Reverse-Auction-Based Competitive Order Assignment for Mobile Taxi-Hailing Systems

Hui Zhao[1], Mingjun Xiao[1](✉), Jie Wu[2], An Liu[3], and Baoyi An[1]

[1] School of Computer Science and Technology, Suzhou Institute for Advanced Study, University of Science and Technology of China, Hefei, China
xiaomj@ustc.edu.cn
[2] Department of Computer and Information Sciences, Temple University, Philadelphia, USA
[3] School of Computer Science and Technology, Soochow University, Suzhou, China

Abstract. Mobile Taxi-Hailing (MTH) is one of the most attractive smartphone applications, through which passengers can reserve taxis ahead for their travels so that the taxi service's efficiency can improve significantly. The taxi-hailing order assignment is an important component of MTH systems. Current MTH order assignment mechanisms fall short in flexibility and personalized pricing, resulting in an unsatisfactory service experience. To address this problem, we introduce a Competitive Order Assignment (COA) framework for the MTH systems. The COA framework mainly consists of the Multi-armed-bandit Automatic Valuation (MAV) mechanism and the Reverse-auction-based Order Assignment (ROA) mechanism. The taxis apply the MAV mechanism to automatically generate the transport service valuations for orders. The platform applies the ROA mechanism to complete each round of order assignment. Then, we analyze the online performance of MAV, and prove that ROA satisfies the properties of truthfulness and individual rationality. Finally, we also demonstrate the significant performances of MAV and ROA through extensive simulations on a real trace.

1 Introduction

With the explosive popularity of smartphones, various mobile applications have been developed to make people's lives more convenient. One of the most appealing applications is the Mobile Taxi-Hailing (MTH) system, such as Uber, Didi Chuxing, Lyft, Ola, etc. By using these MTH systems, passengers need not wait for a long time before hailing a taxi, and taxis also will not spend lots of time searching for passengers. Consequently, more and more passengers are willing to use these systems to hail taxis. Statistics show that there have been more than 5 billion Uber trips in 2017 and 15 million daily active riders on average [2].

A typical MTH system consists of a platform residing on the cloud and a collection of passengers and taxi drivers, who have installed the MTH application

G. Li et al. (Eds.): DASFAA 2019, LNCS 11447, pp. 660–677, 2019.
https://doi.org/10.1007/978-3-030-18579-4_39

in their smartphones, as shown in Fig. 1. If a passenger wants to hail a taxi, he/she would generate a taxi-hailing order and send it to the platform via his/her smartphone. The order includes the start position, destination, and so on. On the other hand, each taxi driver would periodically report its state information to the platform, including the taxi's location, whether the taxi is vacant, etc. After receiving orders from the passengers, the platform would assign each order to a vacant taxi. The order assignment is a vital component of the MTH system.

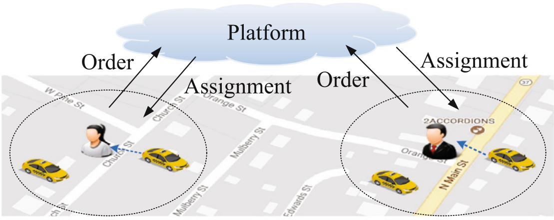


Fig. 1. A typical MTH system

Current MTH systems mainly adopt two types of order assignment strategies. The first is that taxi drivers manually grab the orders publicized by the platform. However, many drivers often complain that they cannot grab any orders most of time, since other drivers might manipulate by using third-party softwares. Another strategy is that the platform directly assigns a vacant taxi to each order according to the distance between them, the reputation of the taxi, etc. However, taxi drivers might be assigned many orders that they do not prefer. In addition, some drivers might wish to compete for their preferred orders by reducing their prices. Nevertheless, this direct assignment strategy has not considered the order competition among taxis and falls short in the personalized pricing requirement. So far, there have been some mechanisms designed for the taxi-hailing order assignment or the taxi dispatch problem, such as [9,10,23]. However, although these mechanisms adopt some complex assignment strategies that aim at different optimization objectives, they have still not involved the personalized pricing and competitive order assignment issues.

To enable taxis to flexibly compete for their preferred taxi-hailing orders with personalized prices, we propose a Competitive Order Assignment (COA) framework for MTH systems. The COA framework mainly includes a Multi-armed-bandit Automatic Valuation (MAV) mechanism and a Reverse-auction-based Order Assignment (ROA) mechanism. On the taxi's side, COA allows each taxi to flexibly set valuations for different taxi-hailing orders. First, each taxi uses some pre-defined rules to determine its preferences for different orders, and then determine multiple candidate mark-up pricing strategies as its service charges based on the preferences. For example, an order whose start position is near or destination is located in familiar areas is a preferred order, for which the taxi may ask a lower service charge. Then, once receiving orders from the platform, each taxi can automatically select a strategy to generate its valuation for each order.

Note that different mark-up strategies mean different rewards. Even the same strategy might lead to different rewards in different spatio-temporal scenarios. Thus, the *automatic valuation* is a complex issue. To solve this problem, we see the automatic valuation as a multi-armed bandit process and design the MAV mechanism, by which each taxi can learn to select appropriate strategies maximizing the cumulative rewards. On the platform side, the order assignment is conducted periodically. We treat each round of order assignment as a *reverse auction* process and design the ROA mechanism. In this way, each taxi can rationally compete for its preferred orders via bidding their truthful valuations. More specifically, our major contributions include:

1. We design a competitive order assignment (COA) framework for MTH systems, in which taxis can automatically generate valuations to compete for the taxi-hailing orders.
2. We see the automatic valuation in COA as a multi-armed bandit process, and propose the MAV mechanism. MAV let each taxi automatically select appropriate mark-up pricing strategies for arriving orders to maximize the cumulative rewards. Further, we analyze the online performance of MAV.
3. We model the competitive order assignment in COA as a series of reverse auction processes, and propose the ROA mechanism, including the winner selection and the payment computation. Moreover, we prove that the ROA mechanism is truthful and individually rational.
4. We conduct extensive simulations on a real trace to verify the significant performances of the proposed ROA and MAV mechanisms.

The remainder of the paper is organized as follows. we introduce the COA framework and problem formalization in Sect. 2. The ROA and MAV mechanisms are proposed in Sect. 3. The theoretical analyses are presented in Sect. 4. In Sect. 5, we evaluate the performances of MAV and ROA. After reviewing the related works in Sect. 6, we conclude the paper in Sect. 7.

2 Framework and Problem Formalization

2.1 The COA Framework

We consider a typical MTH system, including a platform in a cloud, a set of taxis registered in the platform and lots of passengers.

Definition 1 (Taxi-hailing Order). A taxi-hailing order is defined as $o_i = \langle startTime, startLoc, Des\rangle$, where $startTime$, $startLoc$, and Des are the start time, start location, and destination of the corresponding trip, respectively. Moreover, all orders have a common Time-to-Live (TTL). If an order has not been assigned to a taxi during its TTL, it will become invalid.

Definition 2 (Taxi Driver). A taxi v_j is described by its state information: $s_j = \langle isVacant,\ startLoc,\ startTime\rangle$, where $isVacant$ is a boolean indicating whether the taxi is vacant. If $isVacant$ is $true$, it means that the taxi can provide the transport service immediately. Then, *startLoc* and *startTime* are the current location and time of the taxi, respectively. Otherwise, the taxi is carrying passengers. In this case, *startLoc* and *startTime* are the destination and arrival time of the current trip, respectively. In addition, to achieve personalized pricing, each taxi v_j determines its preferences on providing transport services by using some simple pre-defined rules, and then determines multiple candidate mark-up pricing strategies $\mathcal{K}_j = \{1_j, \cdots, K_j\}$ according to the preferences.

The platform continuously receives the taxi-hailing orders from passengers to form an order list $\mathcal{O}$, and receives the real-time state information from taxis to form a taxi state list $\mathcal{S}$. Then, by comparing the *startTime* and *startLoc* values of $s_j \in \mathcal{S}$ and $o_i \in \mathcal{O}$, the platform will know whether taxi v_j can arrive at the start location of order o_i in time. Therefore, the platform can determine the taxis that can provide the transport service to order o_i, denoted by $\mathcal{V}_i$.

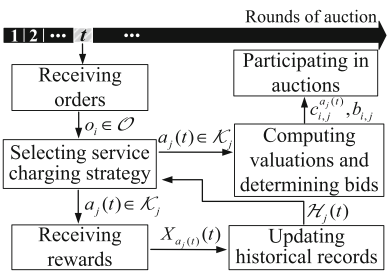


Fig. 2. The automatic valuation process.

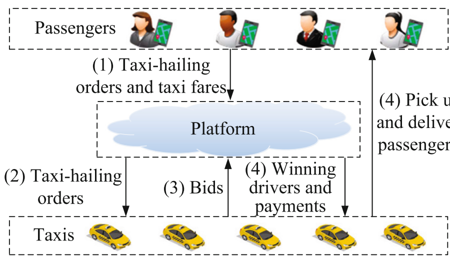


Fig. 3. The reverse auction process.

Based on the above descriptions, we design a Competitive Order Assignment (COA) framework for the MTH system. In the COA framework, taxis set their valuations for arriving taxi-hailing orders based on an automatic valuation mechanism. The platform completes the order assignment via a reverse auction mechanism. The order assignment is conducted periodically, and the period equals to the TTL of orders. At the beginning of each period, the platform conducts a round of assignment for the orders that it has received.

2.2 Automatic Valuation Problem Formalization

After receiving taxi-hailing orders from the platform, each taxi selects the mark-up pricing strategy and further generates the valuation for each order automatically. Since different mark-up strategies mean different rewards over different spatio-temporal scenarios, to select appropriate strategies is a complex issue. Therefore, we formalize the automatic valuation as a multi-armed bandit process. A typical multi-armed bandit process consists of a slot machine with multiple arms, each of which is associated with a reward drawn from an initially unknown distribution. A player needs to sequentially select the arms via some policies, called *bandit policies*, so as to maximize the cumulative reward [3,13]. Then, the formalization is as follows, which is also illustrated in Fig. 2.

First, we see each taxi v_j as a player, and its mark-up pricing strategies $\mathcal{K}_j$ are the arms to be selected. Each arm $k_j \in \mathcal{K}_j$ is associated with a reward $X_{k_j}(t)$. The reward $X_{k_j}(t)$ is a random variable that is i.i.d. and has unknown probability distribution with a bounded support. Without loss of generality, we assume that $X_{k_j}(t)$ lies within the range $[0,1]$ with a mean μ_{k_j}.

Second, the objective of the automatic valuation is to seek a bandit policy to maximize the cumulative reward. Denote the arm selected in the t-th round as $a_j(t) \in \mathcal{K}_j$. The historical records are $\mathcal{H}_j(t) = \{X_{a_j(1)}(1), \cdots, X_{a_j(t)}(t)\}$ with $\mathcal{H}_j(0) = \emptyset$. Then, a bandit policy $\pi_j = (\pi_j(t)_{t=1}^{\infty})$ is defined as a sequence of maps $\pi_j(t) : \mathcal{H}_j(t-1) \rightarrow \mathcal{K}_j$, which specifies the arm that will be selected under the historical records. Based on this, we define the cumulative reward that taxi v_j has received up to the t-th round under the policy π_j as

$$m_j(t) = \sum\nolimits_{k_j=1}^{K_j} \mu_{k_j} \mathbb{E}[N_{k_j}(t)]\Big|_{\pi_j}. \tag{1}$$

Here, $N_{k_j}(t)$ is the total number of times that the k_j-th arm has been selected up to the t-th round, i.e., $N_{k_j}(t) = \sum_{\tau=1}^{t} \mathbb{1}(a_j(\tau) = k_j)$, where $\mathbb{1}(\cdot)$ is an indicator function which is 1 if $(\cdot)$ is true; otherwise, it equals to 0.

Finally, once selecting an arm $a_j(t)$, the valuation, denoted by $c_{i,j}^{a_j(t)}$, is determined as the sum of a base price and the mark-up strategy $a_j(t)$, in which the base price is the inherent transport cost of the taxi v_j serving order o_i. Then, taxis can keep their valuations to participate in the order assignment process. Meanwhile, the taxi will receive a reward $X_{a_j(t)}(t)$. Specially, when taxi v_j loses the order o_i which the arm $a_j(t)$, the reward $X_{a_j(t)}(t)$ will be 0.

For ease of description, we assume that each taxi deals with one order and selects the arms once in each round. If there are multiple orders for taxi v_j in a round, we can construct multiple virtual taxis and let each of which deal with one order. In addition, since the valuation for each order is the private information of each taxi, the bandit process of each taxi is also independent of others.

2.3 Reverse Auction Problem Formalization

Along with the continuous arrival of taxi-hailing orders, the platform conducts the order assignment based on a reverse auction model. In this model, taxis are seen as the sellers of transport services and the platform holding orders is the buyer. Then, the interactions between passengers and taxis via the platform are described as follows, which is also depicted in Fig. 3.

First, when a passenger wants to start a trip, it submits a taxi-hailing order to the platform. Meanwhile, the passenger will also submit a taxi fare that it is willing to pay for the transport service. Second, the platform receives orders from passengers and publicizes them to taxis. Each order will only be publicized to the taxis that can provide the transport service to it. Third, each taxi receives the orders from the platform. For each received order, taxis determine their true valuations and bids, and then submit all of their bids to the platform.

Here, each bid is not necessarily equal to the corresponding valuation, since each taxi might manipulate the claimed charge. Finally, the platform determines the winners of the auction, computes the payment for each winner, and assigns the corresponding orders.

Consider an arbitrary t-th round of auction, where the set of orders is $\mathcal{O}$, the taxis that can provide the transport service for order $o_i \in \mathcal{O}$ is $\mathcal{V}_i$, and the taxis' valuations and bids are $\{c_{i,j}^{a_j(t)} | o_i \in \mathcal{O}, v_i \in \mathcal{V}_i\}$ and $\{b_{i,j} | o_i \in \mathcal{O}, v_i \in \mathcal{V}_i\}$, respectively. The auction process involves the Winner Selection (WS) problem and the Payment Computation (PC) problem, which are formulated as follows.

First, we consider the optimization objective in each round of order assignment is to maximize the social welfare, defined as follows.

Definition 3. The ***social welfare*** is the total taxi fares of the orders that are assigned to some taxis minus the total valuations of these selected taxis.

Then, we can formalize the WS problem as follows.

Definition 4. The ***Winner Selection (WS)*** problem:

$$Maximize: \quad \sum\nolimits_{o_i \in \mathcal{O}, v_j \in \mathcal{V}_i} \phi_{i,j} z_{i,j} \tag{2}$$

$$Subject\ to: \sum\nolimits_{o_i \in \mathcal{O}} z_{i,j} \leq 1,\ z_{i,j} \in \{0,1\} \tag{3}$$

$$\sum\nolimits_{v_j \in \mathcal{V}} z_{i,j} \leq 1,\ z_{i,j} \in \{0,1\} \tag{4}$$

Table 1. Description of major notations

Variable	Description
$a_j(t)$	the arm that taxi v_j selects in the t-th round of auction
$c_{i,j}^{a_j(t)}, b_{i,j}$	the true valuation and bid of taxi v_j for serving order o_i
r_i	the taxi fare of the order o_i
$N_{k_j}(t)$	the number of times that the k_j-th arm has been selected by taxi v_j up to the t-th round
$X_{k_j}(t)$, μ_{k_j}	the reward that taxi v_j receives from the k_j-th arm in the t-th round, and the mean of its probability distribution
$m_j(t)$	the cumulative reward that taxi v_j receives up to the t-th round
$\hat{\mu}_{k_j}(t)$	the average reward that taxi v_j receives from the k_j-th arm up to the t-th round, i.e., the estimated value of μ_{k_j}
μ_j^*	the mean reward associated with the optimal arm
$\phi_{i,j}$	the value of $r_i - b_{i,j}$

Here, $z_{i,j} = 1$ indicates that taxi v_j is selected as the winner to provide the transport service to order o_i; otherwise, if $z_{i,j} = 0$, order o_i will not be assigned to taxi v_j. Moreover, we define $\phi_{i,j} = r_i - b_{i,j}$. Note that our reverse auction mechanism is truthful, which means that all taxis will always submit the true valuations as their bids. Hence, we can directly assume $b_{i,j} = c_{i,j}^{a_j(t)}$. Then $\sum_{o_i \in \mathcal{O}, v_j \in \mathcal{V}_i} \phi_{i,j} z_{i,j} = \sum_{o_i \in \mathcal{O}, v_j \in \mathcal{V}_i} (r_i - c_{i,j}^{a_j(t)}) z_{i,j}$ is the social welfare. We will prove the truthfulness in Sect. 4, which implies that this assumption holds.

Finally, the PC problem is defined as follows:

Definition 5. The ***Payment Computation (PC)*** problem is how to determine the payment for each winner so that the whole auction mechanism satisfies the truthfulness and the individual rationality.

Definition 6 (Truthfulness). Let $b_{i,j}$ be an arbitrary bid for taxi v_j that wins the order o_i, and $p_{i,j}(b_{i,j})$ is the corresponding payment determined by the payment computation algorithm of an auction mechanism. Then, if $p_{i,j}(b_{i,j}) - c_{i,j}^{a_j(t)} \leq p_{i,j}(c_{i,j}^{a_j(t)}) - c_{i,j}^{a_j(t)}$, we say that the auction mechanism is truthful.

Definition 7 (Individual Rationality). For each winning bid $b_{i,j}$, the corresponding payoff is nonnegative, i.e., $p_{i,j}(b_{i,j}) - c_{i,j}^{a_j(t)} \geq 0$.

Definition 6 can guarantee that each taxi claims its valuation truthfully, since an untruthful bid will lead to a worse payoff. Definition 7 shows that each taxi can receive a nonnegative payoff if it participates in the auction. In addition, both the reverse auction mechanism and the automatic valuation mechanism need to achieve computational efficiency, defined as follows:

Definition 8 (Computational Efficiency). Each round of automatic valuation process and reverse auction process can terminate in a polynomial time.

For ease of reference, we list the main notations of this paper in Table 1.

3 The MAV and ROA Mechanisms

3.1 MAV: Automatic Valuation

We have formulated the automatic valuation as a multi-armed bandit process, in which the key is to seek a bandit policy maximizing the cumulative reward of each taxi. Since the distribution of the reward of each arm is unknown a prior, the fundamental challenge in the multi-armed bandit process is to balance the tradeoff between the *exploration* and *exploitation*. On the one hand, taxis have to explore the rewards by randomly selecting the arms. On the other hand, taxis also need to exploit the current knowledge of the rewards to select a best arm.

To solve this bandit dilemma, the Upper Confidence Bound (UCB) policy has been widely used [6,13]. However, this policy needs to select each arm once a time initially, which is impractical to be applied to our system. The ϵ-Greedy policy is another classical policy [20]. This policy selects a random arm with ϵ-frequency, and otherwise selects the arm with the current highest estimated expected reward. However, after enough explorations, the estimated rewards will be increasingly close to the true values. The constant factor $\epsilon \in [0,1]$ prevents the policy from getting arbitrarily close to the optimal arm. Therefore, in this paper, we first let taxis explore in the first t_0 rounds, where $t_0 > 0$. Then, each taxi explores with probability t_0/t, and exploits with probability $1 - t_0/t$. More specifically, in the t-th round, taxi v_j selects an arm $a_j(t) \in \mathcal{K}_j$ according to the

Algorithm 1. MAV: Automatic Valuation

Input: v_j, $\mathcal{K}_j$, $t_0 > 0$
Output: $m_j(t)$
1: $t \leftarrow 0$, $m_j(0) \leftarrow 0$;
2: **if** $t \leq t_0$ **then**
3: $\quad a_j(t) \leftarrow$ a random arm selected from $\mathcal{K}$;
4: **else if** rand() $< t_0/t$ **then**
5: $\quad a_j(t) \leftarrow$ a random arm selected from $\mathcal{K}$;
6: **else**
7: $\quad a_j(t) \leftarrow \text{argmax}_{k_j \in \mathcal{K}_j} \hat{\mu}_{k_j}(t-1)$;
8: $\forall k_j \in \mathcal{K}_j$, update $N_{k_j}(t)$ and $\hat{\mu}_{k_j}(t)$ according to Eq. 8 and Eq. 7, respectively;
9: $m_j(t) \leftarrow m_j(t-1) + X_{a_j(t)}(t)$;
10: $t \leftarrow t+1$;

following rule: when $t \leq t_0$, $a_j(t)$ is an arm randomly chosen from the set $\mathcal{K}_j$; when $t \geq t_0$,

$$a_j(t) = \begin{cases} \text{argmax}_{k_j \in \mathcal{K}_j} \hat{\mu}_{k_j}(t-1), \text{ with probability } 1 - \frac{t_0}{t}, \\ \text{a random arm in } \mathcal{K}_j, \quad \text{ with probability } \frac{t_0}{t}. \end{cases} \tag{5}$$

Here, $\hat{\mu}_{k_j}(t)$ is the estimated value of μ_{k_j}. Moreover, we estimate each expected reward value μ_{k_j} by averaging the rewards actually received, i.e.,

$$\hat{\mu}_{k_j}(t) = \frac{\sum_{\tau=1}^{t} X_{k_j}(\tau) \cdot \mathbb{1}(a_j(\tau) = k_j)}{N_{k_j}(t)}, \tag{6}$$

which is equivalent to the following recursive formulas:

$$\hat{\mu}_{k_j}(t) = \begin{cases} \hat{\mu}_{k_j}(t-1), & \text{if } a_j(t) \neq k_j, \\ \frac{\hat{\mu}_{k_j}(t-1) \cdot N_{k_j}(t-1) + X_{k_j}(t)}{N_{k_j}(t)}, & \text{if } a_j(t) = k_j, \end{cases} \tag{7}$$

and,

$$N_{k_j}(t) = \begin{cases} N_{k_j}(t-1), & \text{if } a_j(t) \neq k_j, \\ N_{k_j}(t-1) + 1, & \text{if } a_j(t) = k_j. \end{cases} \tag{8}$$

Based on the above policy, the MAV mechanism automatically selects the mark-up pricing strategies for each taxi in the order assignment process. Then, the taxis can efficiently determine their valuations for each order. The detailed automatic valuation algorithm is shown in Algorithm 1. Since the automatic valuation algorithm is distributed and conducted on each taxi's side, we only display the automatic valuation process of taxi v_j in Algorithm 1. In Steps 2–7, taxi v_j selects an arm $a_j(t)$. In Steps 8–9, the number of times of each arm that has been selected and the corresponding estimated reward are updated, followed by the computation of cumulative reward.

Algorithm 2. ROA: Payment Computation (PC)

Input: $G=\{\mathcal{O},\mathcal{V}',\Phi\}$, Ψ
Output: $\{p_{i,j}(b_{i,j})|\langle o_i,v_j\rangle\in\Psi\}$
1: Calculate the total social welfare $\sum_{\Psi}\phi_{i,j}$ on the matching Ψ;
2: **for** each $\langle o_i,v_j\rangle\in\Psi$ **do**
3: $\Phi_{-i,j}\leftarrow\Phi-\{\phi_{i,j}\}$; $G_{-i,j}\leftarrow\{\mathcal{O},\mathcal{V}',\Phi_{-i,j}\}$;
4: Finding a maximum weighted matching $\Psi_{-i,j}$ in graph $G_{-i,j}$;
5: Calculate the total social welfare $\sum_{\Psi_{-i,j}}\phi_{i,j}$ on the matching $\Psi_{-i,j}$;
6: $p_{i,j}(b_{i,j})\leftarrow\sum_{\Psi}\phi_{i,j}-\sum_{\Psi_{-i,j}}\phi_{i,j}+b_{i,j}$;

3.2 ROA: Optimal Winner Selection and Payment Computation

To select the winners of the auction and determine the order assignment results, we transform the winner selection problem into finding the maximum weighted bipartite matching problem. Consider the t-th round of order assignment where the set of orders is $\mathcal{O}$, the set of taxis is $\{\mathcal{V}_i|o_i\in\mathcal{O}\}$, and their bids are $\{b_{i,j}|o_i\in\mathcal{O},v_j\in\mathcal{V}_i\}$. We construct a weighted bipartite graph $G=\{\mathcal{O},\mathcal{V}',\Phi\}$, where $\mathcal{V}'=\cup_{o_i\in\mathcal{O}}\mathcal{V}_i$ and $\Phi=\{\phi_{i,j}|o_i\in\mathcal{O},v_j\in\mathcal{V}_i\subseteq\mathcal{V}'\}$. Here, the order set $\mathcal{O}$ and the taxi set $\mathcal{V}'$ are two separate vertex sets. Set Φ indicates the edges across $\mathcal{O}$ and $\mathcal{V}'$, and $\phi_{i,j}=r_i-b_{i,j}$ is the weight of edge $\langle o_i,v_j\rangle$. With the graph G, we can apply an existing maximum weighted matching algorithm, which has polynomial-time computational complexity, such as the famous Kuhn-Munkres algorithm [12,16], to get the optimal matching results. Let $\Psi=\{\langle o_i,v_j\rangle\}$ be the optimal matching with maximum weight in graph G. Then, we can get the winners and the order assignment results. We set $z_{i,j}=1$ if $\langle o_i,v_j\rangle\in\Psi$; otherwise, $z_{i,j}=0$.

In order to ensure that each taxi truthfully reports its true valuation, we compute the payment to each winning taxi based on the VCG auction [17]. The VCG auction can guarantee the truthfulness when the optimal assignment can be achieved. In VCG auction, the winner will be paid with the "externalities" that its presence incurs to others. More specifically, for a given weighted bipartite graph $G=\{\mathcal{O},\mathcal{V}',\Phi\}$ and the optimal matching $\Psi=\{\langle o_i,v_j\rangle\}$, the payment of a winning bid $b_{i,j}$ can be determined as follows.

First, we consider a winner selection with the bid $b_{i,j}$, and the matching solution is Ψ. Then, $\sum_{\Psi}\phi_{i,j}-\phi_{i,j}$ denotes the total social welfare produced by the matching Ψ except for the single social welfare $\phi_{i,j}$. Second, we consider a winner selection without the bid $b_{i,j}$. We remove edge $\langle o_i,v_j\rangle$ from G to get the corresponding weighted bipartite graph without $b_{i,j}$, denoted by $G_{-i,j}$, i.e., $G_{-i,j}=\{\mathcal{O},\mathcal{V}',\Phi_{-i,j}\}$, where $\Phi_{-i,j}=\Phi-\{\phi_{i,j}\}$. Then, we conduct the same maximum weighted matching algorithm over $G_{-i,j}$ to get a matching solution, denoted by $\Psi_{-i,j}$. Then, $\sum_{\Psi_{-i,j}}\phi_{i,j}$ denotes the total social welfare without the presence of bid $b_{i,j}$. Finally, the payment $p_{i,j}(b_{i,j})$ satisfies:

$$r_i-p_{i,j}(b_{i,j})=\sum\nolimits_{\Psi_{-i,j}}\phi_{i,j}-\Big(\sum\nolimits_{\Psi}\phi_{i,j}-\phi_{i,j}\Big),\tag{9}$$

which implies

$$p_{i,j}(b_{i,j}) = \sum\nolimits_{\Psi} \phi_{i,j} - \sum\nolimits_{\Psi_{-i,j}} \phi_{i,j} + b_{i,j}. \tag{10}$$

The detailed payment computation algorithm is shown in Algorithm 2. The total social welfare on the matching Ψ is calculated in Step 1. For each winning bid $b_{i,j}$, the weighted bipartite graph $G_{-i,j}$ is constructed in Step 3. In Steps 4–5, the maximum weighted matching algorithm is conducted over $G_{-i,j}$ and the total social welfare on the new matching $\Psi_{-i,j}$ is calculated. Then, the payment $p_{i,j}(b_{i,j})$ is computed in Step 6.

4 Theoretical Analysis

4.1 Online Performance of MAV

To analyze the online performance of the MAV mechanism, we derive the expected *regret* of an arbitrary taxi v_j. First, we consider an Oracle policy, which knows the value of μ_{k_j} and can select the optimal arm in each round. Let μ_j^* be the mean value associated with the optimal arm, i.e., $\mu_j^* = \max_{k_j \in \mathcal{K}_j} \mu_{k_j}$. Next, we define the loss of selecting the k_j-th arm as $\Delta_{k_j} = \mu_j^* - \mu_{k_j}$. Then, the expected regret, denoted by $R_j(t)$, can be defined as the loss in cumulative reward compared with the Oracle policy, i.e.,

$$R_j(t) = \mu_j^* \cdot t - m_j(t) = \mu_j^* \cdot t - \sum\nolimits_{k_j=1}^{K_j} \mu_{k_j} \mathbb{E}[N_{k_j}(t)] = \sum\nolimits_{k_j : \mu_{k_j} < \mu_j^*} \mathbb{E}[N_{k_j}(t)] \Delta_{k_j}. \tag{11}$$

Based on this, we can derive the following theorem.

Theorem 1. For any $\rho > 1$, $t \geq t_0$, the probability that taxi v_j selects a suboptimal arm l_j under Algorithm 1 is at most

$$\mathbb{P}[a_j(t) = l_j] \leq \frac{t_0}{tK_j} + \left(1 - \frac{t_0}{tK_j}\right)(\alpha + \beta), \tag{12}$$

where $\alpha = \frac{4}{\Delta_{l_j}^2} \exp\left(\frac{\Delta_{l_j}^2}{2}\right)\left(\frac{t_0}{t}\right)^{\frac{t_0 \Delta_{l_j}^2}{2\rho K_j}}$, $\beta = \frac{2t_0}{\rho K_j}\left(\frac{t_0}{t}\right)^{\frac{qt_0}{\rho K_j}} \ln\left(\frac{e^2 t}{t_0}\right)$, $q = \frac{3(\rho-1)^2}{8\rho-2}$.

To prove this theorem, we will make use of the following two inequalities for bounded random variables.

Lemma 1 (Chernoff-Hoeffding bound). Suppose that $X_1, X_2, \cdots, X_n$ are n random variables with common range [0, 1], satisfying $\mathbb{E}[X_t | X_1, \cdots, X_{t-1}] = \mu$ for $\forall t \in [1, n]$. Let $S_n = X_1 + \cdots + X_n$. Then, for any $a \geq 0$, we have:

$$\mathbb{P}[S_n \geq n\mu + a] \leq \exp(-2a^2/n), \mathbb{P}[S_n \leq n\mu - a] \leq \exp(-2a^2/n).$$

Lemma 2 (Bernstein inequality). Suppose that $X_1, X_2, \cdots, X_n$ are n random variables with common range [0, 1], and $\sum_{t=1}^{n} Var[X_t | X_1, \cdots, X_{t-1}] = \sigma^2$. Let $S_n = X_1 + \cdots + X_n$. Then, for any $a \geq 0$, we have:

$$\mathbb{P}\left[S_n \geq \mathbb{E}\left[S_n\right] + a\right] \leq \exp\left(-3a^2/(6\sigma^2 + 2a)\right), \mathbb{P}\left[S_n \leq \mathbb{E}\left[S_n\right] - a\right] \leq \exp\left(-3a^2/(6\sigma^2 + 2a)\right).$$

Proof: For some $\rho > 1$, let $x_0 = \frac{1}{\rho K_j}\big(t_0 + \sum_{\tau=t_0+1}^{t} \frac{t_0}{\tau}\big)$. The probability that taxi v_j selects the l_j-th arm in the t-th round is

$$\mathbb{P}\left[a_j(t) = l_j\right] \leq \frac{t_0}{tK_j} + \big(1 - \frac{t_0}{t}\big)\mathbb{P}[\hat{\mu}_{l_j}(t) \geq \hat{\mu}_j^*(t)], \tag{13}$$

in which

$$\mathbb{P}[\hat{\mu}_{l_j}(t) \geq \hat{\mu}_j^*(t)] \leq \mathbb{P}\big[\hat{\mu}_{l_j}(t) \geq \mu_{l_j} + \frac{\Delta_{l_j}}{2}\big] + \mathbb{P}\big[\hat{\mu}_j^*(t) \leq \mu_j^* - \frac{\Delta_{l_j}}{2}\big]. \tag{14}$$

The analyses of both the terms in the right hand side of Eq. 14 are the same. Let $N_{l_j}^{(R)}(t)$ be the number of times that the l_j-th arm has been selected in the exploration stage up to the t-th round. Then we have,

$$\begin{aligned}
&\mathbb{P}\big[\hat{\mu}_{l_j}(t) \geq \mu_{l_j} + \frac{\Delta_{l_j}}{2}\big] = \sum\nolimits_{\tau=1}^{t} \mathbb{P}\big[N_{l_j}(t) = \tau; \hat{\mu}_{l_j}(\tau) \geq \mu_{l_j} + \frac{\Delta_{l_j}}{2}\big] \\
&= \sum\nolimits_{\tau=1}^{t} \mathbb{P}\big[N_{l_j}(t) = \tau | \hat{\mu}_{l_j}(\tau) \geq \mu_{l_j} + \frac{\Delta_{l_j}}{2}\big] \cdot \mathbb{P}\big[\hat{\mu}_{l_j}(\tau) \geq \mu_{l_j} + \frac{\Delta_{l_j}}{2}\big] \\
&\leq \sum\nolimits_{\tau=1}^{t} \mathbb{P}\big[N_{l_j}(t) = \tau | \hat{\mu}_{l_j}(\tau) \geq \mu_{l_j} + \frac{\Delta_{l_j}}{2}\big] \cdot \exp(-\frac{\Delta_{l_j}^2 \tau}{2}) \\
&\quad \text{(according to the Chernoff-Hoeffding bound in Lemma 1)} \\
&\leq \sum\nolimits_{\tau=1}^{\lfloor x_0 \rfloor} \mathbb{P}\big[N_{l_j}(t) = \tau | \hat{\mu}_{l_j}(\tau) \geq \mu_{l_j} + \frac{\Delta_{l_j}}{2}\big] + \frac{2}{\Delta_{l_j}^2} \exp(-\frac{\Delta_{l_j}^2 \lfloor x_0 \rfloor}{2}) \\
&\leq x_0 \cdot \mathbb{P}\big[N_{l_j}^{(R)}(t) \leq x_0\big] + \frac{2}{\Delta_{l_j}^2} \exp(-\frac{\Delta_{l_j}^2 \lfloor x_0 \rfloor}{2}).
\end{aligned} \tag{15}$$

Since $\mathbb{E}[N_{l_j}^{(R)}(t)] = \frac{t_0}{K_j} + \sum_{\tau=t_0+1}^{t} \frac{t_0}{\tau K_j} = \rho x_0$, and $var[N_{l_j}^{(R)}(t)] = \sum_{\tau=t_0+1}^{t} \big((\frac{t_0}{K_j} + \frac{t_0}{tK_j}) - (\frac{t_0}{K_j} + \frac{t_0}{tK})^2\big) \leq \mathbb{E}[N_{l_j}^{(R)}(t)] = \rho x_0$, according to the Bernstein inequality given in Lemma 2, we have,

$$\mathbb{P}[N_{l_j}^{(R)}(t) \leq x_0] = \mathbb{P}[N_{l_j}^{(R)}(t) \leq \rho x_0 - (\rho - 1)x_0] \leq \exp(-q x_0), \tag{16}$$

where $q = \frac{3(\rho-1)^2}{8\rho - 2}$.

Next, we derive the upper and lower bounds on x_0. Since $x_0 = \frac{1}{\rho K_j}\big(t_0 + \sum_{\tau=t_0+1}^{t} \frac{t_0}{\tau}\big) = \frac{t_0}{\rho K_j}\big(1 + \sum_{\tau=t_0+1}^{t} \frac{1}{\tau}\big)$, and $\ln(\frac{t}{e t_0}) \leq \sum_{\tau=t_0+1}^{\tau} \frac{1}{t} \leq \ln(\frac{e^2 t}{t_0})$, we have

$$\frac{t_0}{\rho K_j} \ln(\frac{t}{t_0}) \leq x_0 \leq \frac{t_0}{\rho K_j} \ln(\frac{e^2 t}{t_0}). \tag{17}$$

Combining Eqs. 15–17, we have

$$\mathbb{P}\Big[\hat{\mu}_{l_j}(t) \geq \mu_{l_j} + \frac{\Delta_{l_j}}{2}\Big] \leq \frac{t_0}{\rho K_j}\Big(\frac{t_0}{t}\Big)^{\frac{q t_0}{\rho K_j}} \ln(\frac{e^2 t}{t_0}) + \frac{2}{\Delta_{l_j}^2}\Big(\frac{t_0}{t}\Big)^{\frac{t_0 \Delta_{l_j}^2}{2\rho K_j}} \exp(\frac{\Delta_{l_j}^2}{2}).$$

In the same way, we can obtain

$$\mathbb{P}\Big[\hat{\mu}_j^*(t) \le \mu_j^* - \frac{\Delta_{l_j}}{2}\Big] \le \frac{t_0}{\rho K_j}\Big(\frac{t_0}{t}\Big)^{\frac{qt_0}{\rho K_j}} \ln(\frac{e^2 t}{t_0}) + \frac{2}{\Delta_{l_j}^2}\Big(\frac{t_0}{t}\Big)^{\frac{t_0 \Delta_{l_j}^2}{2\rho K_j}} \exp(\frac{\Delta_{l_j}^2}{2}).$$

Therefore, according to Eq. 13, the probability $\mathbb{P}[a_j(t) = l_j]$ is as most

$$\frac{t_0}{tK_j} + (1 - \frac{t_0}{t})\Big(\frac{4}{\Delta_{l_j}^2}\Big(\frac{t_0}{t}\Big)^{\frac{t_0 \Delta_{l_j}^2}{2\rho K_j}} \exp(\frac{\Delta_{l_j}^2}{2})\Big) + (1 - \frac{t_0}{t})\Big(\frac{2t_0}{\rho K_j}(\frac{t_0}{t})^{\frac{qt_0}{\rho K_j}} \ln(\frac{e^2 t}{t_0})\Big).$$

The theorem holds. ■

Finally, based on the above theorem, we obtain the following theorem which bounds the expected regret of our bandit policy.

Theorem 2. For any $\rho > 1$, given parameter t_0 such that $t_0 \ge \max\{\frac{2\rho K_j}{\Delta_{min_j}^2}, \frac{\rho K_j}{q}\}$, where $\Delta_{min_j} = \min_{l_j : \mu_{l_j} < \mu_j^*} \Delta_{l_j}$ and $q = \frac{3(\rho-1)^2}{8\rho-2}$. Then, in each t-th round of auction where $t > t_0$, for an arbitrary taxi v_j, the expected regret produced by the bandit policy described in Algorithm 1 is at most $\Big(\sum_{l_j : \mu_{l_j} < \mu_j^*} \Delta_{l_j}\Big) \frac{t_0}{K_j} \ln t + O(\frac{1}{t})$.

4.2 Truthfulness, Individual Rationality and Efficiency

Theorem 3. The ROA mechanism satisfies the property of truthfulness.

Proof: Suppose that taxi v_j submits an untruthful bid $b'_{i,j}$ for order o_i, i.e., $b'_{i,j} \ne c_{i,j}^{a_j(t)}$. The payment to the bid $b'_{i,j}$ and order assignment result is denoted as $z'_{i,j}$. Denote the order assignment result as $z_{i,j}$ under the case $b_{i,j} = c_{i,j}^{a_j(t)}$. Then, there are two cases: (1) $z'_{i,j} = z_{i,j}$; (2) $z'_{i,j} \ne z_{i,j}$.

Case 1 ($z'_{i,j} = z_{i,j}$): If $z'_{i,j} = z_{i,j} = 0$, it is obviously that the payoffs under the truthful information and the untruthful information are the same and equal to 0. If $z'_{i,j} = z_{i,j} = 1$, then the order o_i is assigned to taxi v_j with bid $b_{i,j}$ or $b'_{i,j}$. This means that the presence of bid $b_{i,j}$ or $b'_{i,j}$ incurs no effect to other assignment results. Thus, we can obtain that $p_{i,j}(b'_{i,j}) = \sum_{\Psi} \phi_{i,j} - \sum_{\Psi_{-i,j}} \phi_{i,j} + b'_{i,j} = \sum_{\Psi \setminus \{\langle o_i, v_j \rangle\}} \phi_{i,j} - \sum_{\Psi_{-i,j}} \phi_{i,j} + r_i$. This indicates that the payment is independent of the bid submitted by taxi v_j. Therefore, in this case, $p_{i,j}(b'_{i,j}) = p_{i,j}(b_{i,j})$, and the payoffs are the same as well.

Case 2 ($z'_{i,j} \ne z_{i,j}$): Consider the case $z'_{i,j} = 0$ and $z_{i,j} = 1$. This implies that taxi v_j loses order o_i when bidding untruthfully, and its payoff is 0. Therefore, the misreporting leads to the less payoff than bidding truthfully. If $z'_{i,j} = 1$ and $z_{i,j} = 0$, which means that taxi v_j wins order o_i with the untruthful bid $b'_{i,j}$, then the taxi must claim a lower bid, i.e., $b'_{i,j} < b_{i,j}$, and $b'_{i,j} \le p_{i,j}(b'_{i,j})$. Since taxi v_j loses order o_i with bid $b_{i,j}$, we have $p_{i,j}(b_{i,j}) \le b_{i,j}$. Consequently, its payoff satisfies: $p_{i,j}(b'_{i,j}) - c_{i,j}^{a_j(t)} \le b_{i,j} - c_{i,j}^{a_j(t)} = c_{i,j}^{a_j(t)} - c_{i,j}^{a_j(t)} = 0$. Thus, in this case, the payoff is negative.

Therefore, each taxi cannot increase its payoffs by manipulating its real valuations, which proves the theorem. ■

Theorem 4. The ROA mechanism meets the condition of individual rationality.

Proof: If a taxi v_j does not win the order o_i with the bid $b_{i,j}$, the corresponding payoff will be zero. Otherwise, if taxi v_j wins the order o_i with bid $b_{i,j}$, the corresponding payoff is $p_{i,j}(b_{i,j}) - c_{i,j}^{a_j(t)}$. Here, according to Theorem 3, each bid must be submitted truthfully to achieve the best payoff. Then, $p_{i,j}(b_{i,j}) = p_{i,j}(c_{i,j}^{a_j(t)})$, which implies $p_{i,j}(b_{i,j}) - c_{i,j}^{a_j(t)} = \sum_{\Psi} \phi_{i,j} - \sum_{\Psi_{-i,j}} \phi_{i,j}$. Since $\sum_{\Psi} \phi_{i,j}$ is the optimal solution of the winner selection problem and $\sum_{\Psi_{-i,j}} \phi_{i,j}$ is only a feasible solution where the bid $b_{i,j}$ is absent, we have $\sum_{\Psi} \phi_{i,j} - \sum_{\Psi_{-i,j}} \phi_{i,j} \geq 0$. Hence, the payoff is no less than 0. The theorem holds. ■

Next, we prove the computational efficiency of MAV and ROA.

Theorem 5. The ROA mechanism and the MAV mechanism both have a polynomial time computation complexity in one round of order assignment.

Proof: The ROA mechanism is composed of the winner selection and the payment computation processes. As described in Sect. 3.2, each round of winner selection can be optimally solved with an existing maximal weighted matching algorithm, whose computation complexity is as most $O(\max\{|\mathcal{O}|, |\mathcal{V}|\}^3)$. The payment computation is completed in Algorithm 2, which is dominated by Step 4. Thus, the computation complexity is as most $O(\max\{|\mathcal{O}|, |\mathcal{V}|\}^3 \cdot \min\{|\mathcal{O}|, |\mathcal{V}|\})$. Therefore, the ROA mechanism can terminate in a polynomial time. The computation overhead of Algorithm 1 of the MAV mechanism is dominated by Step 7, i.e., $O(|\mathcal{K}_j| \ln(|\mathcal{K}_j|))$. Therefore, the MAV mechanism can terminate in a polynomial time. Therefore, the theorem holds. ■

Table 2. Simulation settings

Parameter name	Values
the average number of orders per auction period $\lvert\mathcal{O}\rvert$	50, **100**, 150, 200
the average number of taxis per auction period $\lvert\mathcal{V}\rvert$	300, 400, 500, **600**
parameter t_0 in the $\frac{t_0}{t}$-Greedy policy	**100**, 1000, 5000
parameter ϵ in the ϵ-Greedy policy	0.1, 0.01

5 Evaluation

5.1 Algorithms in Comparison

In order to evaluate the online performance of the MAV mechanism, we implement an automatic valuation algorithm based on the ϵ-Greedy policy for comparison [20]. Given a fixed parameter ϵ, the ϵ-Greedy policy selects the arm $k_j = \mathrm{argmax}_{l_j \in \mathcal{K}_j} \hat{\mu}_{l_j}(t-1)$ with probability $1-\epsilon$; otherwise, the policy selects a random arm with probability ϵ.

In order to evaluate the order assignment performance of ROA, i.e., the social welfare performance, we implement three other algorithms for comparison: the Greedy (GRY) algorithm, the Nearest Taxi Selection (NTS) algorithm [9], and the Immediate Selection (IS) algorithm. The GRY algorithm conducts the order assignment under the same bipartite weighted graph as which is constructed in the ROA mechanism. Different from the optimal winner selection algorithm, the GRY algorithm always selects the edge with the largest weight until the taxi set or the order set becomes empty. The taxis in the selected edges are the winners. And the orders in the selected edges are assigned to the corresponding taxis. It is noted that the VCG auction requires that the optimal assignment of the orders must be guaranteed. Consequently, we apply the second price auction to determine the winners' payments in the GRY algorithm. The NTS algorithm selects the nearest taxi for each order. The IS algorithm makes the assignment decision immediately after each order arrives at the platform and each order will be assigned to the taxi that has the maximum single social welfare currently.

5.2 Simulation Parameters and Settings

In the evaluation, we use a trace of New York City's taxi trips on January, 2016 [1], which is also used in [23]. This trace consists of about 100,000 completed trip records in 24 h (after discarding some obvious inaccurate records), which is at most 100 trips per minute on average. Each trip record in the trace is composed of the pick-up and drop-off locations (shown as latitude/longitude), the pick-up and drop-off times, the trip distance, and the payment details. From these records, we directly extract the $startTime$, $startLoc$, and Des values of each order and taxi. According to these values, we derive each round of orders $\mathcal{O}$ and taxis $\mathcal{V}$ through determining the period TTL. For each order $o_i \in \mathcal{O}$, we also determine the set of serviceable taxis $\mathcal{V}_i$.

Since the trace only contains successful transport records without involving any auction mechanisms, there are no records about the taxis' bids. To evaluate MAV and ROA, we first let taxi fare of each order be equal to the payment value in the trace. Next, we generate each valuation $c_{i,j}^{a_j(t)}$ as follows. First, we set a base price $basePrice_i$ for each order o_i as the inherent transport cost and let $basePrice_i = \frac{r_i}{2}$. Second, we set the number of service charging strategies of K_j each taxi as 20, which are set as 20 values randomly chosen from [0, 1], denoted as $Ser^{1_j}, \cdots, Ser^{K_j}$. Third, the rewards $\{X_{k_j}(t) | v_j \in \mathcal{V}, k_j \in \mathcal{K}_j\}$ in each round of order assignment are randomly sampled from a truncated Gaussian distribution with mean $\mu_{k_j} \in [0, 1]$, standard deviation $\sigma_{k_j} \in [0, 1]$, and support [0, 1]. Finally, each taxi v_j selects its service charging strategies by a policy to obtain the maximal cumulative reward, i.e., the minimal cumulative regret. For ease of description, we call the arm selection policy described in Algorithm 1 the $\frac{t_0}{t}$-Greedy policy. Then, if taxi v_j selects the $a_j(t)$-th strategy in the t-th round of order assignment, its valuation for order o_i will be $c_{i,j}^{a_j(t)} = basePrice_i(1+Ser^{a_j(t)})$, meanwhile taxi v_j will receive the corresponding reward $X_{a_j(t)}(t)$.

In addition, we set different values for the parameter t_0 in our $\frac{t_0}{t}$-Greedy policy, and set different values for the parameter ϵ in the ϵ-Greedy policy for concrete comparison. The detailed parameter settings are listed in Table 2, where the default values are in bold fonts.

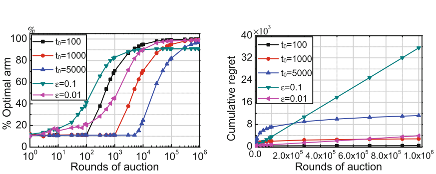
5.3 Evaluation Metrics and Results

The major metrics in our simulations include the *Online Performance* w.r.t. the MAV mechanism, the *Social Welfare*, *Truthfulness*, *Individual Rationality*, *Overpayment Ratio* and *Time Efficiency* w.r.t. the ROA mechanism. Here, the overpayment is the difference between the total payment to winners and the sum of valuations of each winner. Then, the overpayment ratio is defined as:

$$\lambda = \frac{\sum_{\Psi} p_{i,j}(b_{i,j}) - \sum_{\Psi} c_{i,j}^{a_j(t)}}{\sum_{\Psi} c_{i,j}^{a_j(t)}}. \tag{18}$$

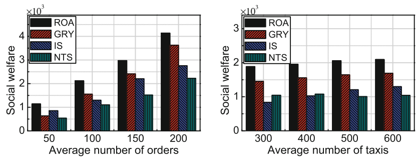
It measures the payments paid by the platform to induce the truthfulness of all taxis. The evaluation results are presented as follows.

Online Performance of MAV. To evaluate the online performance of the MAV mechanism, we track two performance metrics: the cumulative regret and the frequency of selecting the optimal arm (denoted as "% optimal arm"). The results are shown in Figs. 4(a)–(b), in which each curve is the average output of 1000 times of repeated simulations. We can find that the regret generated by the $\frac{t_0}{t}$-Greedy policy grows logarithmically over time, which is consistent with the theoretical analysis in Theorem 2. Moreover, when t_0 is smaller, the regret grows at a slower speed as shown in Fig. 4(b). This illuminates that the policy has learnt well after a small number of pure exploration. We can also discover that the $\epsilon = 0.1$ policy explores more than the $\epsilon = 0.01$ policy, and it finds the optimal arm earlier. The $\epsilon = 0.01$ policy learns more slowly, but it eventually performs better than the $\epsilon = 0.1$ policy. Nevertheless, an optimally tuned $\frac{t_0}{t}$-Greedy policy (e.g., $t_0 = 100$) performs almost best among other policies.



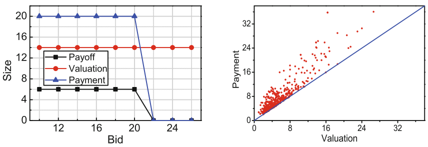
(a) % optimal arm (b) Regret

Fig. 4. Evaluation on online performance of MAV.



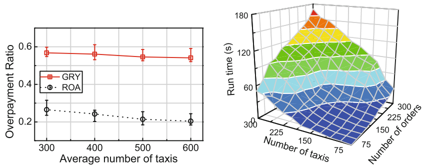
(a) $|\mathcal{V}| = 600$ (b) $|\mathcal{O}| = 100$

Fig. 5. Evaluation on social welfare of ROA.



(a) Truthfulness (b) Individual rationality

Fig. 6. Evaluation on truthfulness and individual rationality.



(a) Overpayment ratio (b) Run time

Fig. 7. Evaluation on overpayment ratio and time efficiency.

Social Welfare of ROA. We evaluate the social welfare performance of the ROA mechanism as follows. First, we set the average number of orders per auction period at $|\mathcal{O}| = 50$, 100, 150, and 200 by randomly selecting 50 orders in each minute and letting the auction period $TTL = 1$, 2, 3, and 4 min. Meanwhile, we fix the average number of taxis per auction period at $|\mathcal{V}| = 600$. Second, we change $|\mathcal{V}|$ from 300 to 600, while fixing $|\mathcal{O}| = 100$. Finally, we conduct the MAV mechanism with $t_0 = 100$ to generate the valuations of each taxi. Taxis report their true valuations as their bids to participate in the auction. The results are shown in Figs. 5(a)–(b). On average, the social welfare of ROA is about 79.38%, 61.66% and 52.05% larger than those of GRY, IS and NTS, respectively. Moreover, the social welfare increases with the increasing numbers of orders and taxis, but the increment of the latter is limited. This is because only a few of increased taxis can win the auction.

Truthfulness and Individual Rationality of ROA. We verify the truthfulness and individual rationality of ROA under the default settings. First, we randomly select a bid and allow the corresponding taxi to claim a bid different from its real valuation. The result, depicted in Fig. 6(a), shows that the payoff remains unchanged when the taxi's bid is smaller than its valuation. This means that each taxi will still be winner when it claims a lower bid than its current winning bid. However, the payoff is zero when the bid is larger than its payment. This means that the payment paid to each taxi is a critical value ensuring to be a winner. We can hence find that each taxi cannot improve its payoff by bidding untruthfully. To verify the individual rationality, we randomly choose plenty of taxis and orders, and compare the valuation of each taxi with the corresponding payment. The result, plotted in Fig. 6(b), shows that each payment is larger than the corresponding valuation. The individual rationality is also guaranteed.

Overpayment Ratio and Time Efficiency of ROA. To evaluate the overpayment ratio performance, we make a comparison with the GRY algorithm. Figure 7(a) shows that the overpayment ratio of ROA is smaller than that of GRY. This implies that the GRY algorithm must pay more so as to induce cooperation from selfish taxis. Moreover, the overpayment ratio of ROA decreases slightly with the increasing number of taxis. This is because that the increasing number of taxis means more taxis with low valuations can be winners, leading to the reduced overpayment ratio. Second, as shown in Fig. 7(b), when the number of orders is 50, and the number of taxis is 300, the run time is less than 1 min, which is smaller than the auction period. Moreover, when the numbers of orders and taxis are both 300, the run time is no more than 3 min. Therefore, the ROA mechanism can work efficiently in real applications.

6 Related Work

In recent years, much attention has been drawn to the study of the taxi-hailing order assignment, taxi dispatch problem, and the task assignment problem in vehicle-based crowdsourcing, such as [9,10,14,15,18,21–23]. However, most of these works are based on the direct assignment strategy without involving any auction and personalized pricing mechanisms. Also, many ride sharing services have appeared along with various algorithms on how to match an order to a taxi which can provide the ride sharing service [4,5,7,8], in which the most related works are [4,5]. Different from our work, [4,5] do not consider the process in which the taxis' valuations for orders can be learnt or refined over time by observing the historical assignment results. In view of this, we introduce the multi-armed bandit model in our COA framework, by which taxis can automatically price for the orders. Then, they participate in the order auction process. The multi-armed bandit is an online learning model which is widely used in crowdsourcing, cognitive radio networks, etc., [11,19]. For example, [19] models the unknown expert recruitment problem in crowdsourcing as the multi-armed bandit game, where the unknown experts are seen as arms.

7 Conclusion and Future Work

In this paper, we study the order assignment problem in the mobile taxi-hailing systems and propose a competitive order assignment (COA) framework. In COA, we let each taxi automatically set valuations for its preferred orders and design the MAV mechanism. Then, we conduct the competitive order assignment based on a reverse auction and design the ROA mechanism. Further, we analyze the online performance of MAV, and proof that ROA is truthful and individually rational. Moreover, the significance performances of ROA and MAV are also verifies through extensive simulations on a real trace.

Acknowledgment. This research was supported in part by National Natural Science Foundation of China (NSFC) (Grant No. 61872330, 61572336, 61572457, 61632016, 61379132, U1709217), NSF grants CNS 1757533, CNS 1629746, CNS 1564128, CNS 1449860, CNS 1461932, CNS 1460971, IIP 1439672, Natural Science Foundation of Jiangsu Province in China (Grant No. BK20131174, BK2009150), Anhui Initiative in Quantum Information Technologies (Grant No. AHY150300), and Natural Science Research Project of Jiangsu Higher Education Institution (No. 18KJA520010).

References

1. Nyc taxi trips (2016). http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml
2. DMR, January 8. https://expandedramblings.com/index.php/uber-statistics/
3. Agrawal, R.: Sample mean based index policies by $o(\log n)$ regret for the multi-armed bandit problem. Adv. Appl. Probab. **27**(4), 1054–1078 (1995)
4. Asghari, M., Deng, D., Shahabi, C., Demiryurek, U., Li, Y.: Price-aware real-time ride-sharing at scale: an auction-based approach. In: ACM SIGSPATIAL (2016)
5. Asghari, M., Shahabi, C.: An on-line truthful and individually rational pricing mechanism for ride-sharing. In: ACM SIGSPATIAL (2017)
6. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Mach. Learn. **47**(2–3), 235–256 (2002)
7. Chen, L., Zhong, Q., Xiao, X., Gao, Y., Jin, P., Jensen, C.S.: Price-and-time-aware dynamic ridesharing. In: IEEE ICDE (2018)
8. Chen, L., Gao, Y., Liu, Z., Xiao, X., Jensen, C.S., Zhu, Y.: PTrider: a price-and-time-aware ridesharing system. Proc. VLDB Endow. **11**(12), 1938–1941 (2018)
9. Gao, G., Xiao, M., Zhao, Z.: Optimal multi-taxi dispatch for mobile taxi-hailing systems. In: ICPP (2016)
10. Garg, N., Ranu, S.: Route recommendations for idle taxi drivers: find me the shortest route to a customer! In: ACM SIGKDD (2018)
11. Kang, S., Joo, C.: Low-complexity learning for dynamic spectrum access in multi-user multi-channel networks. In: IEEE INFOCOM (2018)
12. Kuhn, H.W.: The Hungarian method for the assignment problem. Naval Res. Logistics (NRL) **2**(1–2), 83–97 (1955)
13. Lai, T.L., Robbins, H.: Asymptotically efficient adaptive allocation rules. Adv. Appl. Math. **6**(1), 4–22 (1985)
14. Liu, A., et al.: Privacy-preserving task assignment in spatial crowdsourcing. J. Comput. Sci. Technol. **32**(5), 905–918 (2017)
15. Liu, A., Wang, W., Shang, S., Li, Q., Zhang, X.: Efficient task assignment in spatial crowdsourcing with worker and task privacy protection. GeoInformatica **22**(2), 335–362 (2018)
16. Munkres, J.: Algorithms for the assignment and transportation problems. J. Soc. Ind. Appl. Math. **5**(1), 32–38 (1957)
17. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V.: Algorithmic Game Theory. Cambridge University Press, New York (2007)
18. Tong, Y., Wang, L., Zhou, Z., Chen, L., Du, B., Ye, J.: Dynamic pricing in spatial crowdsourcing: a matching-based approach. In: ACM SIGMOD (2018)
19. Tran-Thanh, L., Stein, S., Rogers, A., Jennings, N.R.: Efficient crowdsourcing of unknown experts using bounded multi-armed bandits. AI **214**, 89–111 (2014)
20. Vermorel, J., Mohri, M.: Multi-armed bandit algorithms and empirical evaluation. In: Gama, J., Camacho, R., Brazdil, P.B., Jorge, A.M., Torgo, L. (eds.) ECML 2005. LNCS (LNAI), vol. 3720, pp. 437–448. Springer, Heidelberg (2005). https://doi.org/10.1007/11564096_42
21. Xiao, M., et al.: SRA: secure reverse auction for task assignment in spatial crowdsourcing. In: IEEE TKDE, p. 1 (2019)
22. Xiao, M., Wu, J., Huang, L., Cheng, R., Wang, Y.: Online task assignment for crowdsensing in predictable mobile social networks. IEEE Trans. Mob. Comput. **16**(8), 2306–2320 (2017)
23. Zheng, H., Wu, J.: Online to offline business: urban taxi dispatching with passenger-driver matching stability. In: IEEE ICDCS (2017)