Order Acquisition Under Competitive Pressure: A Rapidly Adaptive Reinforcement Learning Approach for Ride-Hailing Subsidy Strategies

Fangzhou Shi ✉[0009−0000−4210−712X], Xiaopeng Ke[0009−0006−0039−4013], Xinye Xiong[0009−0009−4825−6180], Kexin Meng[0009−0004−3232−9887], Chang Men[0009−0005−6028−2657], and Zhengdan Zhu[0009−0009−6217−8721]

Didi Chuxing, Beijing, China
{arkshifangzhou,kexiaopeng,cintiaxiong,
kexinmeng,menchang,zhuzhengdan}@didiglobal.com

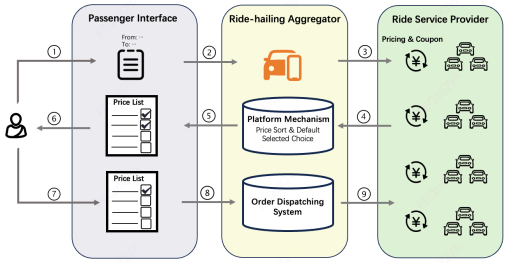
Abstract. The proliferation of ride-hailing aggregator platforms presents significant growth opportunities for ride-service providers by increasing order volume and gross merchandise value (GMV). On most ride-hailing aggregator platforms, service providers that offer lower fares are ranked higher in listings and, consequently, are more likely to be selected by passengers. This competitive ranking mechanism creates a strong incentive for service providers to adopt coupon strategies that lower prices to secure a greater number of orders, as order volume directly influences their long-term viability and sustainability. Thus, designing an effective coupon strategy that can dynamically adapt to market fluctuations while optimizing order acquisition under budget constraints is a critical research challenge. However, existing studies in this area remain scarce. To bridge this gap, we propose FCA-RL, a novel reinforcement learning-based subsidy strategy framework designed to rapidly adapt to competitors' pricing adjustments. Our approach integrates two key techniques: Fast Competition Adaptation (FCA), which enables swift responses to dynamic price changes, and Reinforced Lagrangian Adjustment (RLA), which ensures adherence to budget constraints while optimizing coupon decisions on new price landscape. Furthermore, we introduce RideGym, the first dedicated simulation environment tailored for ride-hailing aggregators, facilitating comprehensive evaluation and benchmarking of different pricing strategies without compromising real-world operational efficiency. Experimental results demonstrate that our proposed method consistently outperforms baseline approaches across diverse market conditions, highlighting its effectiveness in subsidy optimization for ride-hailing service providers.

Keywords: Reinforcement Learning · Ride-Hailing · Subsidy Strategy.

1 Introduction

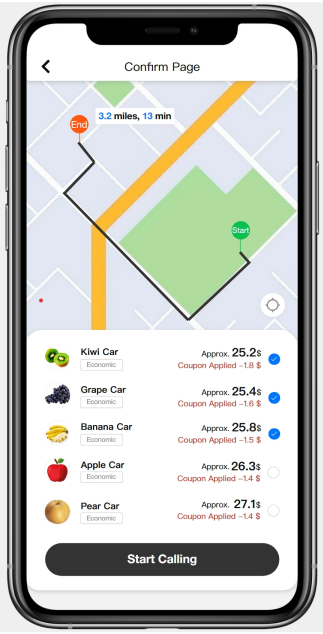
The rise of ride-hailing aggregators [1]—platforms integrating various third-party ride-hailing services—has grown alongside the ride-hailing industry [12]. These platforms

enhance demand realization and social welfare by increasing market thickness and reducing matching frictions, allowing smaller ride-service providers to boost order volume and gross merchandise value (GMV). As depicted in the Figure 1a, key stakeholders include the *Passenger*, *Ride-Hailing Aggregators (RHA)*, and *Ride-Service Providers (RSP)*. The prices quoted by each RSP are sorted and displayed on the passenger's interface (demonstrated in Figure 1b). To improve user experience, RHAs typically automatically select the top-K lowest-priced options. This mechanism strongly motivates RSPs to lower their price to fall within this range, as the majority of passengers tend to maintain this default range when sending orders due to inherent inertia.



① - ③ The passenger inputs the origin and destination, prompting the aggregator to request price quotes from all service providers.
④ - ⑥ Service providers submit their quoted prices after applying coupons. The aggregator ranks them and auto-selects certain options based on its mechanism before presenting the price list to the passenger.
⑦ - ⑨ The passenger send an order with the chosen options, and the aggregator dispatches it to the most suitable driver from the selected service providers.

(a) **The interaction process between a passenger, a ride-hailing aggregator, and ride service providers**.



(b) **A demonstration of the passenger's interface form on the ride-hailing aggregator**.

Fig. 1: **The ride-hailing aggregation process and passenger interface.**

Once an order is sent, it will be dispatched to one of the most suitable drivers of the RSPs selected by the passenger. This means that, for a RSP, the fewer ride-service options a passenger selects in a request, the less competition for order completion it will face. Therefore, competing on price (i.e., the travel fare) with other RSPs essentially prompts passengers to select and send orders, thereby increasing the likelihood of order completion. Unlike other studies that discuss long-term base price pricing strate-

gies [15,19], or dynamic price surging [4] under extreme supply & demand conditions, this paper focuses on RSPs that use instant coupons to granularly adjust prices in response to price competition in RHA, a topic that has been underexplored in previous research.

The limited budget of each RSP necessitates the development of efficient coupon allocation strategies for competitive effectiveness, which presents a significant research challenge. Traditional methods borrow ideas from user marketing [20], using uplift modeling [6] techniques to model the individual treatment effect in the passenger's willingness to send an order, after which the duality method with Lagrangian multiplier is applied to solve the maximization problem of order volume (or GMV) under budget constraints. However, these methods usually assume a stable distribution of features, treating the data as Independent and Identically Distributed (IID), which does not hold in the real world, especially when other RSPs perform aperiodic price adjustments.

The field of Real-Time Bidding (RTB) [17] faces a similar challenge of dynamic price competition. In RTB, merchants bid for ad positions in personalized recommendations, akin to how RSPs use coupons to improve rankings in passenger requests. Dynamic price competition occurs in both areas, with some studies in RTB predicting market prices [10] from historical data or using reinforcement learning for real-time bid adjustments [2,9,14]. However, applying RTB methods to RHA is complicated by two main differences. First, RSP coupons directly affect the rank and the final travel fare, whereas RTB bids are not related to the final sales price. Second, order completion in RHA depends on the number of selected RSPs, real-time supply status, and the RHA dispatch algorithm, which introduces significant uncertainty. In contrast, the conversion rate in RTB depends primarily on the item price and user interest. These differences complicate the design of coupon strategies in the RHA environment.

We propose FCA-RL, a framework that uses Bayesian posterior updates and reinforcement learning to dynamically mitigate the adverse effects of competitors' aperiodic price adjustments on coupon efficiency and budget control. From the perspective of RSPs, we formulate this budget-constrained order maximization problem as a Markov Decision Process (MDP). To solve this MDP under aperiodic price adjustments by competitors, we introduce a **Fast Competition Adaptation (FCA)** module that quickly tracks the evolving price landscape, and a **Reinforced Lagrangian Adjustment (RLA)** module that optimizes coupon allocation according to the updated market conditions. The integration of these two modules ensures sustained efficiency and accurate budget execution in a dynamic RHA environment.

Evaluating coupon strategies in real time is risky due to potential declines in GMV and order volume. To overcome this, we developed **RideGym**, an offline simulation system that emulates the ride-hailing process and allows for flexible configurations, including supply factors and competition levels. In addition, this system serves as an interactive environment for training our reinforcement learning agent. Our evaluations show that our algorithm significantly outperforms baseline models by reducing budget control errors and improving the Finish Return on Investment.

Our main contributions are threefold: **(1) The first work focuses on the coupon strategies of RSPs in RHA.** To the best of our knowledge, we are the first study to

explore using coupons for instant price competition in RHAs from the perspective of a RSP. **(2) A novel dynamic RL subsidy framework.** We propose FCA-RL, a novel reinforcement learning-based coupon strategy framework configured with a module for rapid market response in RHA, which enables precise budget control in dynamic markets without compromising coupon efficacy. **(3) A Full-Chain RHA Simulator.**We have developed a simulation system, **RideGym**, that models the complete lifecycle of an order within the RHA, facilitating comprehensive analysis of coupon strategies under various algorithmic and environmental variations.

2 Related Works

Coupon Strategy in Industry. Coupon strategies are widely employed in e-commerce to enhance customer purchasing willingness while adhering to budget constraints. Zou et al. [20] formulate this problem as a constrained optimization task based on uplift estimation and solve it using a duality method with Lagrangian relaxation, under the assumption of a stable distribution. Dai et al. [5] introduced a PID controller to dynamically adjust the Lagrangian multiplier for real-time traffic control but did not account for the optimality. Xiao et al. [16] define a model-based constrained Markov decision process (CMDP) and propose a joint optimization of both λ and the policy. Zhang et al. [18] utilize offline reinforcement learning to allocate personalized discounts under a predefined budget, leveraging offline dataset augmentation with multiple λ choices; however, their offline approach is unsuitable for scenarios where market conditions are unpredictable and evolve irregularly. Chen et al. [3] design an optimal instant discount strategy for multiple products on ride-hailing aggregators, but their focus on aggregator profitability diverges from our approach.

Online Advertising Bidding. The optimal coupon strategy of RSP in RHA is similar to advertisers bidding in RTB, where merchants compete for prime placements. Zhang et al. [17] model the winning function and advertisers' private values under specific formulations, deriving an optimal bidding strategy. Recently, reinforcement learning has gained traction in bidding. Cai et al. [2] model request distribution transitions in advertising as an MDP, using dynamic programming for optimal bidding. He et al. [9] propose a general bidding framework using the dual Lagrangian method and reinforcement learning for budget control under traffic fluctuations but remains ineffective for market price dynamics. Wang et al. [14,13] model the problem as a Partial Observable Markov Decision Process (POMDP), leveraging Transformers and variational inference to capture market price uncertainties, achieving better adaptability.

3 Problem Formalization

3.1 Original Problem

Notation. Suppose M RSPs participate in the RHA. From the perspective of a single RSP, we define a budget rate $B \in [0, 1]$ and a coupon set $\mathbf{d} = \{d_0, \ldots, d_{H-1}\} \subseteq [0, 1]$, where a 20% discount corresponds to $d_j = 0.2$. Over the period $[t_{start}, t_{end}]$, we collect

a dataset $\mathcal{D}_{t_{start}}^{t_{end}} = \{(\mathbf{x}_i, g_i, d_{i,j}, y_i^{I,j}, y_i^{C,j})\}_{i=0}^{N-1}$, which includes contextual features $\mathbf{x}_i$ (e.g., supply-demand conditions and order distance), base prices $g_i \in \mathbb{R}_+^1$ before coupon application, the applied coupon $d_{i,j}$, and two binary variables: $y_i^{I,j} \in \{0,1\}$, indicating whether the RHA automatically selects the RSP in the passenger's form (Figure 1a-⑤) with coupon $d_{i,j}$, and $y_i^{C,j} \in \{0,1\}$, indicating the order completion status after applying $d_{i,j}$. Notably, $y_i^{I,j}$ and $y_i^{C,j}$ are sampled from the corresponding potential outcome distributions $P_{\mathbf{x}_i}^{I,(j)}$ and $P_{\mathbf{x}_i}^{C,(j)}$ for coupon $d_{i,j}$. Additionally, we introduce a new dataset $\mathcal{D}_{t_{end}+1}^{t_{end}+L}$, where the potential outcome distributions $P_{\mathbf{x}}^{I,(j)}$ and $P_{\mathbf{x}}^{C,(j)}$ for each coupon d_j are influenced by competitors' sporadic price adjustments, leading to distributions that differ from those in $\mathcal{D}_{t_{start}}^{t_{end}}$.

Original Formulation. Given $\mathcal{D}_{t_{start}}^{t_{end}}$, let $z_{ij} \in [0,1]$ represent the counterfactual estimation of $y_i^{C,(j)}$, which can be derived from a model trained on this dataset by uplift modeling techniques [6], indicating the order completion probability for opportunity i when applying coupon d_j over $[t_{start}, t_{end}]$. We define $v_{ij} \in \{0,1\}$ as the decision variable indicating whether coupon d_j is applied to opportunity i, ensuring exactly one coupon per request: $\mathbf{v}_i = (v_{i0}, \ldots, v_{i(H-1)})^\intercal \in \{0,1\}^H$. Our objective is to optimize $\{\mathbf{v}_i\}_{i=0}^{N-1}$ to maximize order completions while adhering to a predetermined budget rate, ensuring that the total coupon expenditure does not exceed the product of total earned GMV and the budget rate. Additionally, we aim to develop an adjustment algorithm to enable rapid adaptation to new data distributions in $\mathcal{D}_{t_{end}+1}^{t_{end}+L}$. We begin by formulating a constrained optimization problem on the original dataset $\mathcal{D}_{t_{start}}^{t_{end}}$

$$
\begin{aligned}
&\min_{\mathbf{v}} -\sum_{i=0}^{N-1}\sum_{j=0}^{H-1} z_{ij}v_{ij} \\
&s.t. \sum_{i=0}^{N-1}\sum_{j=0}^{H-1} g_i z_{ij} v_{ij} d_j \le \sum_{i=0}^{N-1}\sum_{j=0}^{H-1} g_i z_{ij} v_{ij} \cdot B \\
&\quad v_{ij} \in \{0,1\}, \quad \sum_{j=0}^{H-1} v_{ij} = 1 \\
&\quad \forall i \in \{0, \cdots, N-1\}, \quad \forall j \in \{0, \cdots, H-1\}
\end{aligned}
\tag{1}
$$

3.2 Optimal Decision Function

The integrality constraint on v is removed by applying linear relaxation. Other constraints are relaxed using the Lagrange multiplier, and the problem is then reformulated in its dual form (see Appendix C.1 for details):

$$
\max_{\lambda \ge 0} \min_{\mathbf{v}} L(\mathbf{v}, \lambda) = \sum_{i=0}^{N-1}\sum_{j=0}^{H-1} z_{ij}(\lambda g_i d_j - \lambda g_i B - 1)v_{ij} \tag{2}
$$

This Lagrangian dual transformation preserves the optimal value of the original problem through the strong duality (see Appendix C.2 for proof). From Eq. 2, the optimal

discount $j^*(\lambda)$ for each opportunity i, given a fixed λ, is:

$$j^*(\lambda) = \underset{j}{argmin}\ z_{ij}(\lambda g_i d_j - \lambda g_i B - 1) \tag{3}$$

where $v_{ij} = 1$ for $j = j^*$ and $v_{ij} = 0$ otherwise (see Appendix C.3 for details). Given this formulation, the maximum of Eq. 2 is obtained by tuning λ over a sum of piecewise linear convex functions. To handle non-differentiable points, we apply a ternary search to iteratively determine λ^*.

However, competitors' aperiodic price adjustments lead to continuous changes in the distributions of $\mathbf{z}$, making the optimized λ^* from $\mathcal{D}_{t_{start}}^{t_{end}}$ non-generalizable. To address this, we decompose the order completion probability z_{ij} (Eq. 4) into $f_{ij}^{(in)}$ (completion probability when auto-selected) and $f_{ij}^{(out)}$ (when not auto-selected). We explicitly model the RSP's In-Range Rate (*IRR*, Def. 1) as w_{ij}, representing the probability of entering the auto-selection range for each coupon, since the *IRR* is the most sensitive part affected by the competitors' sporadic price changes, unlike $f_{ij}^{(in)}$ and $f_{ij}^{(out)}$. Thus, tracking and calibrating the *IRR* distribution in real-time and applying adaptive optimization are critical to maintaining efficiency in $\mathcal{D}_{t_{end}+1}^{t_{end}+L}$. We model this as a Markov Decision Process (MDP) [11], allowing reinforcement learning to adjust λ at each time step in response to the real-time *IRR* changes.

$$z_{ij} = w_{ij} f_{ij}^{(in)} + (1 - w_{ij}) f_{ij}^{(out)} \tag{4}$$

Definition 1. *(In-Range Rate, IRR). For a ride-hailing opportunity i, the IRR represents the probability that an RSP's quoted price is auto-selected by the RHA to maximize user satisfaction. The number of In-Range RSPs depends on the opportunity's context $\mathbf{x}_i$ and follows a top-$K(\mathbf{x}_i)$ lowest-price rule, where $K : \mathcal{X} \to \mathbb{N}^+$ maps features to the number of auto-selections.*

4 Method

In this section, we introduce FCA-RL, a reinforcement learning-based coupon strategy framework tailored for competitive RHA environments. We define a MDP as outlined in Section 4.1.1 and propose to solve it through our proposed **Reinforced Lagrangian Adjustment (RLA)**, with an online module, **Fast Competition Adaptation (FCA)**, integrated to monitor and adapt to the evolving IRR landscape influenced by competitors' sporadic price adjustments. An overview of our framework is demonstrated in Figure 2.

4.1 Reinforced Lagrangian Adjustment (RLA)

4.1.1 Markov Decision Process We model the λ adaptation process as a finite-horizon MDP $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \mathcal{P}, \mathcal{F}, \gamma)$ with the state space $\mathcal{S}$, the action set $\mathcal{A}$, the transition function $\mathcal{T}$, the reward function $\mathcal{R}$, the penalty function $\mathcal{P}$, the full-achievement function $\mathcal{F}$, and the discount factor γ. In our study, we equate the long-term impact by setting $\gamma = 1$.

State. Each state $s \in \mathcal{S}$ at time step t encapsulates: a normalized time slot index, the previous Lagrangian multiplier λ_{t-1}, the coupon campaign status up to t (including the executed budget rate, gap to target, gained GMV, and Finish ROI), and statistics of the latest *IRR* distribution. This state representation supports effective decision-making based on *IRR* changes.

Action. At each time step t, the action $\mathbf{a}_t \in \mathcal{A}$ adjusts λ_{t-1} via Eq. 5, with η controlling the extent of the adjustment. The updated λ_t is then used in Eq. 3 for coupon assignment. lb and up are set to be the lower bound and the upper bound of λ.

$$\lambda_t = \min(\max(\lambda_{t-1} \cdot \eta \cdot a_t, lb), ub) \tag{5}$$

Transition function. In this study, state transitions $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to P(s_{t+1}|s_t, a_t)$ are integrated in **RideGym** and are not explicitly modeled in our method.

Reward function. After calculating the coupon assignment j^* for each passenger's request by Eq. 3 with λ_t at step t, the reward function $r \in \mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is defined in Eq. 6, which reflects the order completions resulting from this coupon assignment.

$$r_t(s_t, a_t) = \sum_{i=0}^{N_t-1} z^t_{ij^*(\lambda_t(s_t,a_t))} \tag{6}$$

Penalty function. At each time step t, issuing coupons increases the probability of order completions, while also consuming the budget. We thus define a penalty function $p \in \mathcal{P} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, demonstrated in Eq. 7, to record the budget consumption.

$$p_t(s_t, a_t) = \sum_{i=0}^{N_t-1} z^t_{ij^*(\lambda_t(s_t,a_t))} g^t_i d_{j^*(\lambda_t(s_t,a_t))} \tag{7}$$

Full-Achievement function. In our scenario, the constraint is a predetermined budget rate over a specified period $[t_{start}, t_{end}]$, rather than a fixed budget amount. Compliance with this constraint depends not only on the expenditure, but also on the total GMV achieved by our coupon allocation strategy. Therefore, we employ Eq. 8 to integrate past cumulative subsidy expenditures, GMV gains (from t_{start} to t), and forecasts up to t_{end} to assess adherence to the budget rate constraint. We propose a Full-Achievement function, demonstrated in Eq. 10, to balance the total achievement of order completions and budget rate compliance. Note that the total order completions for the entire period are normalized by the previously solved optimal order completions using the ternary search method in $[t_{start}, t_{end}]$, denoted by R^*.

$$CR_t = \frac{\sum_{s=0}^{t-1} p_s^{obs} + \sum_{s=t}^{T-1} p_s(s_s, a_s)}{\sum_{s=0}^{t-1} r_s^{obs} + \sum_{s=t}^{T-1} r_s(s_s, a_s)} \tag{8}$$

$$Constraint\_Penalty = e^{max(\frac{CR_t}{CR^*} - 1,\, 0)} - 1 \tag{9}$$

$$F_t = \frac{\sum_{s=0}^{t-1} r_s^{obs} + \sum_{s=t}^{T-1} r_s(s_t, a_t)}{R^*} - Constraint\_Penalty \tag{10}$$

4.1.2 Parameters Initialization of Decision Function First, we pretrain the fundamental models and initialize λ on the dataset $\mathcal{D}_{t_{start}-L}^{t_{start}-1}$. The fundamental models, $\mathcal{W}$, $\mathcal{F}^{(in)}$ and $\mathcal{F}^{(out)}$, are used to estimate the parameters w_{ij}, $f_{ij}^{(in)}$, and $f_{ij}^{(out)}$, respectively, in Eq. 3 during decision making. This allows us to solve for λ^* on $\mathcal{D}_{t_{start}-L}^{t_{start}-1}$ via ternary search to maximize Eq. 2. The resulting λ^* is then used by the actor to adjust the system at the initial time step.

4.1.3 Actor & Critic In this study, we adopt the Actor-Critic framework [7]. At each time slot t, the actor π_θ, parameterized by θ, is a neural network that takes the current state s_t as input and outputs the mean μ_t and standard deviation σ_t of a Gaussian distribution. An action a_t is then sampled from this distribution. The action a_t adjusts the previous λ_{t-1} by Eq. 5. To stabilize λ updates, we adjust λ using Eq.11, where $\delta\lambda_{t-1}$ denotes the previous change margin and λ_t^o represents the λ originally modified by a_t; this adjusted value is then utilized in Eq.3 to determine the optimal coupon allocation for each request in the current time slot. The critic $Q_\gamma(\cdot,\cdot)$ estimates the Full-Achievement F_t based on the state-action pair (s_t, a_t).

$$\lambda_t = \lambda_{t-1} + (\lambda_t^o - \lambda_{t-1})/(1 + \delta\lambda_{t-1}) \tag{11}$$

4.2 Fast Competition Adaption (FCA)

At each time step t, to promptly respond to fluctuations in (*IRR*) induced by sporadic price adjustments by competitors, we propose modeling the initial *IRR* landscape for each candidate coupon as a set of Beta distributions. This facilitates capturing dynamic changes through Bayesian posterior updates. Under the Homogeneous Competitors Assumption (Assumption 1), Proposition 1 holds, validating the use of Beta distributions to model the initial *IRR* distribution.

Assumption 1 *(Homogeneous Competitor Assumption). All RSPs belong to the same service category (e.g., economy), which implies that when they join the RHA at $t = 0$, the prices $p_m(\mathbf{x})$ offered by all RSPs for requests with identical characteristics $\mathbf{x}$ follow an independent and identically distributed continuous distribution with a cumulative distribution function (CDF) F_X.*

Proposition 1. *Let M denote the total number of RSPs for a given request i. The CDF of the $K(\boldsymbol{x}_i)$-th lowest price, denoted as $F_{X_{(K(\boldsymbol{x}_i))}}$ follows a Beta distribution (proof provided in Appendix D). It follows directly that the IRR for request i at $t = 0$ is governed by:*

$$Pr(p_m(\mathbf{x}_i) \leq X_{(K(\boldsymbol{x}_i))}) \sim Beta(K(\boldsymbol{x}_i), M + 1 - K(\boldsymbol{x}_i)). \tag{12}$$

In the following, we demonstrate how the conjugate properties of the Beta distribution facilitate rapid Bayesian posterior updates at time step t, enabling swift adaptation to competitors' price adjustments at the previous time step.

Clustering. A key challenge is that samples are not identical across time steps, so the observed In-Range outcomes at time $t-1$ cannot be directly applied to individual samples at time t. To address this, we employ the K-Means algorithm [8] to cluster requests with similar features, establishing a mapping $f : \mathbb{X} \to \mathbb{C} = \{0, \ldots, S-1\} \subseteq \mathbb{N}$ from the feature space to the cluster labels. This allows us to implement a Bayesian posterior update at time t based on observations from the corresponding cluster at $t-1$. We denote the most recent parameters of the Beta distribution for cluster c as $\alpha_{c,d}$ and $\beta_{c,d}$, for each $d \in \mathbf{d}$.

Initialization of IRR Prior. In our framework, the tracking of the *IRR* distribution commences at time step t_{start}. We average the predictions from model $\mathcal{W}$ over the dataset $\mathcal{D}_{t_{start}-L}^{t_{start}-1}$, stratified by the previously defined clustering labels and coupon levels. This yields the initial parameters $\alpha_{c,d}^{t_{start}}$ and $\beta_{c,d}^{t_{start}}$ for each cluster c and coupon d. We store these in a dictionary $\mathcal{I}$, with the cluster centers as keys and the corresponding $(\alpha_{c,d}^{t_{start}}, \beta_{c,d}^{t_{start}})$ as values to track the *IRR* landscape.

Bayesian Posterior Update. Assuming intra-cluster homogeneity during the FCA process (with an acceptable loss of precision), we model the number of In-Range samples in each cluster under each candidate coupon as a binomial. Specifically, let $N_{c,d}^{t-1}$ denote the number of samples in cluster c at time $t-1$ that received coupon d, and $N_{c,d}^{in,t-1}$ the number of those samples that are In-Range. Given the Binomial–Beta conjugacy, the posterior at time t also follows to a Beta distribution (proof provided in Appendix E). As a result, the parameters at t can be updated by

$$\begin{aligned} \alpha_{c,d}^{t} &= N_{c,d}^{in,t-1} + \alpha_{c,d}^{t-1} \\ \beta_{c,d}^{t} &= N_{c,d}^{t-1} - N_{c,d}^{in,t-1} + \beta_{c,d}^{t-1}. \end{aligned} \tag{13}$$

To address potential stochastic noise in Beta distribution updates due to limited samples per time step, we introduce a window size parameter, l, aggregating In-Range observations from the preceding l time steps (Setting $l = 1$ is consistent with Eq. 13).

4.3 Integration of FCA in RLA

This section describes the integration of FCA into the RLA actor's decision-making process. FCA influences two key aspects of decision-making:

Augmentation of s_t. At time step t, the general *IRR* status tracked by FCA under coupon d, represented by $\frac{1}{S}\sum_c \alpha_{c,d}^t$ and $\frac{1}{S}\sum_c \beta_{c,d}^t$, is incorporated into the state vector s_t. This allows the actor to adjust λ_{t-1} to λ_t based on the updated *IRR* information.

Refinement of w. To preserve heterogeneity across samples, the original predictions, denoted as $\alpha_{(\mathcal{W}(\mathbf{x},d))}^{t,ori}$ and $\beta_{(\mathcal{W}(\mathbf{x},d))}^{t,ori}$, from $\mathcal{W}$ for each individual sample are retained but adjusted using the latest parameters tracked by FCA. The refined predictions for a sample in cluster c at time step t are as follows:

$$\begin{aligned} \alpha_{\mathbf{x},d}^{t} &= \alpha_{(\mathcal{W}(\mathbf{x},d))}^{t,ori} + \alpha_{c=f(x),d}^{t} \\ \beta_{\mathbf{x},d}^{t} &= \beta_{(\mathcal{W}(\mathbf{x},d))}^{t,ori} + \beta_{c=f(x),d}^{t} \end{aligned} \tag{14}$$

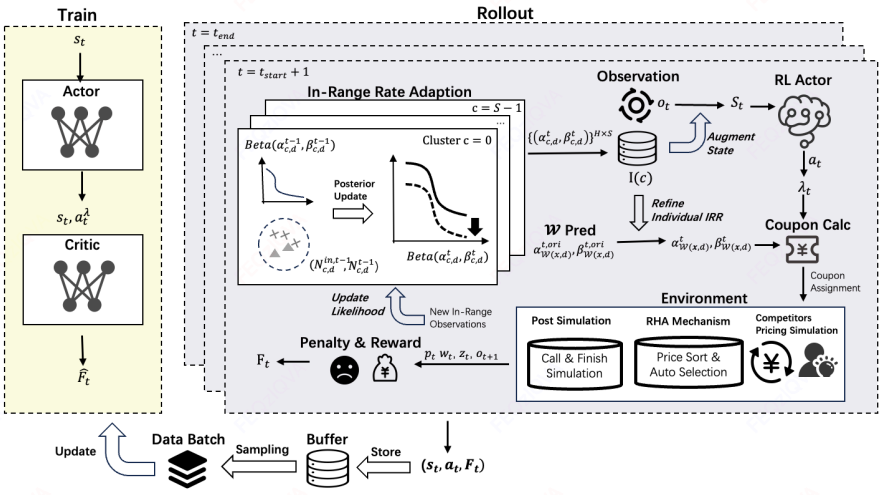


Fig. 2: An overview of our FCA-RL framework which detailing the integration of FCA and RLA at each rollout step. After several rollouts, a batch of tuples (s_t, a_t, F_t) is sampled for reinforcement learning training.

These updated estimates, along with the new λ_t, are incorporated into Eq. 3 to determine the coupon allocation at time step t.

After the coupons are issued, **RideGym** returns the actual In-Range and order completion results for that time step. FCA collects the $N_{c,d}^{in,t}$ and $N_{c,d}^{t}$ to update $\alpha_{c,d}$ and $\beta_{c,d}$ for each cluster and observed coupon. Any unexpected price adjustments by competitors are captured by the updated values of $\alpha_{c,d}$ and $\beta_{c,d}$. The complete process of our FCA-RL is demonstrated in Algo. 1.

5 RideGym

Implementing new subsidy strategies directly in online environments carries the risk of financial loss. To mitigate this risk, we propose the **RideGym** simulation system, which models the complete operations of a ride-hailing aggregator. As depicted in Figure 3, RideGym consists of three core components: the Basic Pricing Engine, the Strategy Engine, and the Post-Pricing Engine. This simulation framework enables the evaluation of different subsidy strategies under diverse market conditions without the potential losses associated with real-world implementation.

Basic Pricing Engine. In this engine, the base price of each order is determined by a configurable price per mile. Price adjustments by one RSP affect others within the RHA. To simulate market competition, we introduce a stochastic price adjustment mechanism by configuring the frequency, lower and upper bounds of each competitor's price adjustment. The actual adjustment amount is sampled from a uniform distribu-

Algorithm 1: FCA-RL Algorithm

Input: B; $\mathcal{W}$; $\mathcal{F}^{(in)}$; $\mathcal{F}^{(out)}$; λ_0^*; R^*; CR^*; $\mathcal{I}$; RideGym $\mathcal{E}$; $\mathbf{d}$; Batch size b;
Output: π_θ; Q_γ; $\mathcal{I}$

1 **while** *not convergent* **do**
2 Set $\lambda_0 = \lambda_0^* + \epsilon$;
3 Observe state $\mathbf{s}_0$ from $\mathcal{E}$;
4 Set $R = 0$; $C = 0$;
5 **for** $t = 0$ *to* T **do**
6 Get an action $\mathbf{a}_t \sim \pi_\theta(\mathbf{s}_t)$;
7 Get λ_t by Eq. 5;
8 Get $\alpha^{t,ori} \in \mathcal{R}^{N_t \times |\mathbf{d}|}$ and Get $\beta^{t,ori} \in \mathcal{R}^{N_t \times |\mathbf{d}|}$ estimated by $\mathcal{W}$;
9 Get $\mathbf{f}^{(in)} \in \mathcal{R}^{N_t \times |\mathbf{d}|}$, $\mathbf{f}^{(out)} \in \mathcal{R}^{N_t \times |\mathbf{d}|}$ estimated by $\mathcal{F}^{(in)}$ and $\mathcal{F}^{(out)}$;
10 Update $\alpha^t \in \mathcal{R}^{N_t \times |\mathbf{d}|}$ and $\beta^t \in \mathcal{R}^{N_t \times |\mathbf{d}|}$ by Eq. 14;
11 Sample $\mathbf{w}^t \in \mathcal{R}^{N_t \times |\mathbf{d}|}$ from $\mathbf{Beta}(\alpha^t, \beta^t)$;
12 Calculate coupon allocation by Eq. 3;
13 Get reward r_t, p_t, s_{t+1}, $\{N_{c,d}^t\}^{|\mathbb{C}| \times |\mathbf{d}|}$ and $\{N_{c,d}^{in,t}\}^{|\mathbb{C}| \times |\mathbf{d}|}$ from $\mathcal{E}$;
14 Augment s_{t+1} by $\{\frac{1}{S}\sum_c \alpha_{c,d}^t\}^{|\mathbf{d}|}$ and $\{\frac{1}{S}\sum_c \beta_{c,d}^t\}^{|\mathbf{d}|}$;
15 Update dictionary $\mathcal{I}$ by Eq. 13 ;
16 Set $R = R + r_t$; $C = C + p_t$; Set $V = 0$; Set $Z = 0$;
17 **for** $k = t + 1$ *to* T **do**
18 Get r_k, p_k by repeat execution of line 6 to 14;
19 Set $V = V + r_k$; Set $Z = Z + p_k$;
20 **end**
21 Set $CR_t = \frac{C+Z}{R+V}$; Set $CP = e^{max(\frac{CR_t}{CR^*}-1,0)} - 1$;
22 Set $F_t = \frac{(R+V)}{R^*} - CP$;
23 Store (s_t, a_t, F_t) in $\mathcal{M}$;
24 **end**
25 Sample b of tuples from $\mathcal{M}$;
26 Update Critic by minimizing the loss $L_{Critic} = \frac{1}{b}\sum_i (F^i - Q(s^i, a^i))^2$
27 Update Actor by maximizing the PPO loss
$L^{\text{CLIP}}(\theta) = \hat{\mathbb{E}}_t \left[\min\left(r_t(\theta)\hat{F}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{F}_t \right) \middle| r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)} \right]$
28 **end**

tion defined by these bounds. The adjusted price for the m-th RSP is calculated as: $p_m^{\text{aft}} = p_m \times (1 + a)$, where a is the sampled adjustment amount.

Strategy Engine. The Strategy Engine generates orders for each time slot, maintaining the desired order count distribution by using multiple normal distributions to approximate the actual distribution. Each RSP can then apply its configurable coupon strategy to assign coupons to the generated orders.

Post-Pricing Engine. The post-pricing engine processes the quoted prices from all RSPs, ranks them in ascending order, and simulates both automatic and passenger selection. Automatic selection by RHA employs a top-K mechanism, while the number of RSPs

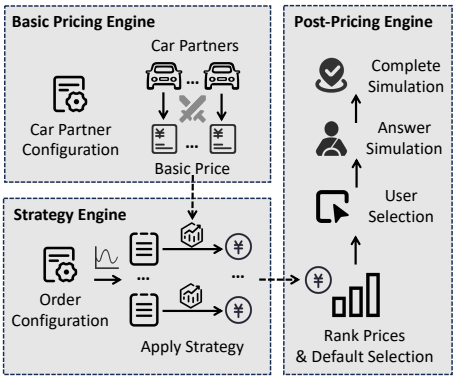


Fig. 3: The pipeline of RideGym.

selected by a passenger, denoted by K', is calculated as follows: given a list of quoted prices $(p_1, p_2, ..., p_M)$ for M RSPs,

$$K' = \text{Clip}\left(K + \log_b\left(\text{density}(p_1, ..., p_M) + b^{-K}\right), 1, M\right)$$

where K is the number of RSPs selected by default, b is an adjustable base, and density$(\cdot)$ calculates the density of the price list. The clipping function ensures that K' stays within the range $[1, M]$. After the order is sent by passengers with selected RSPs, each order is assigned a random supply factor $s \in [0, 1]$ to model supply conditions. Assuming service capabilities $(tp_1, tp_2, ..., tp_M)$ for the M RSPs, the probability of the i-th RSP answering an order is: $P(\text{answer} = i \mid s) = s \cdot \frac{tp_i}{\sum_{j=1}^{M} tp_j}$. The no answer rate of an order is: $P(\text{no answer} \mid s) = 1 - \sum_{j=1}^{M} P(\text{answer} = j \mid s)$. Finally, the simulation includes a random normal distribution to model potential order cancellations.

6 Evaluation

6.1 Evaluation Setup

Dataset Detail. Our RideGym platform generates four different scenarios (referred to as Scenes 1 to 4) under varying configurations. Each scenario consists of three datasets: Pre-train, Train, and Test, with the Pre-train and Train datasets being consistent across all scenarios. In Scenes 1 to 3, price adjustments by other RSPs through changes in base prices occur with increasing frequency. These adjustments directly influence an RSP's ranking within the RHA form and affect the IRR. To assess the robustness of our method in stable pricing environments, we introduce Scene-4, where no RSPs change their prices. See Appendix B for more details on the configuration of the different scenes.

Training and Evaluation Details. In this section, we first describe the training details for each scene. For the FCA-RL model, both $\mathcal{W}$ and $f^{(in)}$ are MLPs trained on the Pre-train dataset. In our implementation, we omit the prediction of $f_{ij}^{(out)}$, assuming that

RSPs placed Out-Range have a negligible probability of being selected by passengers. These pre-trained models are then used to infer the Test dataset without further weight optimization. Further details can be found in Appendix A.1.

Baseline Details. We introduce three baselines to analyze the coupon strategy in RHA: (i) **Optimal Method(OPT)**: This approach utilizes the ground truth of the completion label to derive the optimal solution. (ii) **Primal-Dual Method-A(PDM-A)**: A traditional technique that implements uplift modeling the predictions of the completion probability z_{ij} in an end-to-end fashion. It then solves a constrained optimization problem using the Primal-Dual Method. (iii) **PDM-S**:This method explicitly models w_{ij} and $f_{ij}^{(in)}$, similar to our FCA-RL approach. However, it employs the Primal-Dual Method to derive the solutions without real-time adjustments to the IRR and λ. To construct proper estimates for the parameters in their optimal decision function, we train MLPs to predict w_{ij}, $f_{ij}^{(in)}$ for PDM-S, and z_{ij} for PDM-A. All baselines are solved on the same Train dataset as FCA-RL. Their solutions remain unchanged during evaluation on the Test set.

Evaluation Metric. Here, we introduce three metrics for analyzing algorithms.
Cost Rate Error, CRE quantifies the precision of a coupon strategy in controlling costs. It is defined as the difference between the actual coupon cost rate and the target budget rate, denoted as CR^*. Mathematically, the total CRE is expressed as:

$$CRE = \left| \frac{\sum_i \text{cost}_i}{\sum_i \text{gmv}_i} - CR^* \right| \tag{15}$$

Finish Return Of Interest, FROI assesses the effect of coupon strategies on increasing order volumes, where F, F_0 are the count of finished orders with/without strategy, C is the subsidy cost of the subsidy strategy, A and A_0 are the Average Selling Price (ASP) with/without strategy.

$$FROI = \frac{F - F_0}{\frac{A_0}{A}C} \tag{16}$$

Reinforcement Learning Reward, RLR gauges the order increasing effect and the control ability at the same time:

$$RLR = \frac{\sum_{i=1}^{T} r_i^{obs}}{R^*} - (e^{max(\frac{CR}{CR^*}-1,0)} - 1) \tag{17}$$

6.2 Empirical Results

Here, we summarize our evaluation target to the following research questions. Experiments run on a machine with a 16-core Intel CPU, 40GB RAM, and GeForce RTX 2080 Ti GPU. Results are averaged over five random seeds.

Table 1: Strategy Performance on Three Scenes

	Scene-1			Scene-2			Scene-3		
Method	**CRE**	**FROI**	**RLR**	**CRE**	**FROI**	**RLR**	**CRE**	**FROI**	**RLR**
Optimal	0.000	1.663	0.48	0.001	1.388	0.283	0.000	1.366	0.476
PDM-A	(↑)0.023	1.305	−0.111	(↑)0.032	1.016	−0.499	(↑)0.020	1.062	−0.149
PDM-S	(↓)0.002	1.575	0.127	(↑)0.012	1.228	0.040	(↓)0.007	1.262	0.104
RL-FCA	(↓)**0.001**	**1.578**	**0.168**	(↓)**0.006**	**1.454**	**0.167**	(↓)**0.003**	**1.308**	**0.208**

RQ1: How does FCA-RL compare to the other baselines in terms of performance? We present three key metrics for Scene-1 to 3 across all baselines. Since the ground truth for In-Range and order completion results is generated by our RideGym, we obtain the optimal λ^* using ternary search. As shown in Table 1, our RL-FCA significantly outperforms all baselines on these metrics. It achieves smaller budget rate errors, indicated by the CRE column (up and down arrows correspond to over- and underspending), and higher money efficiency in scenarios with different price adjustment frequencies. In competitive pricing scenarios such as Scene-2 and Scene-3, RL-FCA limits budget rate errors to within 0.3 percentage points (pp), surpassing the second-best method, PDM-S, by reductions of 0.6 pp and 0.4 pp. While occasionally overspending compared to PDM-S, RL-FCA demonstrates higher efficiency, with FROI increases of 0.1% in Scene-1 and 3.6% in Scene-3, defying the typical economic principle that marginal benefits decrease as costs increase.

RQ2: What is the effect of the FCA component? We conducted ablation experiments on FCA modules across various scenarios (Scene-1 to Scene-4, shown in Table 2). Additionally, we analyzed the impact of window size on FCA effectiveness in the most competitive pricing scenario (Scene-3), with results presented in Table 3. Table 2 shows that the FCA module reduces budget rate execution errors in all scenes. It outperforms standard RL without *IRR* adaptation in competitive scenarios (Scene-2 and 3) by 32.2% and 77.4% in RLR. However, in stable price scenarios (Scene-1 and 4), FCA offers no advantage, as the stable distribution may hinder RL training convergence. As shown in Table 3, testing different window lengths in Scene-3 revealed that longer windows improve RL training, with minimal differences beyond a length of 20. We therefore chose a final window length of 24.

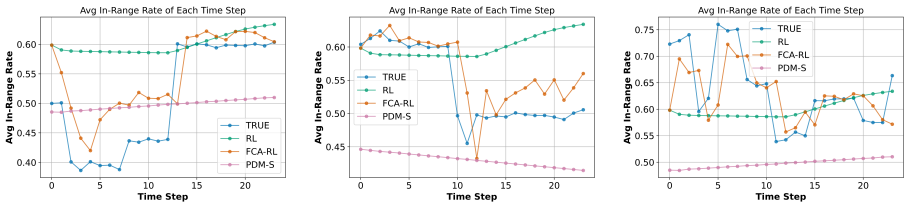
RQ3: How does our RL-FCA outperform the others? Our RL-FCA adapts to market changes by adjusting the Lagrange multiplier (λ) in real-time. Figure 4 illustrates how the true *IRR* distribution shifts due to price adjustments and compares our method's quick adaptation to others relying on training set distribution. To highlight FCA's response to *IRR* distribution changes, we show results with a window size of 1, where FCA is more sensitive but also introduces greater fluctuations in RL training. Figure 5 illustrates how the RL agent adjusts the λ parameter at each time step in response to changes in the *IRR* distribution (first two columns). The third column displays the budget rate implementations for each method. FCA-RL promptly adjusts λ, achieving

Table 2: Ablation Study for FCA

	FCA	CRE	FROI	RLR
Scene-1	w	(↓)**0.001**	1.578	0.168
	w/o	(↓)0.003	**1.653**	**0.197**
Scene-2	w	(↓)**0.001**	1.316	**0.119**
	w/o	(↓)0.011	**1.500**	0.090
Scene-3	w	(↓)**0.003**	1.308	**0.208**
	w/o	(↓)0.02	**1.466**	0.117
Scene-4	w	(↓)**0.002**	1.362	0.125
	w/o	(↓)0.008	**1.453**	**0.195**

Table 3: Study for the Window Size

WS	CRE	FROI	RLR
1	(↓)0.017	1.398	0.133
3	(↓)0.019	1.397	0.143
10	(↓)0.007	1.346	0.182
15	(↓)0.005	1.325	0.192
20	(↓)0.003	1.308	0.203
24	(↓)**0.003**	1.308	**0.208**
30	(↓)**0.003**	1.308	**0.208**



scene-1 scene-2 scene-3

Fig. 4: Average *IRR* adjustment for each method over each time step

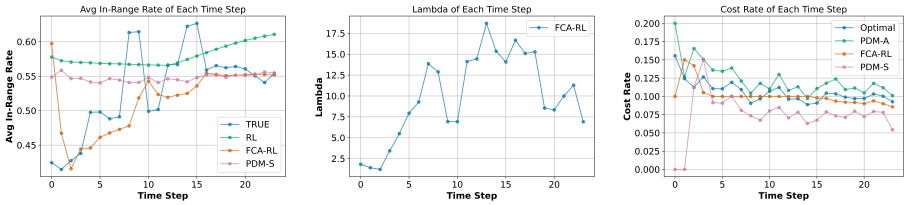
a smoother budget rate and a spending profile that closely aligns with the optimal strategy compared to other methods.

7 Conclusion

We propose FCA-RL, a reinforcement learning-based approach to optimize order acquisition for RSPs under price competition in RHA. Its effectiveness is demonstrated through experiments on RideGym, our self-developed simulation system for managing the full lifecycle of ride requests. In this study, we primarily address temporary data distribution shifts caused by price competition. However, the potential passenger response to coupons and long-term supply-demand dynamics remain underexplored, which we aim to address in future work.

References

1. Bao, Y., Zang, G., Yang, H., Gao, Z., Long, J.: Mathematical modeling of the platform assignment problem in a ride-sourcing market with a third-party integrator. Transportation Research Part B: Methodological **178**, 102833 (2023)



Average *IRR* λ adjusted by RL agent Cost Rate

Fig. 5: Visualization of the performance of our FCA-RL method in Scene-3

2. Cai, H., Ren, K., Zhang, W., Malialis, K., Wang, J., Yu, Y., Guo, D.: Real-time bidding by reinforcement learning in display advertising. In: Proceedings of the tenth ACM international conference on web search and data mining. pp. 661–670 (2017)
3. Chen, J., Xiong, J., Chen, G., Liu, X., Yan, P., Jiang, H.: Optimal instant discounts of multiple ride options at a ride-hailing aggregator. European Journal of Operational Research **314**(2), 718–734 (2024)
4. Chen, M.K., Sheldon, M.: Dynamic pricing in a labor market: Surge pricing and flexible work on the uber platform. Ec **16**, 455 (2016)
5. Dai, J., Li, H., Zhu, W., Lin, J., Huang, B.: Data-driven real-time coupon allocation in the online platform. arXiv preprint arXiv:2406.05987 (2024)
6. Gutierrez, P., Gérardy, J.Y.: Causal inference and uplift modelling: A review of the literature. In: International conference on predictive applications and APIs. pp. 1–13. PMLR (2017)
7. Haarnoja, T., Zhou, A., Abbeel, P., Levine, S.: Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In: International conference on machine learning. pp. 1861–1870. PMLR (2018)
8. Hartigan, J.A., Wong, M.A.: Algorithm as 136: A k-means clustering algorithm. Journal of the royal statistical society. series c (applied statistics) **28**(1), 100–108 (1979)
9. He, Y., Chen, X., Wu, D., Pan, J., Tan, Q., Yu, C., Xu, J., Zhu, X.: A unified solution to constrained bidding in online display advertising. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. pp. 2993–3001 (2021)
10. Ren, K., Qin, J., Zheng, L., Yang, Z., Zhang, W., Yu, Y.: Deep landscape forecasting for real-time bidding advertising. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 363–372 (2019)
11. Szepesvári, C.: Algorithms for reinforcement learning. Springer nature (2022)
12. Wang, H., Yang, H.: Ridesourcing systems: A framework and review. Transportation Research Part B: Methodological **129**, 122–155 (2019)
13. Wang, H., Du, C., Fang, P., He, L., Wang, L., Zheng, B.: Adversarial constrained bidding via minimax regret optimization with causality-aware reinforcement learning. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. p. 2314–2325. KDD '23, ACM (Aug 2023). https://doi.org/10.1145/3580305.3599254, http://dx.doi.org/10.1145/3580305.3599254
14. Wang, H., Du, C., Fang, P., Yuan, S., He, X., Wang, L., Zheng, B.: Roi-constrained bidding via curriculum-guided bayesian reinforcement learning. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. p. 4021–4031. KDD '22, Association for Computing Machinery, New York, NY, USA (2022). https://doi.org/10.1145/3534678.3539211, https://doi.org/10.1145/3534678.3539211

15. Wang, X., He, F., Yang, H., Gao, H.O.: Pricing strategies for a taxi-hailing platform. Transportation Research Part E: Logistics and Transportation Review **93**, 212–231 (2016)
16. Xiao, S., Guo, L., Jiang, Z., Lv, L., Chen, Y., Zhu, J., Yang, S.: Model-based constrained mdp for budget allocation in sequential incentive marketing (2023), https://arxiv.org/abs/2303.01049
17. Zhang, W., Yuan, S., Wang, J.: Optimal real-time bidding for display advertising. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 1077–1086. Association for Computing Machinery, New York, NY, USA (2014)
18. Zhang, Y., Tang, B., Yang, Q., An, D., Tang, H., Xi, C., Li, X., Xiong, F.: Bcorle (λ): An offline reinforcement learning and evaluation framework for coupons allocation in e-commerce market. Advances in Neural Information Processing Systems **34**, 20410–20422 (2021)
19. Zhou, J., Xu, R.: Long-term pricing strategy based on network externalities. Journal of Intelligent & Fuzzy Systems **40**(2), 3035–3043 (2021)
20. Zou, W.Y., Du, S., Lee, J., Pedersen, J.: Heterogeneous causal learning for effectiveness optimization in user marketing (2020), https://arxiv.org/abs/2004.09702