s'), \ldots, Q_N(s'))$.

*Lemma 2:* For any $s \in S$, states that the existence of a NE point $(\pi_1^*, \ldots, \pi_n^*)$ in a state game, with equilibrium payoff $SU_i(\pi_1^*, \ldots, \pi_n^*)$, is equivalent to the existence of Nash equilibrium points in each stage game $(Q_1^*(s), \ldots, Q_n^*(s))$, where the agent's optimal value $SU_i$ in the overall game is linked to its NE payoff in the stage games through the equation:

$$\begin{aligned} Q_1^* (s, a_1, \ldots, a_n) &= r_i (s, a_1, \ldots, a_n) \\ &+ \gamma \sum_{s' \in S} p (s' \mid s, a_1, \ldots, a_n) \\ &\times Q_i^* (s', \pi_1 (s'), \pi_2 (s')), \text{ for } i = 1, 2 \end{aligned} \tag{34}$$

This lemma relates an agent's optimal value in a stochastic game to its Nash Equilibrium in a stage game, stating that if the equilibrium condition is not met, the Q-value will change, indicating a non-equilibrium state.

*Assumption 3:* Every stage game $(Q_1^t, \ldots, Q_N^t)$, for all $t$ and all $s$, has a global optimal point or saddle point, and agents' payoffs in this equilibrium are used to update their Q-functions.

In our model, the Stackelberg game solutions provide equilibria for leaders and followers in each stage game, assured by *Proof 2* and Theorem 1, making Assumption 3 attainable. Lemma 2 establishes that a NE solution for converged Q-values corresponds to a NE point for the overall Stackelberg game. Under Assumptions 1–3, the Q-value $Q$ for all MSP agents updated by (31) converges to Nash Q-value which also serves as the Stackelberg game solution.

## V. Performance Evaluation

In this section, we evaluate the performance of our proposed MADQN-based pricing scheme via simulations. All simulations are implemented in a Python 3.8 environment with Keras and Tensorflow python libraries, installed on a computer with a 3.9 GHz AMD Ryzen 7 3800X CPU, and 16 GB RAM running on a Windows 10 operating system. We also install the Matplotlib package to plot and visualize the performance evaluation results. For performance comparison among the different pricing schemes, we use an actual taxi dataset from Didi Chuxing's

TABLE II
KEY SIMULATION PARAMETERS AND SETTINGS

| Setting | Value |
|---|---|
| Number of passenger requests, $M$ | 150 |
| Number of MSPs, $N$ | 2-4 |
| Number of vehicles for MSPs | 40-80 for each |
| Maximum pickup time, $T_{max}$ | 600 $s$ |
| Pickup time range | [60 - 400] $s$ |
| Discrete space for load-state | [0.33, 0.66, 1] |
| Discrete space for price-action | [0.6, 0.7, ..., 2.9, 3.0] *CNY* |
| Cost per unit mileage for MSP, $c_i$ | 0.5 *CNY* |
| Replay memory size | 2000 |
| Mini-batch sample size | 32 |
| Learning rate | 0.005 |
| Exploration probability | 0.9 |
| Discount factor | 0.9 |
| Single episode | 150 epochs |
| Simulation time | 700 episodes |

GAIA Initiative [35]. The dataset includes taxi orders in the downtown of Chengdu, a city in southwest China. Specifically, we utilize a total of 53899 taxi orders in a day, from the entire dataset. Fig. 2(a) and (b) show the data distribution of the Didi Chuxing order dataset of November 1, 2016, and November 2, 2016, respectively. The number of orders is recorded at 5 minutes intervals for a total of 13 hours each day, i.e., from 8 am to 9 pm. From the analysis of the data on the two different days, it is observed that the data for each day and time is different. Therefore, it is meaningful to fit the model with data from one day and test it with data from another day. We generate 150 passenger requests at random and set the number of MSPs and shared vehicles in a range of 2–4 and 40–80, respectively. Each episode has 150 decision epochs, with one decision epoch translated into the decision to select an MSP-winner to assign a vehicle to the one passenger order in the epoch, i.e., 1 epoch=1 successful passenger request. Since we randomly select only 150 orders at an episode (150 epochs), we believe the number of taxi orders selected from the main dataset is enough to achieve the desired results for this study.

The said dataset is used to initialize the distribution of passenger requests and MSP vehicles in the spatial dimension by dividing the city area into 10 x 10 grids, each representing an urban area of 250 m x 250 m. The traveling time of vehicles between the grids represents the pickup delay, and the pickup time statistics scales from 60 s to 400 s. The maximum pickup time $T_{\max}$ is set to 600 s, and the cost of fuel consumption per unit mileage $c_i$ of an MSP is set to 0.5. To speed up convergence, we discretize the load $\vartheta$ to denote light load, medium load, and heavy load scenarios as [0.33, 0.66, 1]. Since the MSPs dynamically adjust their unit prices to reflect the supply-demand relationship and market competitiveness, we define a set of pricing strategies as [0.6, 0.7, ..., 2.9, 3.0]. The average time for each operator to make pricing decisions is 0.00022 s, and the average time required to complete each transaction is approximately 0.003 s. For the MADQN algorithm implementation, we set the replay memory size, the mini-batch sample size, and the learning rate to 2000, 32, and 0.005, respectively. We also set the exploration rate of the agent to 0.9 and the discount factor to 0.9. The key parameters for our simulation are summarized in Table II.

### A. Benchmarks and Performance Metrics

To evaluate the performance of the MADQN-based pricing algorithm, we set up two experimental scenarios, namely, the one-in-many scenario and the balanced scenario. In the one-in-many scenario, one of the MSPs has a DQN agent deployed to perform intelligent and dynamic pricing, while the other MSPs resort to random pricing (unintelligent pricing). Thus, we refer to our proposed algorithm under the one-in-many scenario as the DQN-based pricing algorithm. In the balanced scenario, each of the MSPs has a DQN agent deployed for intelligent and dynamic pricing, i.e., the MADQN-based pricing algorithm.

With respect to IPS, CPS, and MADRL, we propose dynamic pricing methods based on DQN as IPS-DQN and CPS-DQN. For IPS-DQN, the pricing strategy of the agent is based solely on service demand, while for CPS-DQN, the pricing strategy of the agent depends on service demand and the pricing strategies of other agents. We compare the performance of IPS-DQN and CPS-DQN with the following benchmark pricing schemes:

*1) Q-Learning (QL) Based Pricing:* We deploy Q-learning to learn the optimal pricing strategy, where an agent relies on a Q-table to store past learning experiences (Q-values) to set optimal prices [36].

*2) Premium Pricing:* In premium pricing, the MSPs set high unit prices to obtain higher profits. In our study, we increase the unit price at the NE point by 10% to depict premium unit price [26].

*3) UnderCut Pricing:* In underCut pricing, the MSPs reduce their unit prices to attract more passengers. We decrease the unit price at the NE point by 10% to depict underCut pricing [26].

An MSP may perform random pricing by setting its unit price as a random value between the premium unit price and the underCut unit price. We use random pricing to evaluate the performance of the above-mentioned pricing methods based on the following metrics:

1) *Transaction profit:* The transaction profit refers to the amount of "money" earned by the MSP when it completes a passenger's ride request. We define the profit as the difference between the MSP's total revenue and cost of completing a ride request.
2) *MSP cumulative profit:* The cumulative profit of an MSP is its profit earned on all vehicle dispatching service transactions.
3) *Utility of passenger:* The utility of passenger $j$ is defined as its benefit $BU_j$ in each transaction, as shown in (6).
4) *Social welfare (SW):* The social welfare is the sum of the MSPs' profit and passengers' benefit in each vehicle dispatching service transaction [37].

### B. Convergence Analysis

In this simulation, we evaluate the convergence of the proposed MADQN-based pricing algorithm on the utility of MSPs under IPS and CPS. In addition, we evaluate the price per unit mileage set by different MSPs with increasing passenger requests under both IPS and CPS. We set up 3 MSPs as MSP(0), MSP(1), and MSP(2), in the balanced scenario (all MSPs have agents deployed on them), and allocate 80 vehicles to each of

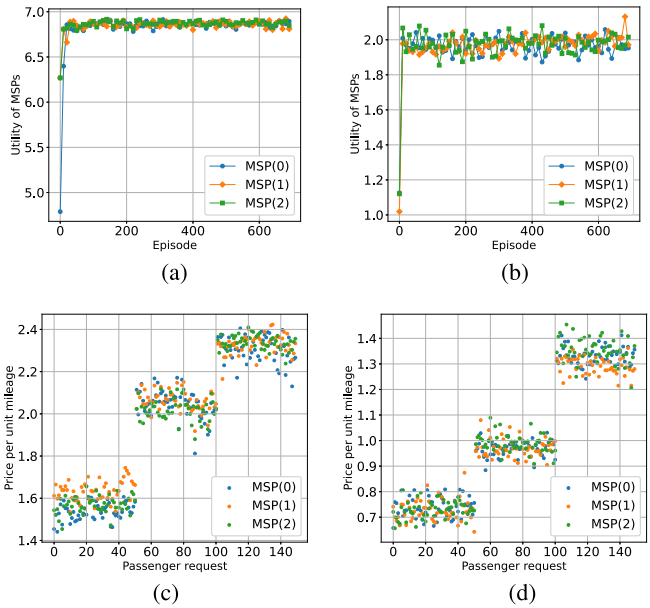



Fig. 3. Convergence analysis and pricing strategy. (a) Convergence under IPS. (b) Convergence under CPS. (c) Pricing under IPS. (d) Pricing under CPS.

them. We train the network for 700 episodes with 150 decision epochs in each episode and use the statistical average of the utility of MSPs over 150 trading transactions as the convergence evaluation metric. Fig. 3(a) and (b) show the convergence on the utility of MSPs under IPS and CPS, respectively. From Fig. 3(a) and (b), we observe that convergence is achieved under both IPS and CPS at approximately the $100th$ episode and utility levels of approximately 6.8 and 2.0, respectively. The difference in the utility levels is because the MSPs under IPS selfishly set their unit prices without considering the unit prices set by their counterparts. By setting their prices independently, the MSPs under IPS tend to achieve more utility than MSPs under CPS, who adjust their unit prices (probably lower prices than their competitors) to be competitive and earn a reasonable market share. This explains our observation of the convergence under CPS experiencing fluctuations even at the $700th$ episode. The reason is that under CPS, there is competition among the MSPs for the largest market share, and each MSP adjusts its unit price to suit the service demand of the passengers. We can conclude that our proposed MADQN-based pricing algorithm can converge to the optimal pricing strategy under IPS and CPS, although the stability under IPS is better than that of CPS.

While observing the convergence of the proposed MADQN-based pricing algorithm, we compute the average price per unit mileage of each MSP in the last 50 episodes over an increasing number of passenger requests. We use discretized values of the load as 0.33, 0.66, and 1, to represent the number of passenger requests at light load, medium load, and heavy load scenarios, respectively. Fig. 3(c) and (d) show the price per unit mileage of the three MSPs with an increasing number of passenger requests under IPS and CPS, respectively. From Fig. 3(c), we observe that the price per unit mileage of all three MSPs increases with an increasing number of passenger requests, under IPS. It is noteworthy that under IPS, the MSPs behave as if they are the only SPs in the MaaS environment. They set their individual unit prices independently and not considering the unit prices set by the other MSPs. At light load, say with about 30 passenger requests, MSP(1) sets the highest unit price, followed by MSP(2) and then MSP(0) in descending order. At medium load, say with 80 passenger requests, the unit prices of all MSPs seem to be in the same range. At heavy load, say with 140 passenger requests, MSP(2) and MSP(1) set higher prices most of the time than MSP(0). The reason for this trend is due to the following: i) As the number of passenger requests increases, there are more requests to serve, and the MSPs decide to increase their unit prices which is why the unit prices increase with an increasing number of passenger requests ii) Under each load scenario, each MSP considers its vehicle load (supply-demand relationship) and sets a unit price. It is likely that an MSP will set higher unit prices when demand is higher than supply and vice versa. Although the MSPs adjust their unit prices, each MSP adjusts its price independently without considering the prices of the other MSPs. Fig. 3(d) follows a similar trend to that of Fig. 3(c), i.e., the price per unit mileage of all three MSPs increases with an increasing number of passenger requests. However, the price per unit mileage set by the MSPs under CPS is lower than that set by the MSPs under IPS. At heavy load, say with 120 passenger requests, the price per unit mileage of all three MSPs is approximately 1.3–1.4 *CNY* each under CPS, which is lower than about 2.3–2.4 *CNY* under IPS. The reason for the trend is similar to that of the results in Fig. 3(a) and (b). We also observe that under CPS, there is no significant difference in the prices set by the different MSPs. The reason for this observation is that all MSPs are equipped with DQN agents and compete to earn the largest market share; hence, they tend to adjust their unit prices considering the prices set by their competitors. We conclude that the MSPs' price per unit mileage under CPS is lower than that under IPS due to the highly competitive nature among the MSPs.

### C. Performance Comparison: Pricing Algorithms

In this simulation, we compare the performance of IPS-DQN, IPS-QL, premium, and underCut pricing schemes with random pricing, in terms of cumulative profit of MSPs and the number of available vehicles, as the number of passenger requests increases. We also compare the performance of the four pricing schemes in terms of passenger utility and social welfare under light, medium, and heavy load scenarios. We set up 3 MSPs for one-in-many scenario (one MSP is equipped with a DQN agent, and the others resort to random pricing), and allocate 75 vehicles to each of them. We run four separate experiments, each comparing the performance of one MSP equipped with intelligent pricing and two MSPs equipped with random pricing. For simplicity in the presentation of results, we show the result of only the MSP with intelligent pricing in each experiment for performance comparison among the intelligent pricing algorithms. That is, we do not show the results of the two MSPs equipped with random pricing in each experiment. Fig. 4(a) and (b) show the evaluation results for the cumulative profit of MSPs and the number of available vehicles, respectively with an increasing number of passenger requests. Fig. 4(c) and (d) show the results

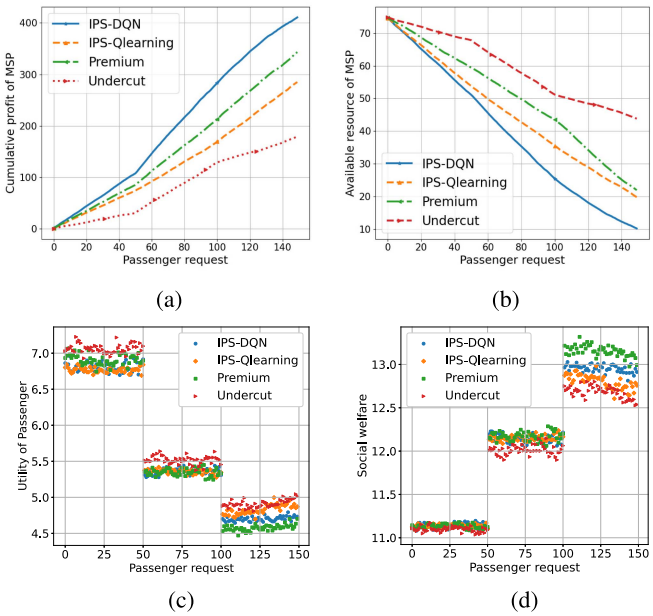



Fig. 4. Performance comparison of four algorithms. (a) Cumulative profit of MSPs. (b) Number of available vehicles. (c) Utility of passenger. (d) SW for each transaction.

for passenger utility and social welfare respectively, for the four pricing schemes with varying load scenarios.

From Fig. 4(a), we observe that the MSPs with IPS-DQN and IPS-QL achieve the highest cumulative profits compared with the MSPs with premium and underCut pricing. For instance, the cumulative profits of IPS-DQN and IPS-QL increase rapidly with increasing passenger requests, eventually reaching profit levels of more than 400 *CNY* and 389 *CNY*, respectively, while the cumulative profits of premium and underCut pricing reach approximately 320 *CNY* and 150 *CNY*, respectively. This is so because the MSPs equipped with DQN agents under IPS can observe the environment and set reasonable prices with an increasing number of passenger requests. The MSP under premium pricing sets higher prices, which demotivates passengers from acquiring services from them. The MSP with underCut pricing sets very low prices to earn more market share, but their profit levels are minimized. The results shown in Fig. 4(b) depict a reciprocating trend of Fig. 4(a). With 140 passenger requests for instance, the number of available vehicles for IPS-DQN, IPS-QL, Premium, and underCut pricing is approximately 4, 18, 25, and 50, respectively. This is because MSPs with IPS-DQN and IPS-QL achieve more successful passenger requests than the MSPs equipped with random pricing, thereby dispatching more vehicles. The MSPs with premium and underCut pricing on the other hand achieve fewer successful passenger requests, losing most of the requests to their counterparts equipped with random pricing. An increase in the cumulative profit of MSPs in Fig. 4(a) means that more passengers are being served by such MSPs and that the number of available vehicles decreases in Fig. 4(b) and vice versa. Comparing the results in Fig. 4(a) and (b) to a CPS scenario, it is anticipated that the MSPs under CPS-DQN and CPS-QL will achieve lower cumulative profits because each MSP under CPS not only sets their unit price to match supply and demand but also strives to increase its competitiveness and eventually obtain a reasonable market share. That is, MSPs under CPS are bound to adjust their unit prices to maximize their competitiveness and achieve a large market share at the same time. This is consistent with the results shown in Fig. 3.

In Fig. 4(c) and (d), the red, green, orange, and blue points in the clusters represent the number of successful passenger requests obtained by MSPs with underCut, premium, IPS-QL, and IPS-DQN, respectively. From Fig. 4(c), we observe that the passenger utilities of all four pricing schemes decrease with increasing passenger requests in general. Specifically, underCut pricing achieves the highest passenger utility levels compared with the other pricing schemes, as premium pricing achieves the worst results at heavy loads. The reason is that the low prices set in underCut pricing increase the passengers' utilities and service demand towards it, while the high prices set in premium pricing discourage passengers from trading with such MSPs since they are likely to record lower utilities. The passenger utilities of IPS-DQN and IPS-QL are between that of underCut and premium pricing schemes due to their ability to intelligently adjust their prices to attract more passengers. From Fig. 4(d), we observe that there is not much difference in the results of the four pricing schemes under light load and medium load. Notably, the premium pricing achieves the highest social welfare at all load scenarios, followed by IPS-DQN, IPS-QL, and undercut pricing in that order, which is evident under the heavy load scenario. Although the MSP with premium pricing achieves the lowest passenger utility in most cases in Fig. 4(c), it achieves more successful passenger requests than the other MSP with underCut pricing. This explains why the MSP with premium pricing has fewer available vehicles than the MSP with underCut pricing in Fig. 4(b). Again, the MSPs with IPS-DQN and IPS-QL achieve moderate social welfare levels compared to those with premium and undercut pricing. We can conclude that MSPs with IPS-DQN and IPS-QL can intelligently set their unit prices to match passenger requests, earning more profits and keeping passenger utility and social welfare at acceptable levels. However, the MSP with IPS-DQN achieves more successful passenger requests than the MSP with IPS-QL.

### D. Performance Comparison: IPS vs CPS

In this subsection, we compare the performance of our proposed MADQN-based pricing algorithm under IPS and CPS in terms of MSPs' unit price for each transaction and passengers' service demand. We consider the balanced scenario where all MSPs are equipped with DQN agents. Under both pricing schemes, we set up 2 or 4 MSPs with 80 or 40 vehicles assigned to them, respectively, as follows; IPS with 2 MSPs (IPS-2 MSP), CPS with 2 MSPs (CPS-2 MSP), IPS with 4 MSPs (IPS-4 MSP), and CPS with 4 MSPs (CPS-4 MSP). Fig. 5(a) and (b) show the performance of IPS and CPS in terms of MSPs' pricing and passengers' service demand, with increasing passenger requests, respectively.

From Fig. 5(a), we observe that the unit price of IPS is higher than that of CPS. With 40 passenger requests, the unit price

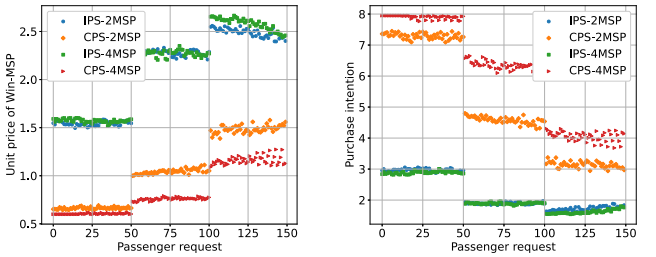



Fig. 5. Transactions of MSPs and passengers. (a) Price for each transaction. (b) Service demand of passenger.

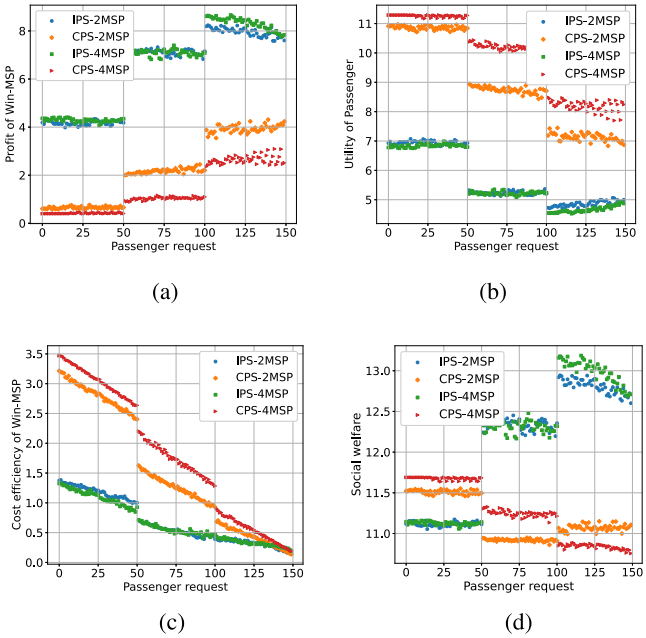



Fig. 6. Performance comparison of IPS and CPS. (a) Win-MSP's profit for each transaction. (b) Passenger's utility for each transaction. (c) Cost efficiency of win-MSP. (d) SW for each transaction.

of IPS-4 MSPs and IPS-2 MSPs are approximately 1.6 CNY and 1.5 CNY, while the unit price of CPS-2 MSPs and CPS-4 MSPs are approximately 0.7 CNY and 0.6 CNY. This is because the MSPs under IPS set their unit price without considering the prices of the other MSPs, while MSPs under CPS compete with one another for the largest market share and are compelled to set lower unit prices than their market competitors. As the number of passenger requests increases, the unit price of all four settings increases. The unit price under CPS-4 MSP is the lowest with the reason being that as the number of competitors increases, each MSP tends to further lower its unit price. On the contrary, Fig. 5(b) shows that the service demand of passengers under all four settings decreases with increasing passenger requests, with IPS-4 MSP achieving the lowest service demand and CPS-4 MSP achieving the highest. Specifically, the trend in Fig. 5(b) is the reverse of that in Fig. 5(a), which aligns with realistic expectations. That is, when the MSPs set lower unit prices, the passengers issue more ride requests. However, when the MSPs set high unit prices, the service demand of the passengers reduces. Since the MSPs with IPS set higher prices than the MSPs with CPS throughout the simulation, we observe that the service demand of the passengers towards the former is lower than that of the latter. We can conclude that CPS is able to decide reasonable unit prices to match higher passengers' service demands. The more the number of competitors, the better the results.

Fig. 6(a), (b), (c), and (d) show the performance of IPS and CPS in terms of the winner-MSP's profit, the passenger's utility, the cost efficiency of the winner-MSP, and the social welfare of each transaction with an increasing number of passenger requests. We keep the settings of MSPs in the previous simulation under both pricing schemes, considering the balanced scenario. From Fig. 6(a) and (b), we observe that the trend is consistent with the results shown in Fig. 5(a) and (b), respectively. That is, the profit of winner-MSP under IPS is higher than that under CPS while the passenger utility under CPS is higher than that of IPS. Under CPS, we observe that the profit of CPS-2 MSP is higher than the profit of CPS-4 MSP. The profit of winner-MSP under IPS-2 MSP and IPS-4 MSP increase from approximately 4.0 *CNY* and 4.2 *CNY* to 8.0 *CNY* and 8.5 *CNY*, respectively, and the profit of winner-MSP under CPS-2 MSP and CPS-4 MSP increase from about 0.8 *CNY* and 0.5 *CNY* to 4.0 *CNY* and 2.8 *CNY*, respectively. The passenger utility of CPS-4 MSP is higher than the passenger utility of CPS-2 MSP, while the result of IPS-2 MSP is higher than that of IPS-4 MSP. The reason for the lower profit of winner-MSP under CPS is that CPS ensures market competitiveness among MSPs. This is achieved by MSPs setting lower prices for their services to improve passengers' willingness to request services. Even though this leads to a slight reduction in the profit of the MSP, passenger benefit is enhanced ensuring high service satisfaction.

From Fig. 6(c), we see that the cost efficiency of winner-MSP decreases with an increasing number of passenger requests under both IPS and CPS. Specifically, the cost efficiency under CPS is better than that under IPS, with CPS-4 MSP achieving the best results. This is because the more the number of competitors, the lower the unit price set by MSPs and the better the service quality. Additionally, stronger competition achieves better results. From Fig. 6(d), we observe that at light load, the social welfare under CPS is higher than that of IPS. Under medium and heavy load scenarios, the social welfare under IPS is higher than that of CPS. The reason for this trend is that at light load, the passengers purchase more resources from the MSPs under CPS. However, at medium and heavy loads, the MSPs under IPS set far higher prices to earn higher profits from the few successful transactions. We can conclude that CPS achieves a better balance between profits and passenger utility while maximizing cost efficiency per transaction. However, the social welfare for each transaction is minimized due to the adjustment of prices to suit passenger service demands.

### *E. Performance Comparison: Profit and Number of Vehicles*

In this simulation, we evaluate the impact of varying number of vehicle configurations on the total profit of MSPs. The sum

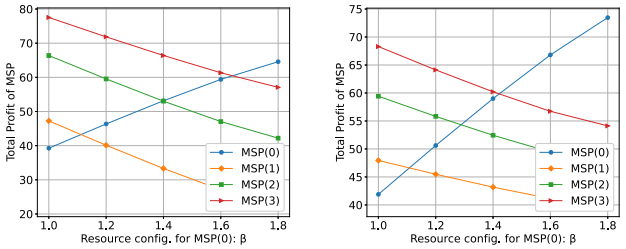



Fig. 7. Profit of MSP with $\beta$ number of vehicles. (a) Balanced scenario. (b) One-in-many scenario.

of vehicles of the 4 MSPs is set to 160, which is unchanged throughout the simulation. We set up five sets of experimental scenarios with 150 passenger requests and 4 MSPs, whose number of vehicle configurations are [$\beta$: 1.2: 1.4: 1.6]. We consider the same vehicle configurations for both balanced and one-in-many scenarios, where the number of vehicle configurations of MSP(1)-MSP(3) are [1.2 : 1.4 : 1.6] and MSP(0) increases its initial number of vehicle configuration by $\beta = [1.0, 1.2, 1.4, 1.6, 1.8]$. Fig. 7(a) and (b) show the performance of total profit vs. number of vehicles in balanced and one-in-many scenarios, respectively.

From Fig. 7(a), we observe that as the number of vehicles increases, the total profit of MSP(0) increases while the profits of MSP(1), MSP(2), and MSP(3) decrease. The profit of MSP(0) increases from about 40 *CNY* to 65 *CNY*, while that of MSP(1), MSP(2), and MSP(3) decrease from approximately 48 *CNY*, 67 *CNY*, and 78 *CNY* to 20 *CNY*, 42 *CNY*, and 58 *CNY*, respectively. This is because MSP(0) is able to increase its initial vehicle configuration, while the other MSPs' vehicle configurations decrease. A similar trend is seen under the one-in-many scenario in Fig. 7(b), where the profit of MSP(0) increases as its vehicle configuration is increased. However, the profit of MSP(0) under the one-in-many scenario is higher than that of the balanced scenario. The reason is that under the one-in-many scenario, only MSP(0) is equipped with a DQN agent and that, it is able to intelligently decide its unit price to earn more profit than the other MSPs with random pricing. In the balanced scenario, all MSPs are equipped with DQN agents to be able to earn more profits by intelligently setting their unit prices. We can conclude that as the initial number of vehicles of an MSP increases, its profit also increases.

## VI. Conclusion

This article investigated the vehicle dispatching service pricing and service demand problem in the MaaS market, taking into account the interaction modeling between MSPs and passengers who request ride-hailing services for mobility. We modeled the service pricing and demand problem of MSPs and passengers, respectively, as a two-stage Stackelberg game, where MSPs act as leaders and passengers as followers. The MA provides a passenger with the unit price for vehicle usage determined by an MSP in the first stage, and the passenger then determines its service demand in the second stage based on the unit price. Our proposed native DQN-based MADRL algorithm was then deployed to obtain the NE, which achieves optimal unit pricing and demand solution for MSPs and passengers, respectively. According to the simulation results, the proposed algorithm outperforms other benchmark schemes under IPS and CPS to effectively raise MSP revenues, while protecting the passenger benefits. Furthermore, the CPS algorithm boosts MSPs' market attractiveness and long-term benefits, encouraging MSPs to participate in vehicle dispatching service provisioning in a competitive MaaS market. Future work will investigate a highly competitive game analysis between MSPs and passengers in the MaaS market. Due to the trustless trading environment, the deployment of blockchain for secure and transparent trading will also be studied.

## References

[1] Y. Z. Wong, D. A. Hensher, and C. Mulley, "Mobility as a service (MaaS): Charting a future context," *Transp. Res. Part A: Policy Pract.*, vol. 131, pp. 5–19, 2020.

[2] Z. Liu, J. Li, and K. Wu, "Context-aware taxi dispatching at city-scale using deep reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 1996–2009, Mar. 2022.

[3] Z. Xu et al., "Large-scale order dispatch in on-demand ride-hailing platforms: A learning and planning approach," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 905–913, doi: 10.1145/3219819.3219824.

[4] M. Battifarano and Z. S. Qian, "Predicting real-time surge pricing of ride-sourcing companies," *Transp. Res. Part C: Emerg. Technol.*, vol. 107, pp. 444–462, 2019.

[5] B. Turan and M. Alizadeh, "Competition in electric autonomous mobility on demand systems," *IEEE Trans. Control Netw. Syst.*, vol. 9, no. 1, pp. 295–307, Mar. 2022.

[6] Y. Cheng, X. Deng, and M. Zhang, "Two-tier sharing in electric vehicle service market," *IEEE Trans. Cloud Comput.*, vol. 10, no. 1, pp. 724–735, Jan.–Mar. 2022.

[7] X. Wang, F. He, H. Yang, and H. Oliver Gao, "Pricing strategies for a taxi-hailing platform," *Transp. Res. Part E: Logistics Transp. Rev.*, vol. 93, pp. 212–231, 2016.

[8] J. C. Castillo, D. Knoepfle, and G. Weyl, "Surge pricing solves the wild goose chase," in *Proc. ACM Conf. Econ. Comput.*, 2017, pp. 241–242.

[9] L. Sun, R. H. Teunter, M. Z. Babai, and G. Hua, "Optimal pricing for ride-sourcing platforms," *Eur. J. Oper. Res.*, vol. 278, no. 3, pp. 783–795, 2019.

[10] R. Iacobucci and J.-D. Schmöcker, "Dynamic pricing for ride-hailing services considering relocation and mode choice," in *Proc. IEEE 7th Int. Conf. Models Technol. Intell. Transp. Syst.*, 2021, pp. 1–6.

[11] F. He, X. Wang, X. Lin, and X. Tang, "Pricing and penalty/compensation strategies of a taxi-hailing platform," *Transp. Res. Part C: Emerg. Technol.*, vol. 86, pp. 263–279, 2018.

[12] M. Chen, D. Zhao, Y. Gong, and Y. Rekik, "An on-demand service platform with self-scheduling capacity: Uniform versus multiplier-based pricing," *Int. J. Prod. Econ.*, vol. 243, 2022, Art. no. 108329.

[13] H. M. Amar and O. A. Basir, "A game theoretic solution for the territory sharing problem in social taxi networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 7, pp. 2114–2124, Jul. 2018.

[14] M. Adil, M. P. Mahmud, A. Z. Kouzani, and S. Khoo, "Energy trading among electric vehicles based on stackelberg approaches: A. review," *Sustain. Cities Soc.*, vol. 75, 2021, Art. no. 103199. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210670721004777

[15] G. Zardini, N. Lanzetti, L. Guerrini, E. Frazzoli, and F. Dörfler, "Game theory to study interactions between mobility stakeholders," in *Proc. IEEE Int. Intell. Transp. Syst. Conf.*, 2021, pp. 2054–2061.

[16] Y. Lu, Y. Liang, Z. Ding, Q. Wu, T. Ding, and W.-J. Lee, "Deep reinforcement learning-based charging pricing for autonomous mobility-on-demand system," *IEEE Trans. Smart Grid*, vol. 13, no. 2, pp. 1412–1426, 2022.

[17] C. Liu, C.-X. Chen, and C. Chen, "META: A city-wide taxi repositioning framework based on multi-agent reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 13890–13895, Aug. 2022.

[18] D. Shi et al., "Deep Q-network based route scheduling for transportation network company vehicles," in *Proc. IEEE Glob. Commun. Conf.*, 2018, pp. 1–7.
[19] M. Li et al., "Efficient ridesharing order dispatching with mean field multi-agent reinforcement learning," in *Proc. World Wide Web Conf.*, 2019, pp. 983–994.
[20] A. Karinsalo and K. Halunen, "Smart contracts for a mobility-as-a-service ecosystem," in *Proc. IEEE Int. Conf. Softw. Qual., Rel. Secur. Companion*, 2018, pp. 135–138.
[21] P. K. Sharma, S. Singh, Y.-S. Jeong, and J. H. Park, "DistBlockNet: A distributed blockchains-based secure SDN architecture for IoT networks," *IEEE Commun. Mag.*, vol. 55, no. 9, pp. 78–85, Sep. 2017.
[22] Y. Jiao, P. Wang, S. Feng, and D. Niyato, "Profit maximization mechanism and data management for data analytics services," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 2001–2014, Jun. 2018.
[23] T. Strutz, *Data Fitting and Uncertainty: A. Practical Introduction to Weighted Least Squares and Beyond*. Wiesbaden, Germany: Vieweg+Teubner, 2010.
[24] Y. Zhong, T. Yang, B. Cao, and T. Cheng, "On-demand ride-hailing platforms in competition with the taxi industry: Pricing strategies and government supervision," *Int. J. Prod. Econ.*, vol. 243, 2022, Art. no. 108301.
[25] D. Aussel and A. Svensson, *A Short State of the Art on Multi-Leader-Follower Games*. Cham, Switzerland: Springer, 2020, pp. 53–76.
[26] T. D. Tran and L. B. Le, "Resource allocation for multi-tenant network slicing: A multi-leader multi-follower stackelberg game approach," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8886–8899, Aug. 2020.
[27] K. Liu, X. Qiu, W. Chen, X. Chen, and Z. Zheng, "Optimal pricing mechanism for data market in blockchain-enhanced Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9748–9761, Dec. 2019.
[28] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: A Bradford Book, 2018.
[29] V. Mnih et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
[30] J. Hu and M. P. Wellman, "Nash Q-learning for general-sum stochastic games," *J. Mach. Learn. Res.*, vol. 4, pp. 1039–1069, May 2022.
[31] G. Tesauro, "Extending Q-learning to general adaptive multi-agent systems," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2003, pp. 871–878.
[32] Y. Yang, R. Luo, M. Li, M. Zhou, W. Zhang, and J. Wang, "Mean field multi-agent reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 5571–5580.
[33] T. Jaakkola, M. Jordan, and S. Singh, "Convergence of stochastic iterative dynamic programming algorithms," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 1993, pp. 703–710.
[34] C. Szepesvári and M. L. Littman, "A unified analysis of value-function-based reinforcement-learning algorithms," *Neural Comput.*, vol. 11, no. 8, pp. 2017–2060, 1999.
[35] "Gaia initiative," Accessed: May 5, 2022. [Online]. Available: https://outreach.didichuxing.com/research/opendata/en/
[36] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3, pp. 279–292, May 1992, doi: 10.1007/BF00992698.
[37] Y. Jiao, P. Wang, D. Niyato, and K. Suankaewmanee, "Auction mechanisms in cloud/fog computing resource allocation for public blockchain networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 9, pp. 1975–1989, Sep. 2019.

**Guolin Sun** (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in communication and information systems from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2000, 2003, and 2005 respectively. He has got eight years industrial work experiences on wireless research and development for LTE, Wi-Fi, Internet of Things, cognitive radio, localization, and navigation. Before he joined the UESTC, as an Associate Professor in August 2012, he was with Huawei Technologies Sweden. He has filed omore than 40 patents, and authored or coauthored more than 60 scientific conference and journal papers, acts as TPC Member of conferences. His research interests include blockchain network, vehicular networks, resource optimization, and network slicing.

**Gordon Owusu Boateng** (Member, IEEE) received the bachelor's degree in telecommunications engineering from the Kwame Nkrumah University of Science and Technology, Kumasi, Ghana, in 2014, and the M.Eng. and Ph.D. degrees in computer science and technology from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2019 and 2023, respectively. He is currently a Postdoctoral Researcher with Hybrid Positioning Research Group, UESTC. From 2014 to 2016, he worked under sub-contracts for Ericsson (Ghana) and TIGO (Ghana). Till now, he has authored or coauthored more than 25 scientific journal and conference papers. His research interests include 5G/6G wireless networks, blockchain, reinforcement learning, vehicular networks, target positioning, and automated valet parking.

**Kai Liu** received the B.Sc. degree from School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, China, in 2018. Since 2019, he has been working toward the M.Sc. degree in computer science from the University of Electronic Science and Technology of China (UESTC), Chengdu, China. He is also a Member of the Mobile Cloud-Net Research Team, UESTC. His research interests include internet of vehicles, resource trading, and deep reinforcement learning.

**Daniel Ayepah-Mensah** received the bachelor's degree in computer engineering from the Kwame Nkrumah University of Science and Technology, Kumasi, Ghana, in 2014, and the master's degree in computer science from the University of Electronic Science and Technology of China, Chengdu, China, where he is currently working toward the Ph.D. degree. His research interests include mobile/cloud computing, 5G wireless networks, artificial intelligence, network virtualization, edge computing, device-to-device communications, blockchain, game theory, and software-defined networking.

**Guisong Liu** received the B.S. degree in mechanics from Xi'an Jiao Tong University, Xi'an, China, in 1995, and the M.S. degree in automatics, and the Ph.D. degree in computer science from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2000 and 2007, respectively. He was a Visiting Scholar with Humbolt University, Berlin, Germany, in 2015. Before 2021, he was a Professor with the School of Computer Science and Engineering, UESTC. He is currently a Professor and the Dean of the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu. He has filed more than 20 patents and authored or coauthored more than 70 scientific conference and journal papers. His research interests include pattern recognition, neural networks, and machine learning.