Understanding the Early Dynamics of Pricing Algorithms

Sahil Patil

Abstract

This paper investigates early convergence dynamics of reinforcement learning pricing algorithms amidst firm competition, focusing on adjustments to annealing schedules, competition strategies, and hyperparameters. Employing Q-learning, the study explores the optimization of the net present value of expected future rewards, revealing insights into the impact of various parameters on convergence. Findings highlight the dominance of firms with higher capacity, the potential for stable results with fewer interactions, and the influence of annealing schedules on convergence speed and stability. The research contributes to understanding the complexities of AI-driven pricing in competitive markets, offering implications for algorithm design and regulatory frameworks.

1 Introduction

Firm competition has evolved beyond mere price wars to encompass the realm of Artificial Intelligence (AI) pricing. These advanced pricing algorithms use competitive pricing strategies across sectors such as e-retail (Chen et al., 2016), (Assad et al., 2020), and stock markets. AI pricing algorithms have come under scrutiny due to legal and ethical concerns surrounding price discrimination and have bolstered research into protecting consumers from such pricing mechanisms (Calvano et al., 2020). To grasp the genuine benefit to consumers, it is vital to understand the pricing mechanisms among these AI-driven pricing algorithms.

Some pricing algorithms can autonomously learn without prior knowledge of their operating environment by exploring their environment during training phases and then providing the best outcomes depending on each circumstance[1]. This has yielded concerning outcomes, as firms employing such algorithms have been observed engaging in collusion despite lacking prior awareness of the market dynamics (**?**). In applied work, there has also been evidence of firm margin improvements after adopting an algorithmic pricing software (Assad et al., 2022). Therefore, the persistence of collusive behavior in these algorithms demands extensive scrutiny and investigation.

Most pricing algorithm studies study convergence dynamics and provide a post-convergence analysis. Although some hyperparameter decisions are taken from literature, like the learning

[1]These are called learning algorithms. Evidence in literature has provided both sides of the arguments on using these. Johnson et al. (2023); Calvano et al. (2021); Klein (2021)

rate and discount factors, convergence criteria are highly debated. Modifying hyperparameters might impact algorithms' convergence, yet little to no quantifying evidence exists. This study takes a step in that direction by looking at early convergence dynamics to understand what drives the deceitful behavior of the firms and how convergence is impacted when hyperparameters are modified. To do this, we study the effect of changing annealing schedules and competition strategy on the early dynamics of the reinforcement learning pricing algorithm.

2 Literature Review

"AI-powered algorithms" have become increasingly common in marketplaces with near-perfect price monitoring, like Amazon and gasoline markets. This has led to a general source of efficiency and potential concerns and risks associated with tacit collusion. Assad et al. (2021) provide an economic overview of these concerns and some solutions to regulate these market algorithms. One such instance occurred in Assad et al. (2022), who identified collusion behavior in the German gasoline pricing market, leading to a significant price increase compared to the socially optimal.

Seele et al. (2021) identify ethicality concerns associated with personalized pricing set by algorithms and identify potential algorithmic transparency as one of how there could be an increase in accountability of firms that design these algorithms. Different market structures have led to collusive behavior with Klein (2021) identifying collusive behavior under sequential pricing setting and **?** theoretically identifying collusive behavior of Q-learning algorithms in a duopoly course competition setting. These findings provide ample evidence for understanding the intricacies of what causes the algorithms to collude, which is sparse in this literature and something this study tries to identify.

3 Methodology

3.1 Q-learning

Q-learning is an unsupervised learning algorithm that aims to maximize the net present value of expected future rewards in an environment with interactions between multiple agents. It assumes that the agent remembers a value of every strategy from past experiences, known as the Q-value, and that the agent chooses using probability (depending on some function) which action to play. The set of actions chosen will be the Markov Chain that is considered.

The algorithm in this paper is trained to maximize expected discounted profits:

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \delta^t \pi_{it}\right] \tag{1}$$

which is identical to **?** where π_{it} are the period t profits of firm i, and $\delta \in (0,1)$ is the discounting rate. Profits are defined by price times the quantity where the costs are considered trivial[2].

[2]The costless assumption allows for inexpensive algorithm training but can be positive when competing against a pre-trained algorithm.

The value in the Q-matrix for the reinforcement learning pricing algorithm is:

$$Q(a_i^t, s^t) = \mathbb{E}(\pi|a_i^t, s^t) + \delta\, \mathbb{E}[\max_{a_i^{t+1} \in A} Q(a_i^t, s^{t+1})|a_i^t, s^t] \tag{2}$$

The first term is firm i's state t period payoff of choosing action a, and the following is the expected future value. Additionally, we assume that states and actions are finite; hence, the Q-function is $|A| \times |S|$ matrix. For the iteration process , the computer chooses an arbitrary Q_{i0} and updates it using the following rule:

$$Q_{i,t+1}(a^t, s^t) = (1-\alpha)Q_{i,t}(a_i^t, s^t) + \alpha \left[\pi_{i,t} + \delta \max_{a_i^t \in A_i} Q_{i,t}(a_i^{t+1}, s^{t+1})\right] \tag{3}$$

and for all other $s \neq s_t$ and $a \neq a_i$, we have $Q_{i,t+1}(a_i^t, s^t) = Q_{i,t}(a_i^t, s^t)$ that is their q-values remain the same in time t and $t+1$.

3.2 Environment

The timeline of the game is as follows:

- At time 0, firms are provided with a random shock and a starting price p_0, which is independent of all choices[3]
- All firms then face a shock d_t, and the only choice in that period that firms can make is quantity.
- The choice of firm's output is made using an annealing function (which is ϵ-greedy) where with probability ϵ, firms choose to play the action that optimizes their q-value for that given state (i.e., choose the output that has the highest q-value) and with probability $1-\epsilon$ choose an action randomly from the available action space.
- In the next period, firms also observe the past price that they received, i.e., p_{t-1}, their competitor's output in that period $q_{j,t-1}$ and the shock that they received, i.e., d_{t-1} leading to full information about one prior period

This shows that the game is of complete information with imperfect monitoring, leading to the fact that once the exploration period ends, firms, in the long run, should converge to the competitive output regardless of the choice made by the other firm.

The action and state space in the study are set to be discrete, where the action space for player i starts from 70 to 105 with a step of 2.5. This helps with ease of computation and, , the ability to compare results identified by **?**. The shocks are randomly assigned between 290 and 310 with a Cournot competition to decide pricing:

$$p_t = d_t - (a_i + a_j)$$

[3]This decision should not matter given if the chain is recurrent and positive (Robert and Casella, 2004)

3.3 Annealing Schedules

Annealing schedules (AS) were initially introduced by Metropolis et al. (1953) to solve optimization problems to find global optima where multiple local optima are present. Simulated Annealing, the technique within each schedule, uses an explore and exploit algorithm to learn the state space (Exploration Phase) and then provide the best response for each potential state in the state space (Exploit Phase). These exploration and exploitation phases help the algorithm learn and optimize the dynamic programming problem and identify the optimal solution for each state space.

? use the standard exponential AS which defines the $\epsilon-$greedy function as:

$$\epsilon_t = e^{-\beta t}$$

where β decides the exploration rate and t is the specific iteration period. This AS exponentially reduces the exploration rate, starting with maximum exploration and slowly reaching exploitation with a convergence criterion of no change in decision after 500,000 iterations. In this study, we adopt a similar exploration criterion, except we allow for multiple exploration and exploitation criteria and reduce the convergence criterion to only 20,000 iterations. The idea of multiple exploration and exploitation phases is that there is a possibility of finding global optima early without the need for longer interaction periods. This study tests three exploitation phases in 200,000 iteration settings, where exploitation ranges are 25,000-60, on 0, 100,000-130,000, and 180,000-200,000.

4 Early Dynamics

The learning rate and the discount rate for the Q-learning algorithm remain the same, but to study the early dynamic, I employ some modifications to the existing environment in terms of the annealing schedule by reducing the total iterations from 2 million to 200,0000 and changing the annealing schedule so that it reduces linearly for half the amount of iterations converging to zero at 100,000 iterations and then exploiting the results for the remaining 100,000 iterations. This gives rise to the following $\epsilon-$greedy function:

$$\epsilon_t = \frac{1}{\#\text{Iterations} - 100,000}$$

This allows us to see if convergence occurs early, a concern brought by **?** as one of the disadvantages of such algorithms. An additional adjustment made to this environment is the strategies that both firms in the environment choose to employ. This involves adjustments made to the action space. In the economic sense, this helps us understand the effect of fundamental world markets where firms are heterogeneous and have capital constraints that reduce their ability to produce the highest quantities available for profit maximization. Additionally, this scenario helps us understand entrant dynamics in the early scenarios where competitor one has been in the market for a set amount of time. In contrast, competitor two only begins at time 0, which leads to the reduced strategy set due to capacity constraints.

This study modifies action spaces and adjusts the frequency of interactions within each iteration. This method addresses the challenge highlighted in the literature regarding the

necessity of increased interactions to reach the global optimum. A higher number of interactions suggests a deeper understanding of the action space; however, pinpointing the threshold in this context could significantly enhance the algorithm's training speed, potentially achieving a nearly tenfold acceleration if fewer than 100 interactions suffice to attain comparable stability and results compared to algorithms requiring ten times more interactions.

Finally, the study delves into the impact of reducing exploration time on algorithm convergence rates. Algorithms lacking assumptions about the environment necessitate extensive exploration to comprehend all state spaces within it. However, if interactions are sufficiently robust, global optima can still be attained irrespective of exploration. Thus, altering the hyperparameter can yield outcomes akin to those documented in existing literature, resulting in supracompetitive profits but potentially at the expense of increased variance.

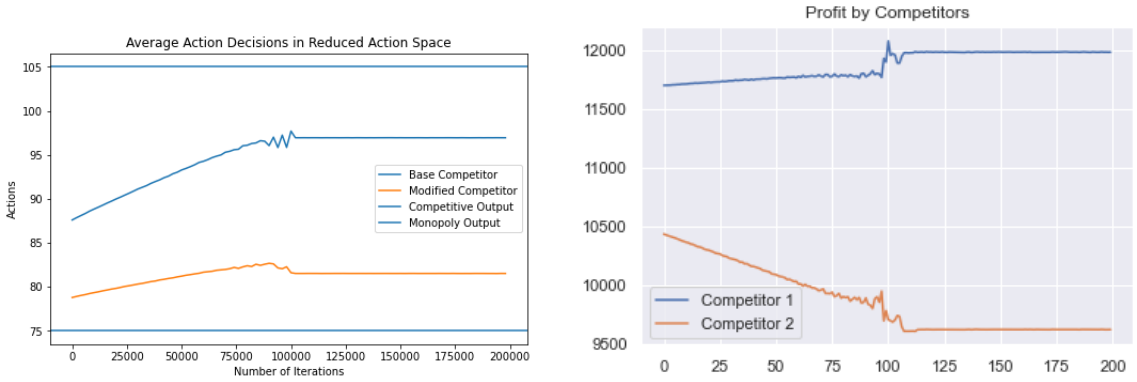
4.1 Adjustment in Action Space

Adjusting action space for one of the competitors allows us to capture two major topics. Firstly, this provides a more realistic understanding of the natural world markets. Firms do not necessarily choose the highest quantity available due to capital constraints and usually make quantity decisions based on or near competitors' choices. Secondly, this allows us to capture a new entrant effect where one firm has a smaller subset of actions available due to its limited information about the market and is competing with a much more informative firm. It also allows us to research strategies that can provide random variables with stochastic dominance properties. In this case, when one firm has significantly fewer options for quantities, their decision has little to no impact on competitor 1's profits and action decisions.

In implementation, Firm 1 competes with the entire action space, while Firm 2 is constrained to actions equal to or less than the average action within the space. The outcomes of these differing action spaces are presented in Figures 1a and 1b. The firm operating within a limited action space fails to attain higher profits than its counterpart with unrestricted access. This indicates that a confined action space yields more favorable outcomes for consumers but produces sub-optimal performance for the firm. Results of the exploitation for the base and modified player are in Table 1. It is evident that the mean action decision for the base player arrives quickly at a low point of 89 in the exploration phase and remains in the range of 96 with little to no deviation once the exploitation phase begins at 100,000 iterations. Similarly, the modified player chooses a significantly lower output in the exploration phase due to capacity constraints and remains there (as a local optimum). Results of the base player are considerably closer to **?**, whereas the player with reduced action space struggles to reach optimal values due to capacity constraints.

Player	Range	Mean	Std Dev	% Diff to Calvano
Base	(0, 25,000)	88.98	0.92	-0.69
	(100,000, 130,000)	96.99	0.31	8.25
	(180,000, 200,000)	96.94	0.11	8.19
Modified	(0, 25,000)	79.41	0.42	-11.36
	(100,000, 130,000)	81.53	0.26	-8.99
	(180,000, 200,000)	81.52	0.14	-9.01

Table 1: Statistics for Reduced Action Spaces



(a) Action results for differing action spaces (b) Profits results for differing action spaces

4.2 Effect of Reduced Interaction

This section examines how increased interaction impacts Markov Chains. Economically, this variation with reduced interaction helps us model settings with low frequency of changes, like gas prices, which do not update in real-time, or seller-requested pricing in online markets, which remain static for specific periods. To explore this, firms face limitations on the extent of their competition in each iteration.

Figure 2 and 3 illustrate the influence of varying interaction intensity on the convergence rate of Markov Chains for all periods and during the exploitation phase only. The results reveal that as interactions increase, the resulting Markov Chain becomes notably more stable, resulting in a higher overall quantity chosen. Notably, within each chain, the collusive outcome of 70 is observed with significantly higher frequency in chains with only one interaction per iteration (6.76%), and this occurrence decreases exponentially to 1.81% for a Markov chain with 300 interactions in each iteration. This underscores the significant role of low interactions in mitigating collusive behavior.

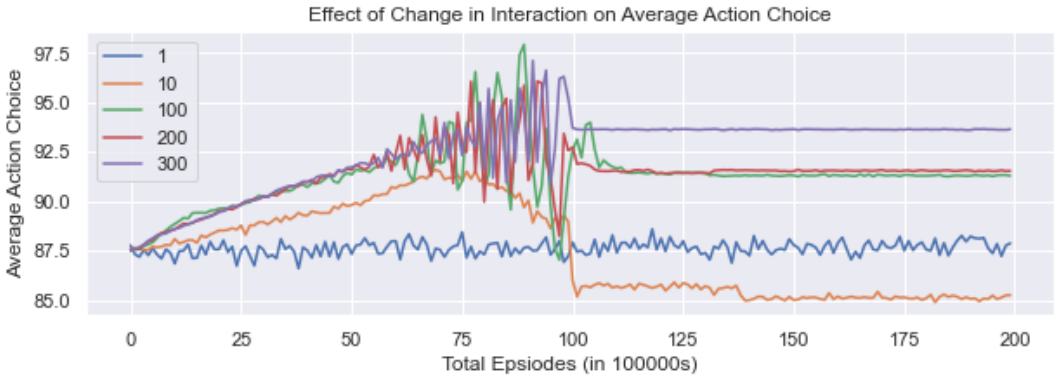


Figure 2: Figure illustrative the effect of change in interactions on Q-learning pricing algorithm

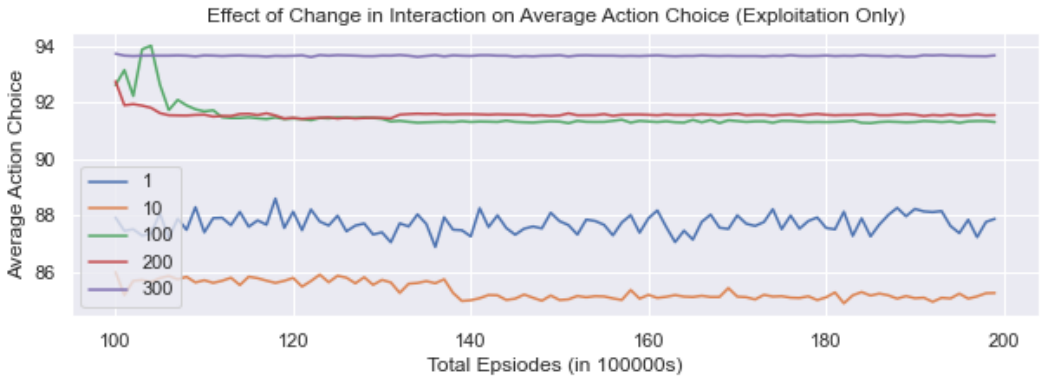


Figure 3: Figure illustrative the exploitative phase of the effect of change in interactions on Q-learning pricing algorithm

4.3 Changes in Annealing Schedule

Creating a model with multiple exploration and exploitation phases involves tuning the algorithm's decay rate. This adjustment causes the exploration rate to decrease to 0 three times and then increase to 1 twice. Notably, the ascent to 1 occurs at double the descent speed.

Figure 4 illustrates the outcomes of various exploration and exploitation phases. The exploration rate decreases at 0-50,000, 75,000-125,000, and 150,000-200,000. Table 2 displays the standard deviation of actions for three sections characterized by increased variability. The table unmistakably reveals collusive behavior early in the interaction, with minimal deviations.

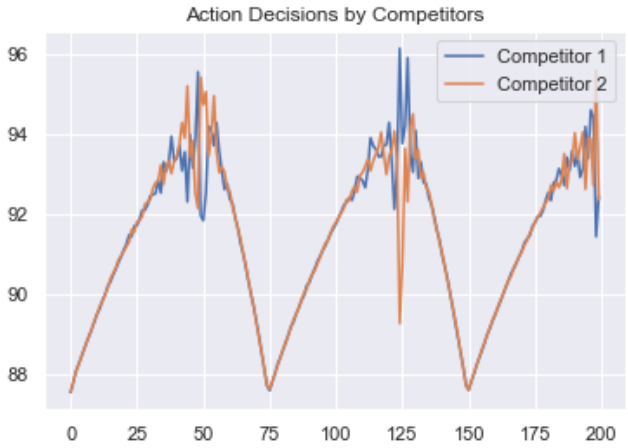


Figure 4: Figure illustrates action choices made by competitors when there are multiple explore exploit phases.

Sections	Mean	Standard Deviation
25,000-60,000	93.06	2.15
100,000-130,000	93.23	2.33
180,000-200,000	93.13	2.33

Table 2: Statistics on Exploit Section

4.4 Changes in Beta

Figures 5 and 6 depict the average action decisions and variance, respectively, comparing the algorithm's performance with a base value of β against a scenario where β is reduced by a factor of 10. The findings suggest that the algorithm with the accelerated β converges to the optimal solution more rapidly on average. However, it exhibits more significant variability while remaining at the optimal solution than the original β, which continues exploration during the initial 200,000 iterations.

Regarding variance, the version with reduced exploration rapidly decreases variance initially but stabilizes at a relatively high level, selecting among five possible outcomes centered around a mean action decision of 88 by the end of the iteration cycle. In contrast, the original version exhibits a more consistent reduction in variance throughout the interaction period, steadily decreasing until completion.

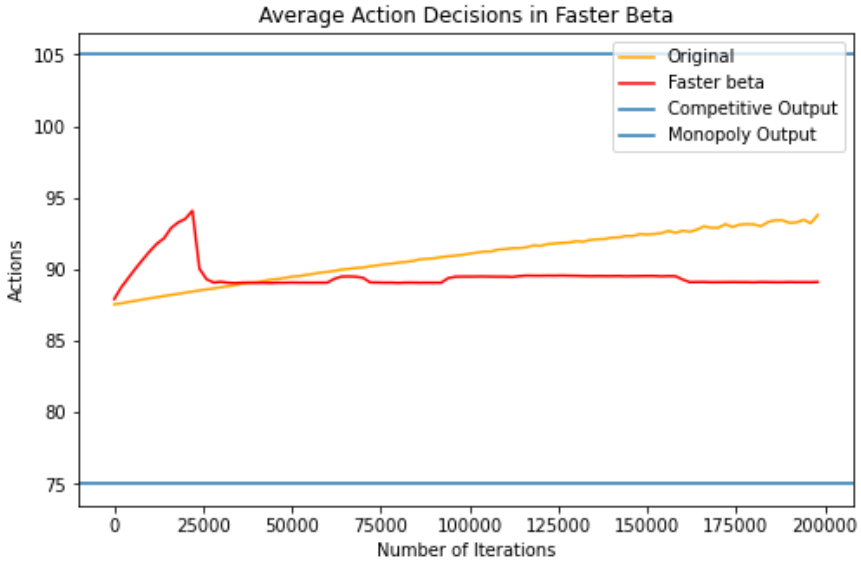


Figure 5: Average action decisions when β hyperparameter is modified.

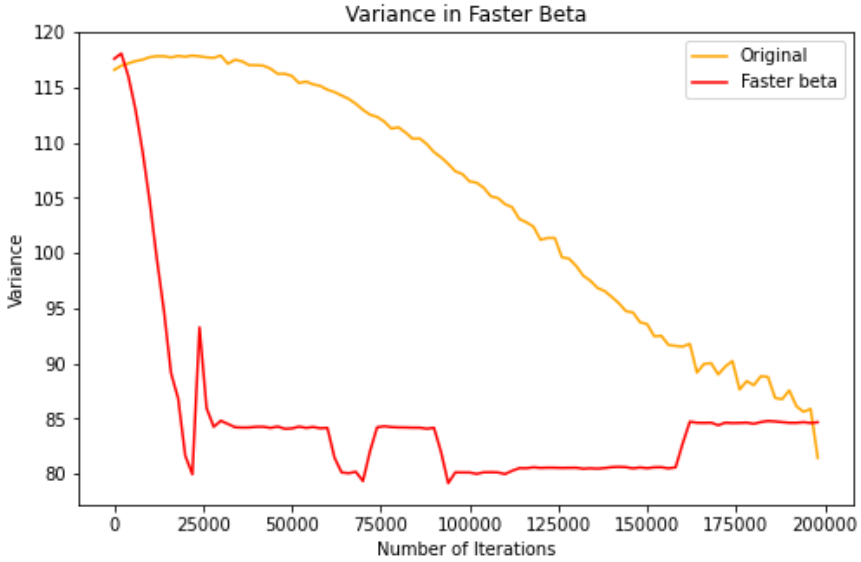


Figure 6: Average variance in action decisions when β hyperparameter is modified

5 Discussion

The early dynamics section contains several notable findings. The adjustment in action space, interaction rates, and annealing schedules all have different impacts on the convergence to the global optima. On a general level, this provides us with evidence that potential convergence criteria are a function of parameters instead of separate entities.

The adjustment in action space yields two significant findings. Firstly, when firms with different capacity constraints train their pricing algorithms against each other, the firm with

higher capacity consistently outperforms the ones with lower capacity. This is evidenced by the fact that, for every state space in the Q-matrix, the firm with a more extensive set of action spaces consistently selects values higher than those chosen by the firm with a lower capacity.

Secondly, this scenario can also be observed as a situation involving an entrant and an existing firm, where the firm with a greater set of actions possesses a deeper understanding of the market. This results in consistently higher profits by producing beyond the capacity of the constrained firm at all interaction levels. This highlights that convergence can be achieved with minimal interactions when one of the pricing algorithms is inherently constrained by capacity.

Reducing the number of interactions yields several insights regarding the adjustments required for grid searching optimal parameters. The figure associated with the effect of reduced interaction illustrates that deficient interaction levels impede sufficient exploration. However, even with only 100 interactions compared to the base of 1000 interactions, the exploitation phase remains remarkably stable with minimal deviation after 120,000 interactions. This suggests that one of the hyperparameters that can be grid-searched for optimal value is the number of interactions within each episode. While computationally intensive, employing a parallel process can facilitate the identification of the required optimal interaction values.

The plots depicting multiple explore and exploit phases in the early dynamics sections reveal intriguing findings. In theory, one would expect the variance (and consequently the standard deviation) to decrease as the number of interactions increases. However, the standard deviation remains stable throughout the 200,000 iterations. Additionally, the average action space in each section remains consistent at 93, leading to supracompetitive profits for the firms. This performance is comparable to that observed in **?**, who found an average decision choice of 89.60.

6 Conclusion

In the evolving landscape of firm competition, integrating Artificial Intelligence (AI) pricing algorithms has shifted dynamics beyond traditional price wars, impacting various sectors such as e-retail, gasoline, and stock markets. While these algorithms promise efficiency, price discrimination and collusion concerns have prompted extensive research into regulating their behavior to protect consumer interests.

This study delved into the early dynamics of reinforcement learning pricing algorithms to understand the drivers of collusive behavior and the impact of hyperparameter modifications. Significant insights emerged by examining the effects of changing annealing schedules, competition strategies, and interaction rates.

Adjusting action spaces revealed disparities in performance between firms with varying capacity constraints, shedding light on the competitive advantage of firms with broader action sets. Furthermore, reducing interaction levels provided insights into optimal parameter settings, which quantified the importance of interaction frequency in achieving convergence at a similar level of convergence outcome.

Exploring multiple explore-exploit phases elucidated the nuanced interplay between exploration and exploitation in reaching optimal outcomes. Notably, the stability of standard

deviation and average action space throughout iterations underscored the robustness of the algorithm's performance but underperformed in reducing variance after multiple exploration phases.

Overall, this study contributes to understanding early convergence dynamics in AI pricing algorithms and provides valuable insights for policymakers and practitioners seeking to regulate and optimize algorithmic pricing mechanisms for consumer welfare. Further research could explore additional factors influencing convergence and extend the analysis to real-world market scenarios for practical implications.

References

Assad, S., Calvano, E., Calzolari, G., Clark, R., Denicolò, V., Ershov, D., Johnson, J., Pastorello, S., Rhodes, A., Xu, L., and Wildenbeest, M. (2021). Autonomous algorithmic collusion: economic research and policy implications. *Oxford review of economic policy*, 37(3):459–478.

Assad, S., Clark, R., Ershov, D., and Xu, L. (2020). Algorithmic pricing and competition: Empirical evidence from the german retail gasoline market. *CESifo Working Paper*, (8521).

Assad, S., Clark, R., Ershov, D., and Xu, L. (2022). Identifying algorithmic pricing technology adoption in retail gasoline markets. *AEA papers and proceedings*, 112:457–460.

Calvano, E., Calzolari, G., Denicolò, V., Harrington, J. E., and Pastorello, S. (2020). Protecting consumers from collusive prices due to ai. *Science*, 370(6520):1040–1042.

Calvano, E., Calzolari, G., Denicoló, V., and Pastorello, S. (2021). Algorithmic collusion with imperfect monitoring. *International Journal of Industrial Organization*, 79(C).

Chen, L., Mislove, A., and Wilson, C. (2016). An empirical analysis of algorithmic pricing on amazon marketplace. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 1339–1349, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Johnson, J. P., Rhodes, A., and Wildenbeest, M. (2023). Platform design when sellers use pricing algorithms. *Econometrica*, 91(5):1841–1879.

Klein, T. (2021). Autonomous algorithmic collusion: Q-learning under sequential pricing. *The Rand journal of economics*, 52(3):538–558.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092.

Robert, C. P. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer New York, NY, 2 edition.

Seele, P., Dierksmeier, C., Hofstetter, R., and Schultz, M. D. (2021). Mapping the ethicality of algorithmic pricing: A review of dynamic and personalized pricing. *Journal of business ethics*, 170(4):697–719.