International Journal for Housing Science and Its Applications
Publish August 3, 2025. Volume 46, Issue 3 Pages 213-228

https://doi.org/10.70517/ijhsa46315

Research on Pricing Strategy Optimization and Decision Model of E-commerce Platform Empowered by Intelligent Algorithms

Zhiqiong Bu[1,*]
[1] Management Institute GuangDong Polytechnic Normal University, Guangzhou, Guangdong, 510665, China
Corresponding authors: (e-mail: buzhiqiong@gpnu.edu.cn).

Abstract Online electronic transactions in the big data era form the basis of the application environment for real-time dynamic pricing, and automated dynamic pricing for e-commerce platforms has become a trend. In this paper, the dynamic pricing strategy optimization problem for e-commerce platforms is deeply explored based on deep reinforcement learning methods. The study defines the dynamic pricing problem and transforms this problem into a Markov decision process model, based on which the A3C algorithm is applied to decouple the model to realize the dynamic pricing strategy optimization of e-commerce platform. Experiments show that. The dynamic pricing algorithm for e-commerce platforms based on the A3C method can show better gain results, with an average gain as high as 4033, and is more stable compared to other benchmark algorithms, which can adapt to the complex demand level as well as a large state space. In addition, the effects of gain threshold and loss threshold on the ordering decision of e-commerce platform mainly occur in the low initial inventory region, and the effects on the pricing decision are more significant in the high initial inventory region. This paper has strong application value for pricing strategy optimization and inventory control decision of e-commerce platforms.

Index Terms deep reinforcement learning, A3C algorithm, dynamic pricing, e-commerce platforms

I. Introduction

In recent years, the e-commerce industry has been developing rapidly, especially since 2020.Between 2020 and 2024, the scale of online shopping users in China has risen from 710 million to 974 million, and the proportion of online shopping users in the overall Internet users has reached 74.8%, and the proportion of users who use online payment in the overall Internet users has exceeded 80%.Online shopping and Internet payment have become the proportion of Internet users who use a relatively High application, with the increase of user satisfaction and experience these scales continue to rise [1]-[3]. However, the rapid development has also made the traditional e-commerce facing the market close to saturation, the disappearance of the user dividend, too high customer acquisition cost has become the biggest bottleneck for the development of traditional e-commerce [4]-[6]. With the development of social media, compared with the traditional e-commerce business, social functions into the formation of social e-commerce model can use social features to reach more potential users, and with the use of social relations fission, reduce marketing costs, thus expanding the e-commerce business, and become a breakthrough point for social traffic realization [7]-[10]. Meanwhile, under the network effect, different promotional strategies are utilized to operate with cross-platform service synergy, implicating dynamic price competition, improving sales performance, as well as enhancing platform service quality and user return rate [11], [12]. In addition to reducing the cost of user recruitment through the traffic portal brought by social relationships, the platform can also effectively enhance user stickiness and promote user retention through content operation and community operation [13]. In the current Internet-based social activities are increasingly common, microblogging, WeChat social networking has become an indispensable part of Internet users, this traffic attribute makes the user social function is particularly prominent, which has given rise to a new direction of e-commerce, this development of e-commerce platform pricing has a different impact, resulting in a variety of pricing strategies [14]-[17]. And the most important thing to maintain the platform's continuous and healthy development is to develop the right pricing strategy.

In addition, e-commerce platforms provide intermediary services for suppliers and consumers to facilitate both parties to reach transactions, with typical bilateral market characteristics [18]. For the management of platforms, the establishment of a user base and the utilization of network effects are two important interrelated propositions, both of which have a direct impact on the platform's growth path. The network effect of the platform is similar to the Metcalfe's rule, which is exponential growth, i.e., the value of the Internet is proportional to the square of the number of people connected to the network [19]. In the process of platform accumulating user base and exerting network

effect, the pricing strategy of the platform plays a direct and vital role, and the number of users also affects the transaction volume of the platform, while the profit of the whole e-commerce platform mainly depends on the amount of transaction volume [20]-[22]. In order to improve the platform service quality and performance level, many e-commerce platforms adopt the supplier classification and hierarchical management method as well as multi-channel retailing form, which also produces different pricing strategies, such as Tmall and Jingdong Mall, which classify the suppliers into flagships, specialties, franchisors and other types, introduce dual-channel e-retail, and charge different supplier membership service fees [23]-[25]. Therefore, the scientific pricing strategy is not only to attract more users to join the platform, but in the long run it is also to improve the profit and ensure the long-term development of the e-commerce platform. With the continuous development of the platform, the current pricing method of the platform has been relatively stable, but the pricing formation mechanism and rationality of the platform has not been analyzed in depth [26]. Based on this background, in order to attract and maintain the user base, the platform needs to continuously explore and improve the pricing strategy.

In this paper, pricing strategy optimization and decision-making model construction of e-commerce platforms are realized with the empowerment of deep reinforcement learning. First, the dynamic pricing problem of the e-commerce platform is mathematically modeled and converted into a Markov decision process. Then, the A3C algorithm is utilized to solve the model, and it is compared with the algorithms of SARSA, DDPG and Q-learning to verify the superiority of the proposed algorithm. Finally, the impact of reference price threshold parameters of e-commerce platforms and the results of network target parameters and average round returns are analyzed to study the impact of non-zero threshold parameters on the optimal strategy, which provides a basis for the optimization of dynamic pricing strategies.

II. E-commerce platform pricing strategy optimization and decision-making model

In this chapter, based on the dynamic pricing problem of realistic and specific application scenarios on e-commerce platforms, a mathematical model as well as a Markov decision process is established with reinforcement learning empowerment, and an A3C-based dynamic pricing algorithm is proposed to solve the dynamic pricing problem.

II. A. Deep reinforcement learning

II. A. 1) Enhanced learning and its elemental components

Reinforcement learning is one of the machine learning methods, which is essentially the interrelationship between the four elements of reinforcement learning, namely, state, action, transfer probability, and reward function, and the intelligent body. The interaction flow of the modules of reinforcement learning is shown in Figure 1. The intelligent body obtains the state by perceiving the current environment, for the current state, the intelligent body selects the corresponding action, and the execution of this action makes the environment transfer from the current state to the next state according to the transfer probability, and the environment will also provide feedback to the intelligent body according to the reward function through the change of the state, and the continuation of this interaction constitutes the learning process of reinforcement learning.

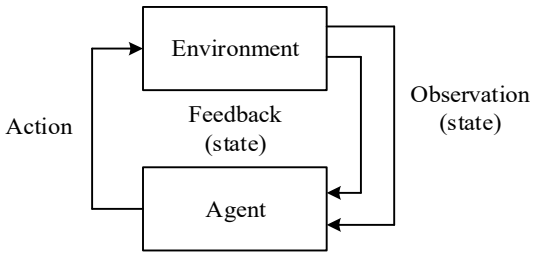

Figure 1: Interactive process of reinforcement learning modules

In a reinforcement learning intelligent-environment interaction process, the following sequence $SO, AO, R1, S1, A1, R2, \ldots$ is generated as the actions, states, and rewards are continuously updated, where S, A, R represent states, actions, and rewards, respectively. That is, a piece of stochastic process is said to have Markovianity if the conditional probability distribution of the states at future moments of the process depends only on the states at the current moment and not on the order of the past states. A process that has this property is made a Markov process and its mathematical expression is as follows:

$$P\left[S_{t+1} \middle| S_t\right] = P\left[S_{t+1} \middle| S_1, \ldots S_t\right] \tag{1}$$

The Mahalanobis decision process can be represented by a tuple (S, A, P, R, γ) where S is the set of states, A is the set of actions that the decision maker can choose from, P is the probability matrix for transferring from the current state to the next state, R is the reward function, and γ is the discount factor, the future reward will be calculated by discount factor to the present value of the current period. The goal of the Mahalanobis decision process is to find the strategy that maximizes the cumulative reward, and the historical information throughout the process is independent of the next state.

There are two most basic ideas of reinforcement learning problems: one is to start from the strategy, directly go to fit the function of the strategy, and train the intelligent body to learn the corresponding strategy by iterating on the strategy gradient, which is called the strategy-based reinforcement learning method, and its typical representative is the strategy gradient method. Another idea is the reinforcement learning method based on the value function, i.e., the Q function is used to approximate the action-value function of the optimal strategy, which is typically represented by Q-Learning.

II. A. 2) Strategies and Value Functions

In reinforcement learning, a state-value function is used to make the following expression for the expected reward that can be obtained by an intelligent being in a good or bad state in the environment, i.e., after choosing a different action according to a certain strategy to go for it:

$$v_x(s) \square \mathrm{E}_r[G_t \square S_t = s] = \mathrm{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s\right] \tag{2}$$

The action-state value function, on the other hand, is the expected reward from a behavior after choosing a behavior in the current state according to a certain policy, and its expression is as follows:

$$q_z(s,a) = \mathrm{E}_\pi\left[\sum_{h=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s, A_t = a\right] \tag{3}$$

By comparing the magnitude of the value function, the good ring of the strategy can be evaluated, and if there are a limited number of strategies π, the optimal strategy π^* can be selected by comparing the value functions under all the strategies, and then the expression is as follows:

$$\forall s, \pi^* = \arg max V^\pi(s) \tag{4}$$

However, the value function of the full number of strategies for comparison will be difficult to achieve in the case of too many strategies or an infinite number of strategies, in this case, the optimization strategy can be updated by iteration, so as to select the optimal strategy, for a strategy $\pi(a \mid s)$, its value function is defined as $Q^\pi(s,a)$, and a new strategy is set up $\pi'(a \mid s)$ with the expression:

$$\pi'(a \mid s) = \arg\max_a Q^\pi(s,a) \tag{5}$$

Execute π', then there is:

$$\forall s, V^{\pi'}(s) \geq V^\pi(s) \tag{6}$$

The method of obtaining the optimal policy is as follows: initialize the policy, calculate the value function corresponding to the policy, update the policy by iterating over the value function, and stop the iteration when the policy converges to the optimum. Methods for updating the policy based on the iteration of the value function include dynamic programming, Monte Carlo method, time-ordered difference method, and so on.

II. A. 3) Strategy Gradient

Unlike algorithms for value function optimization, the strategy gradient algorithm trains an intelligent body by directly optimizing its strategy, i.e., the way in which the intelligent body selects an action in a given state, in the strategy gradient algorithm the current corresponding state of the intelligent body is inputted into the neural network, while the neural network outputs the probability that each action is selected when the action space is discrete, and outputs the probability distribution of the actions when the action space is continuous.

The goal of the strategy gradient is to make the intelligent body learn a strategy that maximizes the cumulative reward through the neural network, which can be expressed by the following equation:

$$\pi = \arg\max_\pi E_\pi\left[\sum_{t=0}^{\infty} r(s_t, a_t)\right] \tag{7}$$

where π denotes the strategy, a_t denotes the t moment intelligent body action, and s_t denotes the t moment intelligent body state. r is the value of the return obtained at moment t, and by accumulating the returns of each moment we get a cumulative return of the whole process from the initial state to the final end, and the strategy that maximizes this cumulative return is the goal of learning.

In the process of learning strategy, the strategy π is a function of θ, denoted as $\pi_\theta(a \mid s)$, which is understood as a mapping of the input to the state s to the output of the action a after the strategy π, and the magnitude of cumulative reward determines the strategy's strength or weakness, so the objective function that defines the function of learning strategy is the cumulative reward value, denoted as:

$$J(\theta) = E_{\tau \sim \pi_\theta(\tau)}[r(\tau)] = \int \pi_\theta(\tau) r(\tau) d\tau \tag{8}$$

where J denotes the objective function and τ represents the whole process from start to finish.

The algorithm of strategic gradient is optimized by gradient descent method. The goal of the method of strategic gradient is to find the optimal neural network parameters θ^* thus maximizing the expectation of the total return function with respect to the trajectory distribution.

The gradient of the objective function of the strategy gradient algorithm becomes by derivation:

$$\nabla_\theta J(\theta) = E_{\tau \sim p_\theta(\tau)}\left[\left(\sum_{t=1}^{\tau} \nabla_\theta \log \pi_\theta(a_t \mid s_t)\right)\left(\sum_{t=1}^{\tau} r(s_t, a_t)\right)\right] \tag{9}$$

where $r(s_t, a_t)$ denotes the reward function of the current state s and the action taken a at the moment t.

In practical applications, it is usually difficult to find the expected value directly, and multiple sampling is often used to continuously approximate the true value through the law of large numbers:

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{i=1}^{N}\left[\left(\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_{i,t} \mid s_{i,t})\right)\left(\sum_{t=1}^{T} r(s_{i,t}, a_{i,t})\right)\right] \tag{10}$$

where N denotes the number of samples.

Finally the parameters are updated:

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta) \tag{11}$$

II. A. 4) Actor-Critic Algorithm

Deep reinforcement learning realizes the fusion of the advantages of both deep learning and reinforcement learning.Actor-Critic algorithm [27] is one of them.This method fuses the advantages of two basic ideas of reinforcement learning, namely the value function estimation and the policy search algorithm, into a single framework by integrating the advantages of the two ideas into a single framework, which outputs the strategy directly through the actor network by using the strategy gradient method, and at the same time, it also outputs the strategy through the value function of the critic network to evaluate the current strategy good or bad. Both the output value function and the strategy are fitted by a deep neural network, and as the algorithm continues to learn iteratively. The strategy will gradually approach the optimum, and at the same time, the evaluation of the value function will be more accurate.

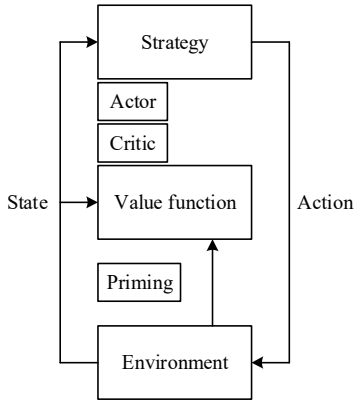

Figure 2: Actor-Critic algorithm

The Actor-Critic algorithm is shown in Figure 2, which consists of two neural networks, the actor network and the critic network. By inputting the current state and outputting the action chosen by the intelligence for that state to fit the strategy model, this is the actor network in the architecture, i.e., the strategy neural network. The critic neural network, on the other hand, fits the value function by inputting the current state and outputting the action value function selected by the intelligence for that state.

II. B.Dynamic Pricing Problem and its Decision Process

II. B. 1) Mathematical Modeling of Pricing Problems

Based on the analysis of the application scenarios of e-commerce platform goods, this paper first establishes a mathematical model of the dynamic pricing problem to analyze the components of its problem. The following assumptions are made for the problem:

n : represents the number of commodities to be sold.

m : represents the maximum number of price changes.

t_i: denotes a time point for changing the pricing ($i=1,....,m$), this paper assumes that the price of the commodity is adjusted every fixed time interval Δ, thus satisfying $t_{i+1}-t_i=\Delta$.

p_{t_i} : denotes the pricing at the moment of time node t_i , and p_{t_i} belongs to a set of prices P , i.e., $p_{t_i}\in P$.

n_{t_i} : denotes the quantity of goods sold in the Δtime period after the pricing p_{t_i} was made at the moment of time node t_i .

r_{t_i}: $r_{t_i}=p_{t_i}n_{t_i}$, denotes the number of items sold in the Δ time period after pricing was made at the moment of the time node t_i . Sales in p_{t_i} time period after pricing was made.

d_{t_i} : represents the actual potential demand in the $t_i+\Delta$ time period.

Therefore, the mathematical model of dynamic pricing in this paper is as follows:

$$\begin{aligned}&\max\sum_{i=1}^{m}r_{t_i}=\sum_{i=1}^{m}p_{t_i}n_{t_i}\\&s.t.0\le n_{t_i}\le d_{t_i}\\&\sum_{i=1}^{m}n_{t_i}\le n\\&p_{t_i}\in P\end{aligned}\tag{12}$$

The two important research variables n_{t_i} and p_{t_i} for this paper's problem can be seen from the dynamic pricing mathematical model in Equation (12). The sales volume n_{t_i} is the response of the environment to the pricing p_{t_i} , and the functional relationship between the two is clarified through environment simulation, and p_{t_i} is the optimal pricing given by the intelligent body by sensing the changes in the environment utilizing a reinforcement learning algorithm to automatically learn the strategy π .

Assuming that a policy π is used to implement the dynamic pricing in this paper, the revenue nature of the dynamic pricing in this paper is analyzed as follows:

If the optimal price that will be offered in each period under the condition of having complete information is $p_{t_i}^*$ and the expected revenue $r_{t_i}^*$, and the price made by the strategy π is p_{t_i} , and the revenue gained is $r_{t_i}^{\pi}$. Then the return performance of the strategy π can be evaluated by a $Regret$, which is defined in this paper as the difference between the cumulative return of the best decision made with complete information and the cumulative return realized through the strategy π in the face of unknown demand information:

$$Regret=\sum_{i=1}^{m}(r_{t_i}^*-r_{t_i}^{\pi})\tag{13}$$

$Regret$ theoretically reflects the merit of a dynamic pricing strategy π . A better strategy π necessarily exhibits a theoretically smaller $Regret$. The intelligence learns to continuously improve the strategy π through reinforcement learning algorithms to make the gap between p_{t_i} and $p_{t_i}^*$ smaller.

II. B. 2) Markov Decision Process for Pricing

Dynamic pricing is a non-stationary sequential decision-making problem, and reinforcement learning is adapted to the modeling of sequential decision-making problems, where the intelligent body makes a pricing decision, and then through the interaction with the environment, captures the changes in the state of the environment, and feeds back the state variables of the environment to the intelligent body, and iteratively and continually learns by trial-and-error by making price adjustments every once in a while in response to the feedback of the stochastic environment, and the intelligent body will gradually learn to the optimal pricing strategy to maximize the cumulative return, i.e., maximize the cumulative return. The premise of reinforcement learning to solve the dynamic pricing problem is to define the dynamic pricing problem as a Markov decision process. For the real-time dynamic pricing problem with finite inventory, the Markov decision process [28] is described in Fig. 3 and defined as follows:

State space: in this paper, the state space is composed of multi-factors $S=[z_1,z_2,\ldots,z_k]$, where each component represents a state influencing factor, k represents the number of state influencing factors, and $s_{t_i}=[z_{t_i1},z_{t_i2},\ldots,z_{t_ik}]\in S$ represents the state at the t_i moment. For example, z_{t_i1} can represent the number of remaining items to be sold at the t_i moment, portraying the current state inventory level, and $z_{t_i2},\ldots,z_{t_ik}$ represents other state factors in specific dynamic pricing application scenarios, which can be perceived and used for decision making by the intelligent body. The intelligent body's action decision dominates the change of some state factors, and the other part of the state factors change dynamically over time without being regulated by the intelligent body, as auxiliary action decision information.

ACTION SPACE: The action space A denotes the pricing set, which can be a continuous or discrete set of prices depending on the specific application of the pricing problem, and $A(s_{t_i})$ denotes the set of prices that the seller can choose from at the moment of t_i when the state is s_{t_i}. $a_{t_i}\in A(s_{t_i})$ denotes a pricing given by the intelligence at the moment t_i when the state is s_{t_i}.

Reward function: for each time step in the sequential decision, the reward function $r_{t_i}(s_{t_i},a_{t_i})$ defined in this paper denotes that, at moment t, with the state $s_{t_i}\in S$, the intelligent body gives pricing as $a_{t_i}\in A(s_{t_i})$ for the immediate gain.

The state transfer probability function: $p_{t_i}(s_{t_{i+1}}\mid s_{t_i},a_{t_i})$ denotes the probability of a state transfer to $s_{t_{i+1}}$ at the moment of t_i, when the state is s_{t_i} and the pricing is a_{t_i}, when the transition to the moment of t_{i+1}. The source of uncertainty in this paper lies in the dynamics of the external influences themselves as well as the resulting changes in customer visits and the strength of customer demand response to pricing a_{t_i}.

For the dynamic pricing problem in this paper, the objective of the reinforcement learning algorithm is to maximize the total expected revenue by continuously optimizing the strategy $\pi: s_{t_i}\rightarrow a_{t_i}$ to reach the state that starts at any $s_{t_1}\in S$, either indirectly or directly:

$$\max E_{\pi}\left[\sum_{i=1}^{m} r_{t_i}(s_{t_i},a_{t_i})\mid s_{t_1}\in S\right] \tag{14}$$

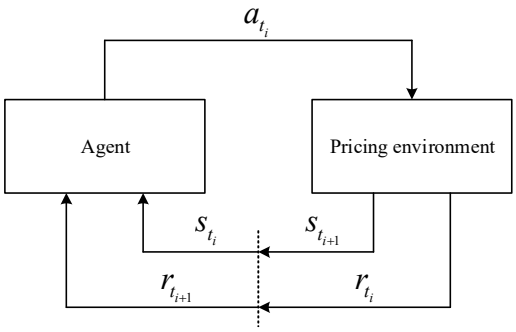

Figure 3: Description of Markov decision process of dynamic pricing

II. C.Dynamic pricing based on the A3C algorithmic framework

The A3C algorithm [29] framework is an Actor-Critic algorithm framework optimized in terms of both model and training architecture.

(1) The A3C algorithm framework employs N-step cumulative discounted returns for $r(\tau)$ estimation to reduce model bias:

$$r(\tau)=\sum_{i=0}^{N-1}\gamma^{i}r_{t+i}+\gamma^{N}v\left(s_{t+N}\right)-v\left(s_{t}\right) \tag{15}$$

(2) An entropy $H\left(\pi_{\theta}\left(s_{t}\right)\right)$ regularity term is introduced for the computation of the strategy gradient to encourage increased exploration of the model:

$$\nabla_{\theta}J(\theta)=\beta\nabla_{\theta}H\left(\pi_{\theta}\left(s_{t}\right)\right)+E_{\tau\square\pi_{\theta}(\tau)}r(\tau)\sum_{t=0}^{\infty}\nabla_{\theta}\log\pi_{\theta}(a_{t}\left|s_{t}\right.) \tag{16}$$

where β denotes the regularization factor.

The pricing set used in the A3C dynamic pricing algorithm in this paper is a continuous discount rate. Equation (16) demonstrates that the loss function of the strategy network consists of two parts: strategy loss and information entropy loss, while the probability distribution of generating actions on the continuous action space conforms to the Gaussian distribution, and the output layer of the strategy network is expressed as the mean μ and the variance σ^{2} of the distribution, then the strategy loss and information entropy loss are as follows:

Strategy loss:

$$\log\pi_{\theta}(a_{t}\left|s_{t}\right.)=-\frac{\left(a-\mu\right)^{2}}{2\sigma^{2}}-\ln\sqrt{2\pi\sigma^{2}} \tag{17}$$

Information entropy loss:

$$H\left(\pi_{\theta}\left(s_{t}\right)\right)=\frac{1}{2}\ln\left(2\pi\sigma^{2}\right)+1 \tag{18}$$

(3) The biggest improvement of the A3C algorithm framework is that it abandons the previous reinforcement learning approach of using Replay Buffer to solve the problem of strong correlation between intelligences and the environment to generate sequences. The A3C algorithm framework adopts a multithreaded concurrent approach to interact with the environment for the purpose of collecting samples quickly, and adopts an asynchronous training approach to update the parameters.

The network architecture of A3C is shown in Figure 4. Aiming at the diversity of the environment due to the differences of each e-shop in the dynamic pricing environment of e-commerce platforms, it is ensured that each thread interacts with different environments, which can effectively avoid generating highly similarity sample data. The global model and the neural network of P worker thread models have the same structure. The role of each worker thread model is to interact with the environment and compute the gradient values of the parameters, but it does not update the worker's own thread model, but is used to update the global model separately. The role of the global model is to save the parameters and to update its own parameter values to the P-worker thread models at regular intervals. The neural network model of the value network and the strategy network inputs state s, the value network outputs state values $v(s)$, and the strategy network outputs $\pi(a|s)$.

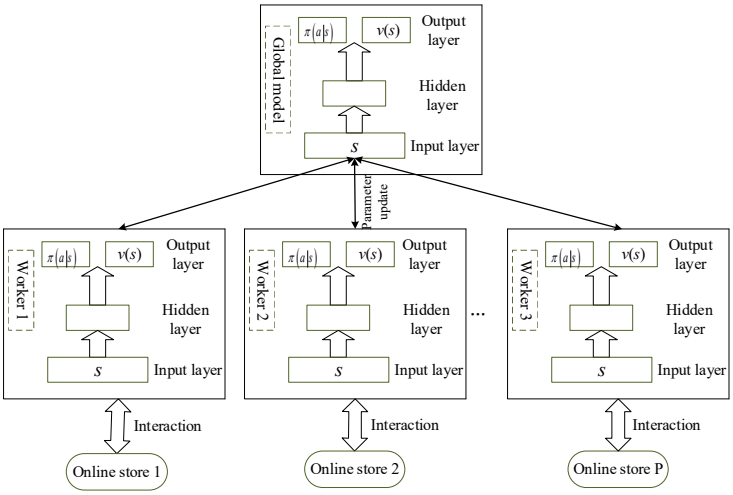

Figure 4: Network architecture of the A3C

In this paper, the inputs to the A3C deep reinforcement learning dynamic pricing algorithm are the individual state components of the dynamic pricing Markov decision process, and thus the same internal neural network architecture within which the higher-order features are extracted is used, with the main difference being in the expression of the information in the output layer. On the one hand, the output layer of the Actor-Critic network architecture expresses the policy and state-value functions, whereas the output layer of the Q-network architecture expresses the state-action values for each action. On the other hand, the loss functions of the two are different, and the specific expressions of the network parameters are different.

III. Dynamic pricing and inventory decision-making for e-commerce platforms based on A3C

In order to verify the effectiveness of the A3C algorithm in dynamic pricing strategy optimization and inventory decision-making of e-commerce platforms, this paper evaluates the performance of the algorithm and analyzes the impact of threshold parameters on ordering decisions and pricing decisions of e-commerce platforms.

III. A. Performance evaluation of the A3C algorithm

In this section of the experiment, specific experimental parameters will be set, and specific numerical results will be obtained through the deep reinforcement learning algorithm A3C interacting with the designed simulation environment. In this way, the performance results of the algorithm in the simulation application environment will be analyzed and it will be judged whether the algorithm can be applied to the real environment.

III. A. 1) Experimental parameterization and hyperparameter selection

This section begins with the setup of the neural network based on the A3C algorithm. The algorithm features two neural networks with the same structure and their parameters are θ and $\hat{\theta}$. Each neural network has two hidden layers and uses the Relu activation function. Set the capacity size of the experience pool N to 12000, and importance sampling will be performed each round. The number of interval steps for updating the target network C is set to 400.

(1) Generalized parameter settings for the experiment

1) The product life cycle l is set to $\{4,5,6\}$, and the lead time L is $\{0,1,2\}$.

2) The product price set adopts a discrete pricing set in the form of a discount rate, with a price discount $discount=\{0.1,0.2,\ldots 0.9,1.0\}$ and an original pricing $P_{base}=30$.

3) The quantity of products ordered takes the form of $d+x$, setting the range of values for x to $\{-30,\ldots,30\}$.

4) For the $\varepsilon-greedy$ strategy, set the exploration value ε will gradually decay to the threshold value during the learning process, initially $\varepsilon_{init}=0.9$, and linearly decreasing in iterations until $\varepsilon_{end}=0.1$:

$$\varepsilon=\varepsilon-\frac{\varepsilon_{init}-\varepsilon_{end}}{episodes\times\beta} \tag{19}$$

(2) Hyperparameter selection

Since the A3C algorithm introduces a deep neural network, and the hyperparameters of the model generally need to be set manually, different from the parameters estimated from the data, the selection of hyperparameters will directly affect the stability and convergence of the training of the intelligent body, and is related to the training effect, so how to select hyperparameters to ensure the effectiveness of the strategy is a key issue of reinforcement learning in the practical application. In this section, we will make a comparison for the learning rate and gamma value hyperparameters, and analyze their influence on the training effect.

1) The learning rate $\alpha=0.005,0.001,0.0001$ of A3C-based dynamic pricing algorithm was tested respectively with the same other parameters. The experimental results of model training at different learning rates are shown in Figure 5. The learning rate is a very important parameter in deep learning model training, which relates to the degree of updating of the neural network parameters, which in turn affects the convergence of the model. Learning rate parameter is set too high or too low will lead to poor model performance, learning rate is too high will probably lead to the model does not converge, when the learning rate is selected as 0.005 algorithm of the first period of the oscillation amplitude is very large, and the performance of the later period is not as good as the algorithm that selects the learning rate of 0.001. If the learning rate is too low, the algorithm will take too long to train and need more time to converge.

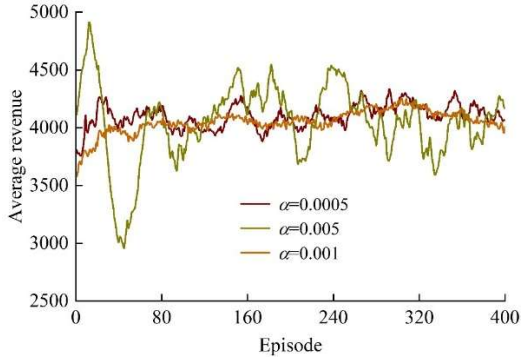

Figure 5: Experimental results of model training under different learning rates

2) Fixing the other parameters, the cases of gamma value $\gamma = 0.9, 0.95, 0.99$ were tested respectively, and the experimental results of model training under different gamma values are shown in Figure 6. For gamma values, the higher the parameter setting, the more the intelligences will focus on the overall future gains and hardly focus on the immediate short-term gains, thus training will be difficult and slow.

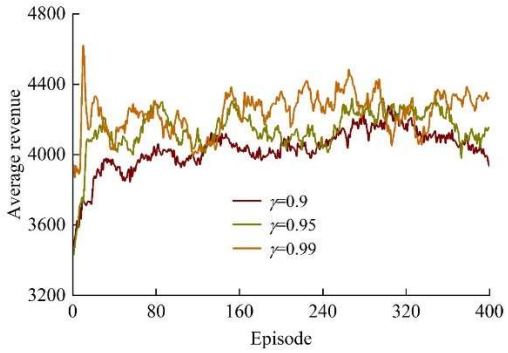

Figure 6: Experimental results of model training with different gamma values

III. A. 2) Algorithm Performance Experiments

Reinforcement learning methods applied to the field of revenue management has been studied, Q-learning algorithm and SARSA algorithm are more widely used. The DDPG algorithm of deep reinforcement learning methods has also been applied in the field of e-commerce, this paper will be based on the above parameter settings, and the following benchmark model for comparison experiments:

(1) DDPG algorithm: it is a strategy learning method that incorporates deep neural networks into deterministic behavioral strategies, in this paper, we set the two Actor networks and the two Critic networks of DDPG to be two-layer fully-connected layers, the activation function to be the Relu function, and the learning rate, the gamma value, and the exploratory value ε to be set to the previous attenuation strategy.

(2) Tabular Reinforcement Learning Methods: The State-Action-Reward-State-Action (SARSA) algorithm and Q-learning algorithm are mainly used as the benchmarks. The SRASA algorithm solves the optimal policy through homogeneous time-ordered differential updating, while Q-learning is a heterogeneous algorithm that updates the optimal action value estimation in a different way from the SRASA algorithm. As a value function iterative reinforcement learning method, the Q-learning algorithm will record the values of the state-action pairs through the Q-value table and update the Q-table based on the bellman equation. In this paper, the learning rate, gamma value and other hyperparameters of SRASA algorithm and Q-learning algorithm are set to be consistent with the above. The comparison of A3C-based dynamic pricing algorithm for e-commerce platforms with DDPG algorithm, SRASA algorithm and Q-learning algorithm gain results is shown in Fig. 7. Aggregate to get the model gain performance comparison is shown in Table 1.

Combining Fig. 7 and Table 1, it can be seen that the A3C algorithm has the best performance in terms of gain, which is in the interval of [3230,4202], and the average gain is as high as 4033. Followed by the DDPG algorithm and Q-learning algorithm, and the SARSA algorithm has the lowest gain. The A3C algorithm also outperforms the other algorithms in terms of stability, with both the DDPG and Q-learning algorithms having larger oscillations compared to the A3C algorithm.

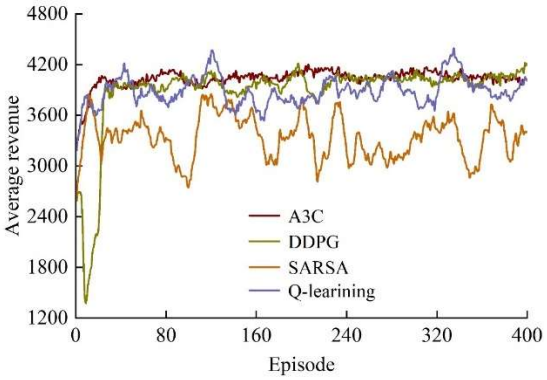

Figure 7: Comparison results of A3C, DDPG, Q-learning and SARSA experiments

Table 1: Comparison of model income performance

Model	Average revenue	Revenue upper bound	Revenue lower bound
A3C	4033	4202	3230
DDPG	3888	3878	2591
Q-learning	3887	3830	3182
SARSA	3339	3462	2639

In reality, the changes in demand for goods on e-commerce platforms are extremely complex, and various factors cause the demand to show strong stochastic and non-stationary fluctuation changes.The A3C dynamic pricing algorithm is able to solve the problem of dimensional catastrophe, and also provide e-commerce platforms with near-optimal ordering and pricing strategies. It can be seen that the dynamic pricing algorithm for e-commerce platforms based on the A3C method has a very wide range of application value.

III. B. Analysis of threshold parameters

Analyzing the influence of different threshold parameters on the optimal decision value of the e-commerce platform has the following difficulties: the threshold parameters are two-dimensional data, the decision space is a two-dimensional space, the state space is likewise a two-dimensional space, and the confluent six-dimensional space is difficult to be clearly represented by a single image, so in this section, through the slicing method in the data dimensionality reduction method, the matching analysis of the data is carried out separately, so that we can get the control variables under the condition of the the same experimental results as the original data, and provide experimental basis for the decision-making of the e-commerce platform when considering the threshold parameter.

(1) The influence of gain threshold on ordering decision under different initial inventory levels is shown in Figure 8. It can be known that:

(1) The higher the initial inventory level, the ordering level increases with it, which is completely in line with common sense, and the lowest ordering level is roughly maintained near 0.

(2) The effect of gain threshold on ordering decision is mainly concentrated in the low inventory level, for different gain thresholds, there will be fluctuations in ordering decision for extreme gain thresholds, for example, when the gain threshold is larger, the ordering level will increase accordingly.

(2) The effect of gain threshold on pricing decision at different initial inventory levels is shown in Fig. 9, which can be observed:

1) The higher the initial inventory level, the lower the pricing level, which is consistent with common sense.

2) The effect of GAIN threshold on pricing decision is significant. The general rule is that when the initial inventory level is high, the pricing will fluctuate dramatically with the change of the GAIN threshold, and this price volatility also side-steps the important role of the GAIN threshold on the pricing decision. On the other hand, when the initial inventory level is low, the gain threshold has less influence on the pricing decision of the e-commerce platform, i.e., when the initial inventory level is low, the ordering decision of the e-commerce platform is more influenced by the gain threshold, and with the increase of the initial inventory level, the pricing decision of the e-commerce platform is gradually influenced by the gain threshold.

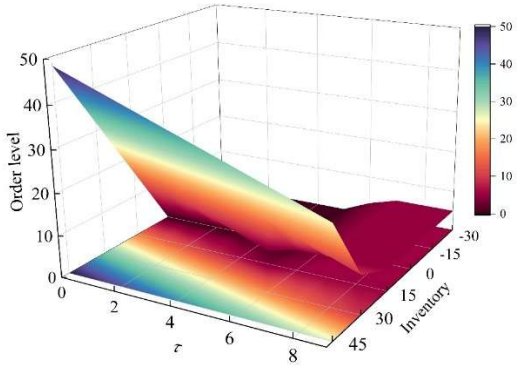

Figure 8: Effects of gain threshold on ordering decisions at different initial inventory levels

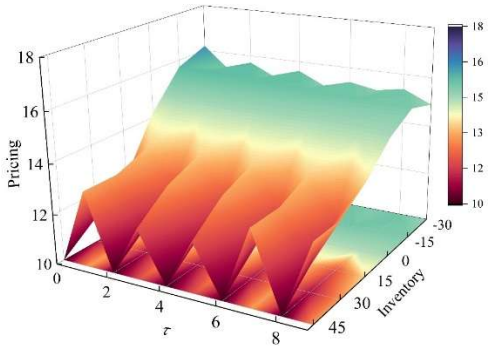

Figure 9: Effects of gain threshold on pricing decisions at different initial inventory levels

(3) The effect of gain threshold on ordering decision under different reference prices is shown in Fig. 10, and the following conclusions are obtained from the analysis:

1) When the gain threshold is low, the ordering level decreases with the increase of the reference price, which is called the reference price regular period.

2) As the gain threshold increases, in a certain region, the ordering level will no longer change with the reference price, called the reference price non-sensitive period.

3) When the gain threshold increases to a certain level, the ordering decision of the e-commerce platform increases abruptly to a certain level, and the fluctuation of the reference price affects the ordering decision of the e-commerce platform, which is called the reference price fluctuation period.

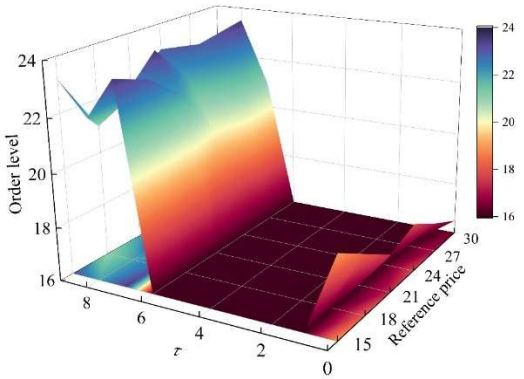

Figure 10: Effects of gain threshold on ordering decisions at different reference prices

(4) The effect of gain threshold on pricing decisions under different reference prices is shown in Fig. 11, and the following conclusions can be drawn from the analysis:

1) With the increase of the reference price, the pricing decision of the e-commerce platform is a tendency to increase.

2) Gain threshold has a significant impact on the pricing decision of e-commerce platform. In the face of different reference price levels, the difference of gain threshold significantly determines the pricing level of the e-commerce platform, for example, when the gain threshold is between 3-5, the e-commerce platform's pricing level is very low, even if it presents a certain degree of regularity under the influence of the reference price, but due to the drastic change of the price, the e-commerce platform is forced to consider the role of the gain threshold on its own pricing decisions in the interval and the influence effect.

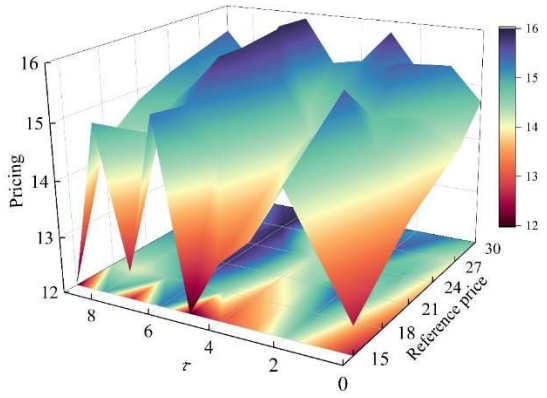

Figure 11: Effects of gain threshold on pricing decisions at different reference prices

(5) The effect of loss threshold on ordering decision at different initial inventory levels is shown in Fig. 12, which can be observed:

1) The higher the initial inventory level, the ordering level increases with it, which is fully consistent with common sense, and the lowest ordering level is roughly maintained near 6.

2) The effect of loss threshold on ordering decision is mainly focused on low inventory level, for different loss threshold, there will be fluctuation of ordering decision for extreme loss threshold, for example, when the loss threshold is small, the ordering level will be reduced accordingly.

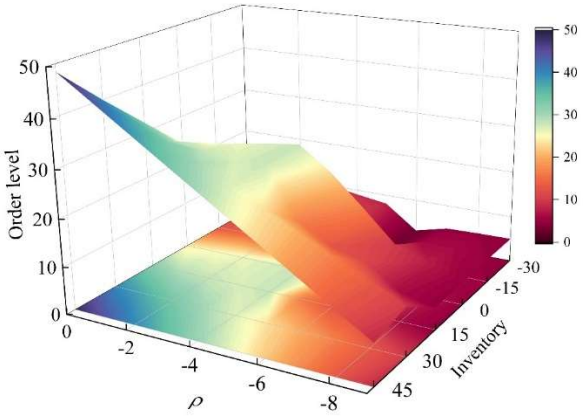

Figure 12: Effects of loss threshold on ordering decisions at different initial inventory levels

(6) The effect of the loss threshold on the pricing decision at different initial inventory levels is shown in Figure 13, and observing the trend shows that:

1) The higher the initial inventory level, the lower the pricing level, which is consistent with common sense.

2) The effect of loss threshold on pricing decision is very significant. The general rule is that when the initial inventory level is high, the pricing will fluctuate dramatically with the change of the loss threshold, and this price volatility also side-steps the important role of the loss threshold on the pricing decision. On the other hand, when the initial inventory level is low, the loss threshold has less influence on the pricing decision of the e-commerce platform, and with the increase of the initial inventory level, the pricing decision of the e-commerce platform is gradually influenced by the loss threshold.

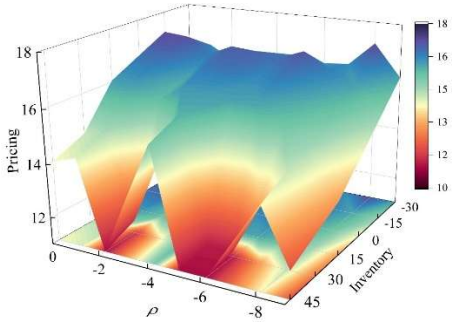

Figure 13: Effects of loss threshold on pricing decisions at different initial inventory levels

(7) The effect of loss threshold on ordering decision under different reference prices is shown in Fig. 14, and the following conclusions are obtained from the analysis:

1) When the loss threshold is high, the ordering level decreases with the increase of reference price, which is called the reference price regular period.

2) When the loss threshold is reduced to a certain level, the ordering decision of the e-commerce platform suddenly drops to a certain level, and the fluctuation of the reference price affects the ordering decision of the e-commerce platform, which is called the reference price fluctuation period.

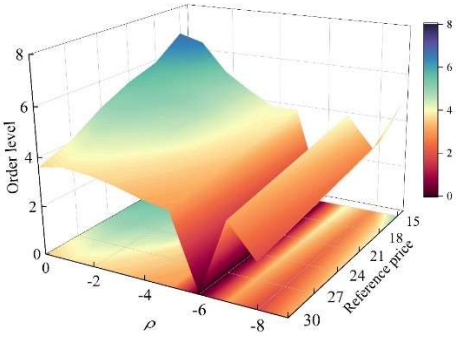

Figure 14: Effects of loss threshold on ordering decisions at different reference prices

(8) The effect of loss threshold on pricing decision under different reference prices is shown in Fig. 15, and it can be observed that:

1) With the increase of the reference price, the pricing decision of the e-commerce platform is a tendency to increase, but it is not very significant.

2) The loss threshold has a significant impact on the pricing decision of the e-commerce platform. For example, when the loss threshold is between -2 and -4, the pricing level of the e-commerce platform is very low, even if it shows a certain regularity under the influence of the reference price, but because the price changes are too drastic, the e-commerce platform has to consider the effect of the loss threshold on its own pricing decisions and its impact effect.

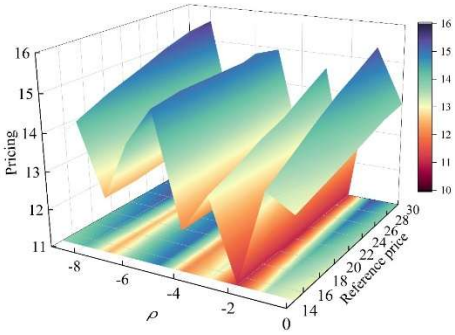

Figure 15: Effects of loss threshold on pricing decisions at different reference prices

III. C. Objective Function and Average Round Returns

The loss function of the algorithm contains three parts, which are the substitution function $J(\theta)$, the mean square error of the value function V_{loss}, and the entropy term $Entropy$, and the change of the average value of these three sub-objective functions with the number of updates is recorded during the training process, and then the intelligence of each episodic situation is counted. Body's average gain value and Ground truth value, and the obtained results are shown in Figures 16~19.

Combined with Figures 16~19, the following conclusions can be drawn:

(1) The substitution function gradually becomes larger with the number of updates, and finally tends to stabilize. The value of the substitution function represents the objective function of the gradient update, and the stability of its value can reflect the stability of the strategy learned by the actor network of the intelligent body.

(2) The mean square error of the Critic network decreases with the number of updates, and finally converges to 0. The Critic network is able to provide a benchmark value for the training output of the actor network, and the smooth error at the end indicates that the policy environment is relatively smooth, and the training has already converged and reached a stable output state.

(3) At the beginning of the update, the value of entropy increases, indicating that the intelligent body has a higher motivation to explore the action space, and then with the gradual smoothing of the update, the intelligent body's motivation to carry out exploration gradually decreases, and the value of entropy decreases and eventually stabilizes. It is worth mentioning that, due to the Beta distribution used in the randomized strategy, the probability of the action being at an extreme value is much greater than 1, and thus the recorded entropy value is negative.

(4) During the first 3000 times of training, the average gain value of the intelligences gradually increases and approaches the GROUND TRUTH value, roughly increasing from 90 to near 220. After that, the training process is smooth, the whole action space has been explored, and the stochastic strategy learned by the intelligent body gradually converges to the optimal strategy, and then after 3000 times, the average gain value of each cycle is basically the same as that of the ground truth value, and it can be seen from the image that the A3C algorithm is more stable, and does not have the phenomenon of a sudden collapse of the gain that is often found in the PG class of algorithms.

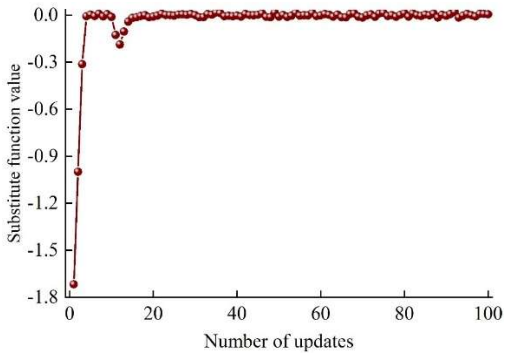

Figure 16: The change of surrogate function with the number of updates

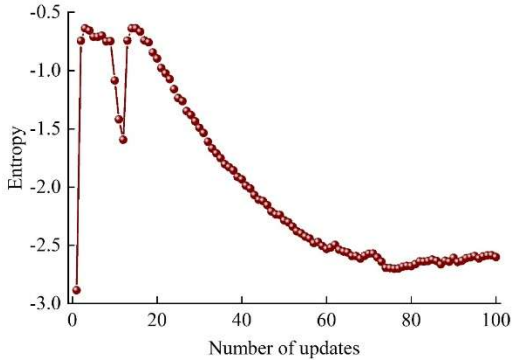

Figure 17: The change of entropy with the number of updates

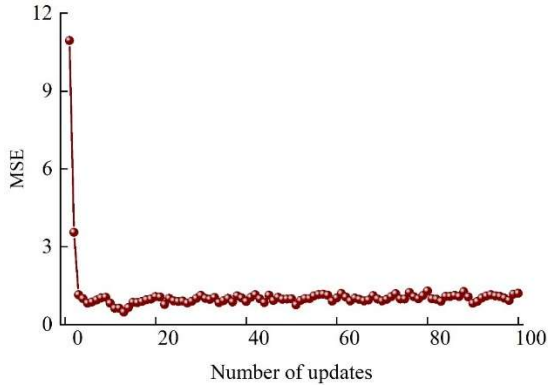

Figure 18: The change of MSE with the number of updates

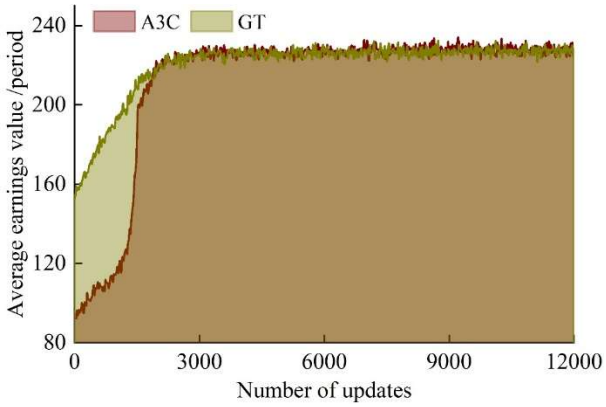

Figure 19: The change of mean episode profits of A3C and Ground truth

IV. Conclusion

This paper constructs a dynamic pricing model based on the A3C algorithm to provide a modeling tool for pricing strategy optimization and decision-making of e-commerce platforms, and evaluates the model performance and analyzes the impact of non-zero threshold parameters on the optimal strategy.

Compared with the DDPG algorithm, Q-learning algorithm and SARSA algorithm, the A3C algorithm used in this paper has the best performance in terms of gain, which is distributed in the interval of [3230,4202], and the average gain is as high as 4033. At the same time, the A3C algorithm is better than the other algorithms in terms of stability, with smaller oscillation amplitude, and the average gain value of each cycle is basically the same as the value of ground truth after 3,000 iterations of training. basically the same as the ground truth value.

Through the threshold parameter analysis, this paper draws the following conclusions:

(1) In the vast majority of cases, the higher the initial inventory level of an e-commerce platform merchant, the higher its ordering level and the lower its selling price.

(2) As the reference price increases, e-commerce platform merchants demonstrate a tendency to increase the selling price.

(3) The effects of gain thresholds and loss thresholds on merchants' ordering decisions mainly occur in regions with low initial inventory levels. If the gain threshold is large, the ordering level is high, and if the loss threshold is small, the ordering level is low. And when the gain threshold is small or the loss threshold is large, the ordering level of merchants decreases with the increase of the reference price.

(4) The gain threshold and the loss threshold have a great impact on the pricing decisions of merchants on e-commerce platforms, and if the initial inventory is high, the pricing decisions of the merchants fluctuate drastically with the changes of the two thresholds.

About the Author

Zhiqiong Bu was born in Yiyang, Hunan. China, in 1975. She obtained a Master's degree from Wuhan University in China. She currently working at the Management Institute, GuangDong Polytechnic Normal University. She main research direction is ecommerce, management information system, etc.

References

[1] Pentina, I., Zolfagharian, M., & Michaud-Trevinal, A. (2022). Toward a comprehensive scale of online shopping experiences: a mixed-method approach. Internet Research, 32(3), 814-842.

[2] Ebrahimabad, F. Z., Yazdani, H., Hakim, A., & Asarian, M. (2024). Augmented reality versus web-based shopping: how does Ar improve user experience and online purchase intention. Telematics and Informatics Reports, 15, 100152.

[3] Peneder, M., Bilek-Steindl, S., Bärenthaler-Sieber, S., Bock-Schappelwein, J., & Charos, A. (2025). Business use of online platforms: Competition, satisfaction and willingness to pay. Technology in Society, 102887.

[4] Fahmy, H. (2021). How technological emergence, saturation, and rejuvenation are re-shaping the e-commerce landscape and disrupting consumption? A time series analysis. Applied Economics, 53(6), 742-759.

[5] Li, X., Guo, H., Jin, S., Ma, W., & Zeng, Y. (2021). Do farmers gain internet dividends from E-commerce adoption? Evidence from China. Food Policy, 101, 102024.

[6] Li, X., Wu, Z., Huang, Q., & Liu, J. (2024). Pay more attention to consumers: exploring customer acquisition strategies of large third-party sellers on e-B2C market. Industrial Management & Data Systems, 124(4), 1558-1581.

[7] Farrag, M., & Nasr, M. (2017). Social Media and Mobile for Measurable Results in E-Commerce. American Journal of Computer Science and Information Engineering, 4(5), 43-51.

[8] Attar, R. W., Almusharraf, A., Alfawaz, A., & Hajli, N. (2022). New trends in e-commerce research: Linking social commerce and sharing commerce: A systematic literature review. Sustainability, 14(23), 16024.

[9] Semenda, O., Sokolova, Y., Korovina, O., Bratko, O., & Polishchuk, I. (2024). Using social media analysis to improve E-commerce marketing strategies. International Review of Management and Marketing, 14(4), 61-71.

[10] Peng, Y., & Lu, L. (2024). The pre-purchase search channel and purchase behavior: Role of social commerce vs traditional e-commerce. Journal of Retailing and Consumer Services, 81, 104024.

[11] Zhu, S., Yang, X., Zhou, W., & Cao, P. (2025). Dynamic price competition and promotion strategy with cross-network effect: Implications for e-commerce platforms. Journal of the Operational Research Society, 1-15.

[12] Yan, K., Hua, G., Cheng, T. C. E., Choi, T. M., Dong, J. X., & Li, X. (2023). Optimal pricing and quality decisions under cooperative promotion of cross-market service platforms. IEEE Transactions on Engineering Management, 71, 10091-10115.

[13] Gao, X., Yee, C. L., & Choo, W. C. (2022). How attachment and community identification affect user stickiness in social commerce: a consumer engagement experience perspective. Sustainability, 14(20), 13633.

[14] Shao, J., Li, P., & Zhang, M. (2024). Traffic transfer between social media and E-commerce platform: the role of social media affordances. Behaviour & Information Technology, 1-14.

[15] Prabowo, N. A., Pujiarto, B., Wijaya, F. S., Gita, L., & Alfandy, D. (2021). Social network analysis for user interaction analysis on social media regarding e-commerce business. International Journal of Informatics and Information Systems, 4(2), 95-102.

[16] Guan, L., Chen, H., Ma, H., & Zhang, L. (2022). Optimal group-buying price strategy considering the information-sharing of the seller and buyers in social e-commerce. International Transactions in Operational Research, 29(3), 1769-1790.

[17] Zhang, H., Sui, R., & Zha, X. (2025). The key opinion leader introduction and pricing strategy for live streaming e-commerce platforms considering the impact of network effects. Journal of Retailing and Consumer Services, 82, 104077.

[18] Sun, L., Li, L., & Liu, B. (2020). The Analysis of Different Industries under the Bilateral Platform Environment in E-Commerce Enterprise. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems.

[19] Singh, B. (2023). A Comparative Doctrinal Study of E-Commerce Platforms: Predatory Pricing and Network Effects. Available at SSRN 4665560.

[20] Li-feng, M. U., Fang-yuan, W. A. N. G., & Li-hui, C. H. E. N. (2020). Research on pricing strategy of E-commerce platform based on strategic consumers. Operations Research and Management Science, 29(10), 225.

[21] Xu, B., Yao, Z., & Wu, S. Y. (2021). Pricing strategies for a bundled channel with services network effects. International Journal of Production Research, 59(10), 3152-3168.

[22] Yang, Z., Shi, Y., & Yan, H. (2017). Analysis on pure e-commerce congestion effect, productivity effect and profitability in China. Socio-Economic Planning Sciences, 57, 35-49.

[23] Xi, X., & Zhang, Y. (2023). The interplay between marketplace channel addition and pricing strategy in an e-commerce supply chain. International Journal of Production Economics, 258, 108807.

[24] Ji, Y., Li, Y., & Tang, W. (2022). Service investment and pricing strategies in e-commerce platforms with seller competition. International Journal of Information Systems and Supply Chain Management (IJISSCM), 15(1), 1-21.

[25] Fu, Y., Gu, B., Xie, Y., Ye, J., & Cao, B. (2021). Channel structure and differential pricing strategies in dual-channel e-retail considering e-platform business models. IMA Journal of Management Mathematics, 32(1), 91-114.

[26] Momin, U., & Mishra, P. (2024). E-commerce management and ai based dynamic pricing revenue optimization strategies. Migration Letters, 21(S4), 168-177.

[27] Jingpan Bai, Yifan Zhao, Bozhong Yang, Houling Ji, Botao Liu & Yunhao Chen. (2024). Joint Optimization Strategy of Task Migration and Power Allocation Based on Soft Actor-Critic in Unmanned Aerial Vehicle-Assisted Internet of Vehicles Environment. Drones,8(11),693-693.

[28] Xiaoming Duan, Yagiz Savas, Rui Yan, Zhe Xu & Ufuk Topcu. (2025). On the detection of Markov decision processes. Automatica,175,112196-112196.

[29] Hongjian Wang, Wei Gao, Zhao Wang, Kai Zhang, Jingfei Ren, Lihui Deng & Shanshan He. (2023). Research on Obstacle Avoidance Planning for UUV Based on A3C Algorithm. Journal of Marine Science and Engineering,12(1),