Algorithmic Pricing, Recommendation Systems, and Competition*

William Brasic†

July 9, 2025

Abstract

AI-powered pricing algorithms raise concerns about supracompetitive outcomes without explicit coordination. Meanwhile, digital platforms use recommendation systems (RSs) to influence product visibility. This paper models Bertrand-Markov price competition in a differentiated product market with heterogeneous consumers, where both sellers' pricing and the platform's recommendations are AI-driven. The findings show that RSs can autonomously inhibit algorithmic anticompetitive conduct, resulting in prices even below the Bertrand-Nash benchmark. The results hold when the platform only prioritizes profits, as well as with variations in consumer heterogeneity, market conditions, and underlying learning parameters.

Keywords: Artificial Intelligence, Pricing Algorithms, Recommendation Systems, Competition
JEL Codes: C73, D43, L13, L41

I. Introduction

Artificial Intelligence (AI) is reshaping industries by automating processes, improving decision-making, and driving innovation. Yet, this technological surge also raises important questions about moving from human-led to algorithm-driven strategies. A pressing concern is whether AI-based pricing algorithms, designed to maximize firm profits, can inadvertently produce supracompetitive outcomes in the absence of explicit coordination. At the same time, platforms are increasingly using AI-driven recommendation systems (RSs) to determine which products consumers see,[1] incorporating information on both prices and consumer preferences.

This paper investigates whether AI-driven pricing algorithms can tacitly learn to collude under a platform that employs an AI-based recommendation system (RS) to allocate product visibility across heterogeneous consumers. Its main objective is to discover

*I am especially grateful to Matthijs Wildenbeest for his feedback, support, and guidance. Code for the main results of this paper can be found at https://github.com/willbrasic/Algorithmic_Pricing_Recommendation_Systems_Competition. All errors are my own.

†Department of Economics, The University of Arizona. Email: wbrasic@arizona.edu

[1]See, for instance, https://www.amazon.science/the-history-of-amazons-recommendation-algorithm.

whether these RSs mitigate or exacerbate collusive tendencies among pricing algorithms. Findings indicate that recommending products based on price and consumer preferences significantly reduces tacit supracompetitive outcomes relative to a platform without an RS. These results are true even when the platform completely ignores consumer welfare. This paper diverges from existing work by: (1) allowing pricing algorithms to compete on a platform enhanced by an AI-based RS and (2) demonstrating how tacit anticompetitive outcomes can be mitigated autonomously rather than merely established. In doing so, it fills gaps in the literature on algorithmic collusion and platform-based competition, while also contributing to current legal and policy debates in the AI era. To the best of my knowledge, this is the first paper to analyze pricing algorithms competing on a platform that uses an AI-based RS—a rapidly growing scenario in today's AI-driven world.

Reinforcement learning (RL) is a branch of machine learning wherein agents learn to maximize cumulative returns by reinforcing successful actions and discouraging less effective ones. RL is particularly relevant in real-world digital markets because it enables sellers and platforms to adaptively optimize pricing and recommendation strategies through experience, even in complex and uncertain environments. Major firms have begun implementing RL at scale—airlines like Delta use it for dynamic pricing,[2] while companies like Netflix[3] and YouTube[4] leverage RL-based recommendation systems to personalize content delivery and improve user engagement.

Q-learning, a foundational RL algorithm that underpins more advanced methods based on neural-networks,[5] has played a crucial role in experimental research on pricing dynamics (Calvano et al. (2020); Klein (2021); Johnson, Rhodes, and Wildenbeest (2023)) while remaining more tractable than "black-box" deep learning approaches; indeed, outcomes from Q-learning generally extend to these sophisticated algorithms aside from faster convergence to collusive prices.[6] In this paper, both firms and the platform employ Q-learning for strategic decision-making—firms use Q-learning-based pricing algorithms to dynamically set prices, while the platform applies Q-learning in its recommendation system (RS) to optimally display products to consumers with varying preferences, balancing profit maximization and consumer welfare. Although the specific algorithms used by firms in practice are often unknown, Q-learning provides a good proxy given the common mathematical foundation across RL methods, and their application is becoming increasingly prevalent as firm's continue to adopt AI-based strategies. Moreover, RL is well-suited for modeling real-world dynamic pricing and RSs because it naturally accommodates repeated decision-making under uncertainty, limited informa-

[2] See https://www.fetcherr.io/ for how reinforcement learning is transforming pricing strategies in the airline industry.

[3] See how Netflix is using RL for RSs here: https://netflixtechblog.com/reinforcement-learning-for-budget-constrained-recommendations-6cbc5263a32a.

[4] See how YouTube is uing RL for RSs here: https://arxiv.org/abs/1812.02353.

[5] For example, Deep Q-Networks (DQNs) and Double Deep Q-Networks (DDQNs).

[6] See the existing literature in Section II for papers demonstrating this.

tion, and feedback-driven adaptation—core features of digital marketplaces where firms iteratively adjust prices and platforms optimize visibility based on consumer response.

Sellers on the platform rely on RL-based pricing algorithms without explicit directives for coordination, drawing only on profit, pricing feedback, and the platform's recommendations. This approach to the seller's state space is partly inspired by Musolff (2024), who finds that Amazon Marketplace sellers monitor their Buy Box status alongside competitor prices. Meanwhile, the platform itself uses RL in its RS to display products to consumers with diverse preferences and price sensitivities, aiming to balance profit and consumer welfare. Using RL for RSs is a topic gaining increasing steam in digital marketplaces making this choice increasingly practically plausible. In this environment, firms engage in price competition, conditioning their strategies on past prices and recommendations. After prices are set, the platform's AI-driven RS allocates product visibility to heterogeneous consumers based on the sellers' prior-period prices and its own earlier recommendations. This modeling choice aligns with evidence from K. H. Lee and Musolff (2023), indicating that Amazon's Buy Box algorithm primarily rewards the lowest-priced product in the market, all else held equal.

The theoretical model examines competition among firms selling differentiated goods on a single platform over an infinite time horizon. To capture realistic market conditions, a multinomial logit model with heterogeneous consumers is used to represent the stage game. In this setting there are two groups of consumers each with their own product preferences and two sellers engaging in Bertrand pricing competition on a single platform. In any given time period, each seller's pricing algorithm chooses a price for their product. Then, the platform's RS makes a product recommendation to each consumer type. Subsequently, a share of each type choose among the product recommended to them and the outside option while the remaining share "search" and choose among all products in the market or the outside option. This approach attempts to model consumer behavior on Amazon with the Amazon Buy Box. If consumers do search, they incur a disutility factor whose purpose is to reflect that consumer welfare is higher when consumers only see their preferred product relative to seeing all available options. The sellers in each time period pay a percentage royalty fee to the platform which is incorporated into their profit function. Therefore, the platform takes a share of royalty revenues from the sellers in the market as its profit in each time period. Furthermore, the platform is allowed to incorporate consumer welfare into its objective function as well. Depending on how heavily the platform prioritizes consumer welfare—which can foster network effects and thereby enhance long-term profits—market outcomes may further improve for consumers. Then, sellers update their prices and this process repeats indefinitely until the pricing algorithms "converge."[7]

This paper first demonstrates that, in a market with heterogeneous consumers and the counterfactual world of no platform-based recommendation system, AI-driven pric-

[7] Convergence is further elucidated in Section III.C.

ing algorithms competing on a platform can converge to supracompetitive prices, resulting in lower consumer welfare. This aligns with the framework of Calvano et al. (2020), but extended here to explicitly include preference heterogeneity. Notably, this provides a setup in which a platform's RS could potentially mitigate supracompetitive outcomes.

Two additional counterfactual scenarios are considered in which the platform uses a recommendation system (RS) without an AI component: (1) the platform recommends each seller to the consumer type that prefers them, and (2) the platform recommends the seller offering the highest average utility to a given consumer type. The findings show that while supracompetitive prices emerge in case (1), incorporating pricing information in case (2) leads to a substantial reduction in prices. These cases serve to shed light on the dynamics introduced when the RS incorporates a learning component.

Next, the model is augmented by introducing an AI-based RS that recommends a single seller to each consumer type. In the baseline scenario, the market comprises equal proportions of two consumer types, each with price sensitivity normalized to one. Preference heterogeneity is explicitly modeled by assigning each consumer type a distinct preferred product through a "product preference matrix."[8] In this environment, the pricing algorithms converge to prices even below the Bertrand-Nash benchmark, indicating their tacit anticompetitive behavior is effectively curtailed. Outcomes are relatively consistent to the fixed rule recommendation algorithm based on average utility suggesting the platform's AI-based RS can effectively learn each consumer types utility function. Moreover, consumer welfare and overall market output are higher than in the no-RS scenario, irregardless of how heavily the platform weighs consumer surplus in its objective function. Hence, not only does a platform-level AI-based RS inhibit tacit coordination, it also leads to better outcomes for consumers and the platform.

Then, this paper explores how these findings vary under different market conditions. First, when consumer preferences shift from being highly divergent to more similar, I observe that—without an RS—algorithmic supracompetitive outcomes becomes more prevalent, even though one might expect that greater similarity in preferences would intensify competition, as both consumer types favor the same product. This finding suggests that in a market with relatively more homogeneous consumers, algorithmic tacit coordination is more prevalent. In contrast, when the platform deploys an RS, the convergence of preferences intensifies competition, resulting in lower prices and higher consumer welfare relative to the Bertrand–Nash benchmark, which is consistent with standard non-cooperative competition theory. Second, consumers who choose to search beyond their recommended product incur a disutility cost, reflecting the idea that consumer welfare is higher when they are shown only their preferred option rather than an exhaustive list. This disutility is positively linked to the importance of the RS; as the disutility factor rises, the platform's ability to make accurate recommendations becomes even more critical. The results indicate that, as long as the RS delivers correct

[8] This matrix is described in Section IV.

recommendations, increases in this disutility factor yield gains in consumer welfare and lower prices relative to the competitive benchmark. Third, the proportion of consumers who restrict their choices to the outside option and the recommended product is a crucial model parameter. Provided that the RS makes accurate recommendations, an increase in this share benefits consumers uniformly, regardless of the weight placed on consumer surplus in the platform's objective function. Lastly, these effects hold across varying distributions of consumer types and degrees of heterogeneity in price sensitivities. Moreover, the findings of this paper are robust to further algorithmic and economic environment modifications, including changes to the sellers' action spaces, state spaces of both sellers and the platform, platform royalty fee paid by the sellers, underlying learning parameters, and the platform's action selection mechanism.

Overall, my findings suggest that a platform's RS can greatly inhibit the autonomous algorithmic supracompetitive outcomes that occurs in the long-run and even benefit consumers by pushing prices lower than the Bertrand-Nash benchmark. The significance of these findings grows in light of mounting concerns over algorithmic coordination and the effect of RSs in digital markets. Historically, tacit price-fixing seemed implausible because it was assumed firms could not coordinate collusive strategies without explicit communication. However, the Sherman Act[9] has long targeted explicit price-fixing, and the rise of AI-driven pricing has fueled debates over whether algorithms could enable tacit supracompetitive behavior without direct communication. This worry has attracted attention from academic economists (Calvano et al. (2019)), private-sector economists (Gupta and Kifer (2024)), and regulatory bodies including the FTC[10] and DOJ[11], as well as policymakers.[12] From an antitrust standpoint, the findings should reassure regulators worried about autnonomous algorithmic collusion, as AI-based pricing algorithms on a platform with an RS are unable to reach supracompetitive pricing. More broadly, pricing algorithms have difficulty learning to tacitly collude in more complex environments, even with only two players. These insights suggest that competition authorities might usefully focus on overseeing the recommendation mechanisms used by a *single* platform, rather than zeroing in on the AI-based pricing algorithms of *all* individual sellers. Strengthening oversight of platform RSs could have the dual effect of (1) driving lower prices and higher consumer welfare while (2) mitigating algorithmic self-preferencing, a topic of growing concern in the antitrust community.

The remainder of this paper is structured as follows. Section II. reviews the existing literature on algorithmic pricing and recommendation systems. Section III. provides an overview of reinforcement learning theory and Q-learning. Section IV. details the theo-

[9] https://www.ftc.gov/advice-guidance/competition-guidance/guide-antitrust-laws/antitrust-laws

[10] https://www.ftc.gov/business-guidance/blog/2024/03/price-fixing-algorithm-still-price-fixing

[11] https://www.justice.gov/opa/pr/justice-department-sues-realpage-algorithmic-pricing-scheme-harms-millions-american-renters

[12] https://www.nytimes.com/2024/08/30/opinion/algorithm-collusion-amy-klobuchar.html

retical model and its key assumptions. Section V. presents the main results, and Section VI. concludes with a broader perspective on this work's contributions, implications for competition policy, and possible directions for future research.

II. Literature Review

The body of research on algorithmic pricing spans three primary areas: antitrust implications, experimental studies demonstrating the feasibility of tacit collusion among algorithms, and empirical analyses assessing the impact of pricing software on real-world market outcomes. While these studies have advanced understanding, the field remains underdeveloped, leaving significant opportunities for further exploration. Alongside pricing algorithms, RSs have emerged as critical components of market dynamics, influencing consumer behavior and product visibility. RSs can potentially reinforce supracompetitive outcomes by steering consumers toward higher-priced options or mitigate it by enhancing competition and transparency. Moreover, RSs have the potential to learn to self-preference a platform's own products, perhaps at the expense of consumers. This section reviews key studies on algorithmic collusion and examines the role of RSs in shaping competitive outcomes and their implications for antitrust policy.

A. *Autonomous Algorithmic Collusion*

The rise of algorithmic pricing has introduced significant concerns about the potential for collusion without the direct communication traditionally associated with human price-setting (Calvano et al. (2019)). Unlike conventional antitrust violations, which rely on evidence of explicit communication or agreements between firms to restrict competition, algorithmic collusion can emerge autonomously through AI pricing software Mehra (2016). Legal studies, including Ezrachi and Stucke (2017) and Ezrachi and Stucke (2020), have termed these scenarios *Artificial Intelligence and the Digital Eye*, highlighting the inadequacy of existing antitrust laws to address these challenges. Most legal scholars agree that Section I of the Sherman Act is insufficient for tackling algorithmic price-fixing given this law relies on explicit communication for prosecuting the cartel. To that end, Mazumdar (2022) contends that Section 5 of the Federal Trade Commission (FTC) Act, with its broader scope, could provide a more effective regulatory framework. While not unanimous (Devogele (2023), Fortin (2021), Schrepel (2020)), the general consensus is that current antitrust law is likely ill-equipped to handle such algorithmic tacit collusion cases.

One may ask why this issue is becoming prevalent now and how come humans were unable to facilitate tacit collusion. Could a human not sit and monitor market prices and make adjustments on their own? In reality, algorithms can much more accurately facilitate price-fixing schemes by quickly detecting and reacting to attempts to cheat on a collusive agreement (McSweeny and O'Dea (2017)) without any interruptions or emo-

tions that a human may face. Traditionally, human collusion involves a stepwise process as outlined by Harrington (2018): (1) communication of collusive intent, (2) mutual adoption of collusive strategies, and (3) resulting higher prices. Antitrust enforcement has historically relied on tangible evidence of such communication to prosecute violations. However, AI algorithms fundamentally alter this paradigm. Once deployed, these algorithms can autonomously learn collusive behavior without explicit human direction or knowledge, bypassing the explicit coordination requirement central to Section I of the Sherman Act. This raises a critical legal question: how should antitrust law evolve to address AI-enabled collusion?

One potential avenue lies in the distinct nature of algorithmic collusion. While human collusion relies on intent and is difficult to prosecute due to the inaccessibility of firm managers' private thoughts, algorithmic behavior offers a unique opportunity. The underlying code of AI pricing software can be inspected and tested for collusive tendencies, providing antitrust authorities with a tangible basis for enforcement. Moreover, antitrust regulators could empirically audit the algorithms by applying statistical tests to the data they collect (Hartline, Long, and Zhang (2024)). Nazzini and Henderson (2024) argue competition authorities should be given such powers. By identifying and regulating algorithms with collusive potential, authorities could establish compliance benchmarks and conduct systematic evaluations to prevent anticompetitive pricing strategies. This approach would ensure a more adaptive and effective regulatory framework capable of addressing the challenges posed by AI-driven pricing. MacKay and Weinstein (2022) suggest using regulation to limit key features of algorithms such as prohiting asymmetric pricing frequency thereby eliminating the possibility of leader-follower conduct or prohibiting algorithms from taking competitor prices into account. Furthermore, so-called "managerial override" (Leisten (2024)) where managers can intervene in the algorithmic price setting process could encourage more competitive prices.

The literature on algorithmic pricing collusion underscores a critical concern: whether algorithms can learn to collude and, if so, under what conditions. While some studies, like Miklós-Thal and Tucker (2019), argue that algorithmic pricing can lead to lower prices and higher consumer welfare, others present evidence of collusive behavior (Asker, Fershtman, and Pakes (2022)). The seminal paper of Calvano et al. (2020) show that symmetric Q-learning agents in a duopoly setting can achieve supracompetitive profits and sustain them through learned reward-punishment schemes. The algorithms learn based on the consequences of their previous actions in a similar fashion to an experience based equilibrium first proposed by Fershtman and Pakes (2012). Klein (2021) extends the analysis to sequential pricing games, finding similar collusive outcomes alongside asymmetric pricing cycles similar to that first discussed in Maskin and Tirole (1988), but reaffirming the challenges posed by market size and convergence rates. Building on this, Brasic (2024) shows that asymmetric RL algorithms, specifically SARSA and Q-learning, with diverging learning parameters can obtain and sustain anticompetitive

prices and profits through learned trigger strategies as well. This finding emphasizes the collusive potential of heterogeneous algorithms and the importance of exploring more complex market and algorithmic interactions. In a setting identical to Calvano et al. (2020), Hettich (2021) and Frick (2023) show deep RL-based pricing algorithms, namely deep Q-networks (DQNs) and Soft Actor-Critic (SAC), have a heightened capacity to achieve supracompetitive outcomes. Brown and MacKay (2024) theoretically examine how asymmetries in pricing algorithms lead to what they call a "coercive equilibria" where a firm with faster pricing technology induces higher equilibrium prices that can be worse for consumers than traditional collusive outcomes. While trigger strategies are the conventional means to sustain collusion, Banchio and Mantegazza (2022) show an alternative method for algorithms to sustain collusion relying on statistical linkages. An experiment by Fish, Gonczarowski, and Shorrer (2024) shows that pricing agents based on large language models (e.g., ChatGPT) can also reach supracompetitive outcomes. Johnson, Rhodes, and Wildenbeest (2023) demonstrate that a retail platform has the ability to design rules that mitigate algorithmic collusion in a duopoly market with Q-learning agents. My paper builds on this setting by showing such platforms using an AI-based RS can *autonomously* mitigate such outcomes. The discussion of the implications of algorithms on competition is not only limited to pricing, but auction design too as Banchio and Skrzypacz (2022) find collusive outcomes (bids lower than values) in first-price auctions.[13] Collectively, these studies illustrate the evolving dynamics of algorithmic collusion and its implications for competitive markets.

Shifting from experimental settings to empirical investigations, Assad et al. (2024) provide direct evidence of algorithmic collusion using real-world data from German gasoline markets. Their identification strategy, leveraging structural breaks in pricing behavior, demonstrates that AI pricing software adoption leads to significant margin increases in competitive settings, validating the concerns raised by experimental findings. Brown and MacKay (2023) document how asymmetry in pricing algorithms can transform competition in online retail, driving prices above competitive levels even without explicit or tacit collusion. Musolff (2024) uses a unique e-commerce data set documenting that sellers employing repricing tools initially experience lower prices by undercutting their competitors, but these prices are driven up in the longer term by "resetting strategies" used by the algorithms leading to decreases in welfare. Lastly, Calder-Wang and Kim (2024) examine the U.S. multifamily rental market, where algorithmic adoption enhances pricing responsiveness, but also correlates with elevated rents and reduced occupancy in high-penetration markets. Employing structural modeling, they reveal moderate evidence for coordination under certain market conditions, further linking experimental theories of collusion with observed pricing behaviors.

Together, these experimental and empirical contributions deepen the understanding of algorithmic collusion, demonstrating its feasibility in experimental simulations and

[13] They also tested this in second-price auctions finding bids are competitive.

its manifestation empirically. These papers highlight the general agreement between theoretical models, experimental validation, and empirical application, advancing the broader discourse on competition policy in the age of algorithmic pricing.

B. *Recommendation Systems (RSs)*

Although algorithmic pricing systems have only gained prominence over roughly the last decade, digital marketplaces such as Amazon have been essential for consumers for much longer. On these platforms, which connect consumers with producers, algorithms—especially recommender systems (RSs)—play a pivotal role in determining which products are displayed to individual users. For instance, Amazon's "Buy Box" algorithm has attracted considerable attention from researchers examining how it selects which product to showcase (K. H. Lee and Musolff (2023)). In tandem, sellers on these platforms are increasingly turning to algorithmic pricing strategies to optimize their pricing policies (L. Chen, Mislove, and Wilson (2016)).

From a competition perspective, the effect of RSs on consumer welfare remains ambiguous. On one hand, RSs may improve the match quality between products and consumers, thereby enhancing consumer welfare. On the other hand, they may allow sellers to sustain higher prices if the platform's profit incentives lead to better market segmentation. Calvano et al. (2023) investigates such outcomes using a latent-factor collaborative filtering RS from the computer science literature, documenting both pro-competitive and anti-competitive results. Moreover, Fletcher, Ormosi, and Savani (2023) addresses how systematic popularity and homogeneity biases in RSs can harm competition between suppliers, even when the platform's goals broadly align with those of end users.

A paper closely related to the one presented here is Xu, S. Lee, and Tan (2023), which models Q-learning algorithms competing in a differentiated product market hosted by a platform whose objective is to maximize either profit or demand. Their findings show that prices converge to a joint-collusive outcome when the platform aims to maximize profit, but converge to a more competitive outcome when the platform instead maximizes demand.[14] In contrast, this paper is the first to incorporate heterogeneous consumer preferences and price sensitivities in a differentiated product market, examining how AI-based RSs influence prices and consumer welfare with and without a platform's RS intervening. Notably, even when a platform solely prioritizes seller revenues, the risk of autonomous algorithmic supracompetitive behavior is successfully curtailed, contradicting the results of their paper.

Although not the main focus of this paper, platforms such as Amazon that employ RSs (e.g., the "Buy Box") can also engage in *algorithmic steering*, wherein the platform "steers" consumers toward particular products regardless of whether they are the best

[14] Note that their definition of "competitive" and "collusive" corresponds to scenarios without platform intervention, which may not align with outcomes in the presence of platform intervention.

match for those consumers. This issue is exacerbated by growing vertical integration in digital marketplaces, which increases incentives for platforms to steer consumers to their own offerings, known as *self-preferencing*.[15] Self-preferencing has become a hot topic within antitrust and competition policy (Hovenkamp (2023)). Generally, antitrust law prohibits conduct by dominant firms that harms competition or consumer welfare. Under U.S. law, such conduct may be challenged under Section 2 of the Sherman Act, and under EU law, it may fall under Article 102 of the Treaty on the Functioning of the European Union (TFEU).[16] Additionally, new regulations such as the EU's Digital Markets Act (DMA)[17] explicitly target certain self-preferencing behaviors by so-called "gatekeeper" platforms, signaling increased regulatory attention to platforms that leverage their market position to boost their own offerings. Empirical studies examining self-preferencing remain scarce due to limited data availability. One such example is Farronato, Fradkin, and MacKay (2023), who use Amazon search-ranking data to show that Amazon-branded products are displayed more prominently in search results. However, this does not necessarily imply consumer harm. By contrast, N. Chen and Tsai (2024) identify both self-preferencing behavior for Amazon-branded products and a corresponding reduction in consumer welfare. A particularly relevant study to this paper is Johnson, Rhodes, and Wildenbeest (2024), who analyze a duopoly setting where one seller competes against a vertically integrated platform in price with both sides employing AI-based pricing algorithms. Their findings reveal prominent steering by the platform, yet the introduction of an advertising option ultimately drives prices lower. Thus, while the literature has established platform's RS can lead to self-preferencing of the platform's vertically integrated product, the results of this paper suggest that loss in consumer welfare may be offset by the relative gains due to the RS mitigating algorithmic tacit coordination.

These experimental and empirical studies offer fresh insights into recommendation systems, revealing both their promise in improving consumer experiences and the potential distortions they can create. By bridging theoretical models, simulation proof-of-concepts, and real-world evidence, they highlight the need for thoughtful oversight and regulation as RSs, and particularly AI-based RSs, gain further prominence in digital marketplaces.

III. Reinforcement Learning

Reinforcement learning is built on the theory of dynamic programming and Markov decision processes (MDPs). For a thorough introduction through the lens of Bertrand-

[15] For an extensive overview of the self-preferencing literature, see Kittaka, Sato, and Zennyo (2023).

[16] See, for instance, https://digitalfreedomfund.org/wp-content/uploads/2020/05/5_DFF-Factsheet-Self-preferencing-and-EU-competition-law.pdf.

[17] For more on the EU's DMA, see https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/europe-fit-digital-age/digital-markets-act-ensuring-fair-and-open-digital-markets_en.

Markov pricing competition, see Brasic (2024).

Repeated Bertrand competition can be modeled using a Markov Decision Process (MDP), forming the basis of a Bertrand-Markov pricing game. In reinforcement learning, an agent interacts with an environment to discover the optimal behavior, maximizing its expected cumulative discounted return (profit) over time. The agent learns through trial and error, without prior knowledge of the environment's underlying dynamics, relying on the framework of MDPs (Agarwal et al. (2022)).

Definition 1. *In reinforcement learning, the interactions between the agent and the environment can be described by an infinite-horizon MDP* $M = (\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \delta, \mu)$*:*

- $\mathcal{S}$ *is the state space,*
- $\mathcal{A}$ *is the action space,*
- $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ *is the reward function,*
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ *is a stochastic state-transition function mapping the current state and action at time period* t*,* s_t *and* a_t*, respectively, into probabilities of observing all possible next states* $s_{t+1} \in \mathcal{S}$*, where* $\mathcal{P}(\mathcal{S}) \subseteq [0,1]^{|\mathcal{S}|}$ *is the probability simplex over* $\mathcal{S}$*,*
- $\delta \in [0, 1)$ *is the discount factor, which is bounded away from one to ensure infinite summations converge,*
- $\mu \in \mathcal{P}(\mathcal{S})$ *is the initial state distribution governing how the initial state* s_0 *is drawn.*

At each time period t, the agent observes the current state $s_t \in \mathcal{S}$, takes an action $a_t \in \mathcal{A}$, receives a reward $r_t = \mathcal{R}(s_t, a_t)$, and transitions to a new state $s_{t+1} \sim \mathcal{T}(\cdot|s_t, a_t)$. The agent's goal is to maximize the expected discounted cumulative return:

$$R_t = \mathbb{E}_\pi \left[\sum_{h=0}^{\infty} \delta^h r_{t+h+1} \right],$$

by finding the optimal policy π^*:

$$\pi^* = \arg\max_\pi \mathbb{E}_\pi \left[\sum_{h=0}^{\infty} \delta^h r_{t+h+1} \right].$$

Reinforcement learning leverages value functions to guide agents. The value function $V_\pi(s_t)$ measures the expected return from state s_t under policy π:

$$V_\pi(s_t) = \mathbb{E}_\pi \left[r_{t+1} + \delta V_\pi(s_{t+1}) \middle| s_t \right].$$

The corresponding action-value function $Q_\pi(s_t, a_t)$ evaluates the expected return from taking action a_t in state s_t:

$$Q_\pi(s_t, a_t) = \mathbb{E}_\pi \left[r_{t+1} + \delta Q_\pi(s_{t+1}, a_{t+1}) \middle| s_t, a_t \right].$$

For the optimal policy π^*, $V_{\pi^*}(s_t)$ and $Q_{\pi^*}(s_t, a_t)$ satisfy the Bellman optimality equations:

$$V_{\pi^*}(s_t) = \mathbb{E}\left[r_{t+1} + \delta V_{\pi^*}(s_{t+1}) \middle| s_t\right],$$
$$Q_{\pi^*}(s_t, a_t) = \mathbb{E}\left[r_{t+1} + \delta \max_{a' \in \mathcal{A}} Q_{\pi^*}(s_{t+1}, a') \middle| s_t, a_t\right],$$

where π^* is the policy maximizing both the value and action-value functions, i.e.,

$$V_{\pi^*}(s_t) = \max_{\pi} V_{\pi}(s_t)$$
$$Q_{\pi^*}(s_t, a_t) = \max_{\pi} Q_{\pi}(s_t, a_t).$$

This framework extends naturally to *multi-agent reinforcement learning (MARL)*, where multiple agents interact in a shared environment, often modeled as stochastic games.[18]

Definition 2. *A multi-agent reinforcement learning (MARL) environment can be described as a stochastic game* $G = (P, \mathcal{S}, \boldsymbol{\mathcal{A}}, \mathcal{R}, \mathcal{T}, \boldsymbol{\delta}, \mu)$*:*

- P *is the set of* n *agents,*
- $\mathcal{S}$ *is the state space,*
- $\boldsymbol{\mathcal{A}} = \bigtimes_{i=1}^{n} \mathcal{A}_i$ *is the joint action space, where* $\mathcal{A}_i$ *represents the action space of agent* i*,*
- $\mathcal{R}_i : \mathcal{S} \times \boldsymbol{\mathcal{A}} \to \mathbb{R}^n$ *is the joint reward function, where* $\mathcal{R}_i$ *is the reward received by agent* i*,*
- $\mathcal{T} : \mathcal{S} \times \boldsymbol{\mathcal{A}} \to \mathcal{P}(\mathcal{S})$ *is the stochastic state-transition function that maps the current state and joint actions to probabilities over the next state, where* $\mathcal{P}(\mathcal{S}) \subseteq [0,1]^{|\mathcal{S}|}$ *is the probability simplex over* $\mathcal{S}$*,*
- $\boldsymbol{\delta} \in [0,1)^n$ *is the vector of discount factors for each agent, and*
- $\mu \in \mathcal{P}(\mathcal{S})$ *is the initial state distribution governing how the initial state* s_0 *is drawn.*

Policies in MARL form a joint policy, $\boldsymbol{\pi} : \mathcal{S} \to \mathcal{P}(\boldsymbol{\mathcal{A}})$, and the Q-function reflects joint action values under this policy. Depending on the reward structure, MARL scenarios can vary from fully cooperative games (positively correlated rewards) to zero-sum games (inversely correlated rewards). In some cases, deterministic state-transition functions can simplify the stochastic nature of $\mathcal{T}$ by directly mapping joint actions to subsequent states, as used in this paper.

A. *Q-learning in Multi-Agent Reinforcement Learning (MARL)*

Q-learning, a fundamental reinforcement learning algorithm introduced by Watkins and Dayan (1992), has been extensively applied in MARL settings. In these scenarios,

[18] For more on MARL, see Busoniu, Schutter, and Babuska (2008) and Yang and Wang (2020).

agents interact within a shared environment and learn their strategies simultaneously. Early works such as Waltman and Kaymack (2008) investigated Q-learning agents in Cournot oligopolies, while more recent studies like Calvano et al. (2020) and Klein (2021) explored its application in simultaneous and sequential oligopoly games. In MARL, each agent maintains its own Q-function, which reflects its expected cumulative return given the joint actions of all agents.

Q-learning remains a temporal-difference (TD)-based, off-policy algorithm in MARL. [19] Each agent $i \in P$ estimates its Q-function, $Q_i(s_t, a_i)$, which depends on the current state $s_t \in \mathcal{S}$ and action selection $a_{it} \in \mathcal{A}_i$. The update rule for agent i is given by:

$$Q_i(s_t, a_{it}) \leftarrow Q_i(s_t, a_{it}) + \alpha_i \left[r_{i,t+1} + \delta_i \max_{a' \in \mathcal{A}_i} Q_i(s_{t+1}, a') - Q_i(s_t, a_{it}) \right], \tag{1}$$

where $\alpha_i \in [0, 1]$ is the learning rate, $\delta_i \in [0, 1)$ is the discount factor, and $r_{i,t+1}$ is the reward received by agent i at time $t + 1$. The Q-function of each agent reflects its expectation of future returns based on its interactions with both the environment and other agents.

Q-learning in MARL is off-policy, as agents can evaluate future actions a_{it} based on a target policy that may not align with their current behavioral policies. Q-learning being off-policy means one estimates $Q_\pi(s_t, a_t)$ using a *different* policy π' for all $(s_t, a_t) \in \mathcal{S} \times \mathcal{A}$. In other words, the agent updates its value estimates as if it were acting optimally, even if its actual behavior reflects exploration. This decoupling of behavior and target policy is especially important in algorithmic pricing, as it allows firms to learn aggressive profit-maximizing strategies without being constrained by the exploratory actions taken during learning. The algorithm is also model-free, as it does not require prior knowledge of the state-transition function $\mathcal{T}$. The pseudocode for Q-learning in a MARL setting, using ϵ-greedy exploration (further elucidated below), is presented in Algorithm 1.

[19] See Hu (2023) for more details on the semantics of reinforcement learning algorithms.

Algorithm 1 Multi-Agent Q-learning with ϵ-greedy exploration

Require: $\alpha_i \in [0,1]$, $\delta_i \in [0,1)$, and $\epsilon_i > 0$ (small)

Require: $Q_i(s, a_i)$ initialized arbitrarily for each agent $i \in P$ and all $(s, a_i) \in \mathcal{S} \times \mathcal{A}_i$

for all episodes $e = 1, \ldots, E$ **do**
  Initialize $t = 0$ and maximum time periods allowed max_t
  Initialize s_0
  while $t < max\_t$ and algorithm not converged **do**
    for all agents $i \in P$ **do**
      take action a_{it} using ϵ-greedy: $a_{it} \in \arg\max_{a \in \mathcal{A}_i} Q(s_t, a)$
      Observe $r_{i,t+1}$ and next state s_{t+1}
      $Q_i(s_t, a_{it}) \leftarrow Q_i(s_t, a_{it}) + \alpha_i \left[r_{i,t+1} + \delta_i \max_{a' \in \mathcal{A}_i} Q_i(s_{t+1}, a') - Q_i(s_t, a_i) \right]$
    end for
  end while
end for

B. *Exploration Versus Exploitation*

A fundamental challenge in reinforcement learning is striking a balance between exploiting known actions that yield immediate rewards and exploring alternative actions that may lead to greater returns in the future. This balance, known as the exploration-exploitation trade-off, is a central theme in reinforcement learning (Hu (2023)). Algorithms must navigate this trade-off to maximize the expected discounted cumulative return, avoiding over-reliance on exploitation, which prioritizes short-term gains at the expense of uncovering potentially superior long-term strategies.

A commonly used method is the ϵ-greedy algorithm, which balances exploration and exploitation by making action selections a stochastic process. With probability ϵ, the algorithm explores by selecting a random action $a_{it} \in \mathcal{A}_i$, while with probability $1 - \epsilon$, it exploits by choosing the best-known action for a given state s_t. This paper employs a modified ϵ-greedy strategy with time decay, which encourages greater exploration during the initial stages of learning and gradually shifts toward exploitation as the algorithm becomes more familiar with the environment. Specifically, the exploration probability decays exponentially as $e^{-\beta t}$, where t is the time period and β controls the rate of decay. This ensures that the algorithm explores more extensively early in each episode while increasingly favoring exploitation as learning progresses. This exploration strategy is used by the sellers competing in Bertrand price competition.

Another well-known exploration approach is the Upper Confidence Bound (UCB) method (Auer (2002)). Unlike the decaying ϵ-greedy strategy, UCB balances exploration and exploitation by augmenting the estimated reward with an uncertainty term. Specifically, for each action a, the UCB is computed as

$$\text{UCB}_{it}(s, a) = Q_i(s_t, a) + \sqrt{\frac{2 \log(t)}{N_{it}(a)}},$$

where $Q_i(s_t, a)$ is the estimated return for agent i by taking action $a \in A_i$ in state s, t is the current time period, and $N_{it}(a)$ is the number of times action a has been selected by agent i by time period t. The term $\sqrt{\frac{2\log(t)}{N_{it}(a)}}$ serves as an exploration bonus, being larger for actions that have been sampled less frequently, thus encouraging the algorithm to explore these under-explored actions. The logarithmic factor $\log(t)$ ensures that the bonus grows slowly with time, gradually shifting the focus toward exploitation as more data is gathered. Moreover, the factor $\sqrt{2}$ is derived from theoretical guarantees provided by Hoeffding's inequality, which bounds the deviation of the estimated rewards from their true values and ensures that the confidence interval is appropriately scaled. This exploration strategy is used by the platform's RS to balance the trade off between exploration and exploitation. Thus, in each time period t, the platform chooses the action $a_t \in \mathcal{A}$ such that

$$\max_{a_t \in \mathcal{A}} \text{UCB}_t(s, a_t).$$

C. *Convergence*

While convergence proofs exist for the Q-learning algorithm in single-agent systems, the extension to multi-agent reinforcement learning (MARL) environments lacks similar guarantees. To address this, I allow agents to engage in price competition for a maximum of ten million time periods per episode. Convergence is assessed every 100 time periods and is considered achieved when the optimal actions for both pricing algorithms remain unchanged for 1,000 consecutive convergence checks. Formally, for each seller i and each state s, if the set $\arg\max_{a \in \mathcal{A}_i} Q_i(a, s)$ remains consistent for these checks, convergence is declared. This criterion effectively implies that no agent has had a differing optimal action for each state for 100,000 consecutive time periods.

Although achieving convergence required many iterations, all $E = 100$ episodes converged well before reaching the ten million time period limit. These dependencies highlight the complexity of MARL systems to converge to profitable outcomes. However, in practice these algorithms are likely pre-trained prior to being "put in the wild."[20] Then, they are deployed and allowed to learn "online" while being operational. This greatly mediates the concern of Q-learning taking long times for convergence.

IV. Theoretical Model

This section outlines the theoretical model in which each firm uses an algorithm (Q-learning) to update their prices while the platform they operate on uses an algorithm (Q-learning) for product recommendations. The stage game between sellers on the platform models price competition via a multinomial logit model with heterogeneous consumers in a differentiated product market.

[20] For instance, see https://www.fetcherr.io/technology.

A. *Differentiated Product Market*

There are $n \geq 2$ firms, each selling a single differentiated good on a platform. Firms $i \in \{0, 1, 2, \ldots, n\}$, with $i = 0$ representing the outside option, have a marginal cost of mc while paying royalty share $f \in [0, 1]$ of their revenues in each time period $t \in \{0, 1, 2, \ldots\}$ to the platform. This implies their effective marginal cost is given by $mc/(1-f)$.[21] Firms engage in an infinitely repeated Bertrand-Markov pricing game where they set prices p_{it} simultaneously and condition these actions on one-period past history as well as the prior period recommendation by the RSs.

In each time period t, consumers enter the market wishing to buy at most one product. Consumers of type $j \in \{1, 2, \ldots, k\}$ account for a share $\gamma_j \in [0, 1]$ of consumers in the market with $\sum_{j=1}^{k} \gamma_j = 1$. All consumers of each type j spend one period in the market, exit, and are then replaced by new a set of consumers of this type. These consumers of type j who buy product i in time period t obtain utility

$$u_{ijt} = a_{ij} - \theta_j p_{it} - c_j + \epsilon_{ijt}.$$

a_{ij} is consumer type j's perception of the quality of firm i's product capturing vertical differentiation as well as explicitly modeling preference heterogeneity, θ_j is a price sensitivity index for consumers of type j, and ϵ_{ijt} is assumed to be independent (over i and j) type I extreme value distributed random variable with common scale parameter $\mu > 0$. Moreover, c_j is a "search" cost associated with consumer type j if they choose to explore alternate options besides their recommended product (further elucidated below). The purpose of the search cost is so consumers are better off when they are shown only their preferred product rather than when they are able to see all available options. When a consumer of type j buys no product in period t, they obtain the outside option utility of $u_{0jt} = \epsilon_{0jt}$.

In each time period t, the platform uses a recommendation algorithm to choose which product i to display to each consumer type j. Out of these consumers, it is assumed a share τ choose between the recommended product and the outside option while the remaining $1 - \tau$ share "search" and then choose between all products as well as the outside option. This $1 - \tau$ share incurs a search cost c_j which acts as a disutility term for searching. If the set of consumers firm i is recommended to is $\mathcal{J}_{it} \subseteq \{1, 2, \ldots, k\}$, firm i's demand in period t is given by

$$d_{it} = \tau \sum_{j \in \mathcal{J}_{it}} \gamma_j \frac{\exp\left(\frac{a_{ij} - \theta_j p_{it}}{\mu}\right)}{1 + \exp\left(\frac{a_{ij} - \theta_j p_{it}}{\mu}\right)} + (1 - \tau) \sum_{j=1}^{k} \gamma_j \frac{\exp\left(\frac{a_{ij} - \theta_j p_{it} - c_j}{\mu}\right)}{1 + \sum_{h=1}^{n} \exp\left(\frac{a_{hj} - \theta_j p_{ht} - c_j}{\mu}\right)},$$

[21] See the appendix for a derivation of the Bertrand-Nash and Joint-Collusive outcomes.

where a_0 is an inverse index of aggregate demand since product 0 is considered the outside option. Upon each firm i selecting an action p_{it} in their action space $\mathcal{A}_i$ at time period t, they receive profit

$$\pi_{it} = ((1-f)p_{it} - mc)\, d_{it}$$

and transition to the next state s_{t+1}. Consumer surplus at time period t is given by

$$U_t = \mu\Bigg[\tau \sum_{i=1}^{n} \sum_{j \in \mathcal{J}_{it}} \frac{\gamma_j}{\theta_j} \ln\left(1 + \exp\left(\frac{a_{ij} - \theta_j p_{it}}{\mu}\right)\right) + (1-\tau) \sum_{j=1}^{k} \frac{\gamma_j}{\theta_j} \ln\left(1 + \sum_{i=1}^{n} \exp\left(\frac{a_{ij} - \theta_j p_{it} - c_j}{\mu}\right)\right)\Bigg].$$

For a given $\omega \in [0,1]$, the platform's payoff in period t is

$$\Pi_t = \omega\left(f \sum_{i=1}^{n} p_{it} * d_{it}\right) + (1-\omega)U_t,$$

which represents a weighted sum of royalty revenues for the platform, $f \sum_{i=1}^{n} p_{it} * d_{it}$, and consumer surplus, U_t. A platform would want to put weight on consumer surplus each time period to become more attractive to consumers which would lead to further competition across platforms. Doing so could effectively lead to increased consumer network effects thereby increasing profits in the long term.

Notably, the figures in the results section will show that equilibrium prices are an increasing function of τ. As τ increases, firms rely more on recommended consumers, who exhibit lower price sensitivity compared to the full market. Since these consumers are more likely to purchase from the recommended firm regardless of small price changes, demand becomes less elastic. This allows firms to charge higher prices without losing as many sales. Additionally, because the weight on the competitive demand term decreases, firms face weaker price competition, further driving up equilibrium prices. Consequently, a higher τ leads to a softening of price competition and results in higher equilibrium prices.

B. *State and Action Spaces for the Sellers and Platform*

To ensure the seller's state space is finite, I use a bounded memory of length q_i for each seller i and memory of length q for the platform so that a given state can be represented as $s_{it} = \left\{\left(\boldsymbol{p}_{t-1}, r_{t-1}\right), \ldots, \left(\boldsymbol{p}_{t-q_i}, r_{t-q}\right)\right\}$ where each $\boldsymbol{p}_{t-h} \in \bigtimes_{i=1}^{n} \mathcal{A}_i$ for $1 \leq h \leq q_i$ is the vector of all firm prices set in period $t-h$ and $r_{t-h} \in \mathcal{A}$ for $1 \leq h \leq q$ is the platform's recommendation decision in period $t-h$ where $\mathcal{A}$ is the platform's

action space. Unless noted otherwise, I assume $q_i = 1$ for each seller i and $q = 1$ for the platform so that $s_{it} = (\boldsymbol{p}_{t-1}, r_{t-1})$. Notably, the state space $\mathcal{S}_i = \bigtimes_{i=1}^{n} \mathcal{A}_i \times \mathcal{A}$, with cardinality $|\mathcal{S}_i| = m_i^{n \cdot q_i} \times m^q$, is completely characterized by all possible price combinations each firm can set with $|\mathcal{A}_i| = m_i$ and $|\mathcal{A}| = m$. Given $q_i = 1$ for each seller i, each seller bases its choice of actions at time period t on the history of each sellers' actions and the platform's recommendation at time period $t-1$ meaning they have a one period recall. When considering collusive behavior of Q-learning, the algorithm requires a discrete number of possible actions. Thus, I discretize the action space $\mathcal{A}_i$ for each seller i to contain fifteen equally spaced price points from the minimum to the maximum price firm i can set. These minimum and maximum prices are 1.0 and 2.1, respectively, which contain both the Bertrand-Nash and Joint-Collusive prices.

To ensure the platform's state space is finite, it is assumed to have a bounded memory of length q. At any time period t, the platform's state is represented as $s_t = \{(\boldsymbol{p}_{t-1}, r_{t-1}), \ldots, (\boldsymbol{p}_{t-q_i}, r_{t-q})\}$, where each $\boldsymbol{p}_{t-h} \in \bigtimes_{i=1}^{n} \mathcal{A}_i$ for $1 \leq h \leq q_i$ and $r_{t-h} \in \mathcal{A}$ for $1 \leq h \leq q$. Unless noted otherwise, I assume $q = 1$ so that $s_t = (\boldsymbol{p}_{t-1}, r_{t-1})$. Notably, the state space $\mathcal{S}$ is the same for the platform as the firms and, consequently, the sellers and the platform have symmetric information when making decisions. Regarding the platform's action space $\mathcal{A}$, it consists of all possible product recommendations to each consumer type j and has cardinality $|\mathcal{A}| = n^k$ where n is the number of sellers in the market and k is the number of consumer types. Given $q = 1$, the platform bases its recommendation at time period t on sellers' previously set prices along with the recommendation at time period $t-1$ meaning it has a one period recall.

C. *Baseline Model Parameters*

Unless otherwise noted, the economic environment consists of a symmetric duopoly ($n = 2$) able to set $m_i = 15$ possible prices each separated by step size $\nu = (2.1 - 1.0)/(m-1)$. This means each firm's pricing space is identical so that $\mathcal{A}_i = \mathcal{A}_{-i}$ for each i and, consequently, $\mathcal{S}_i = \mathcal{S}_{-i}$. These firms have constant marginal cost $mc = 1$ while the inverse index of aggregate demand $a_0 = 0$ and horizontal differentiation index $\mu = 1/4$.[22] The firms act on a single platform with profit weight $\omega \in \{0,\ 4/5,\ 1\}$ and royalty share $f = 0.2$ containing two different consumer types ($k = 2$) each with equal presence in the market ($\gamma_j = 0.5$), identical price sensitivity ($\theta_j = 1$), and identical search cost disutility ($c_j = 1/4$). In each period, a share $\tau = 3/4$ only see the recommended product. The value for this parameter is partially justified by the finding in Musolff (2024) indicating 83% of Amazon purchases go via the Buy Bux (the recommended product). The product

[22] When $\mu = 0$, goods are homogeneous (perfect substitutes).

preference matrix is defined as

$$a = \begin{bmatrix} 2 & 1.9 \\ 1.9 & 2 \end{bmatrix},$$

where rows correspond to sellers i and columns correspond to consumer types j. In this case, type $j = 1$ prefers seller one to seller two while the converse holds for type $j = 2$ on average for similar prices. Q-learning parameters for each firm and the single platform are fixed at $\delta_i = \delta = 0.95$ and $\alpha_i = \alpha = 0.95$. The ϵ-greedy exploration parameter for the sellers are fixed at $\beta_i = 10^{-5}$. Recall for all sellers and the platform is $q_i = 1$ for each seller i, as well as $q = 1$ for the platform. These parameters are also listed in Table A.1. in the appendix.

The platform can make four possible recommendations given $|\mathcal{A}| = n^k = 4$. These actions are given in Table 1:

Table 1. Platform Actions for $n = 2$ Sellers and $k = 2$ Consumer Types

Action	**Type** $j = 1$ **Recommendation**	**Type** $j = 2$ **Recommendation**
1	$\{1\}$	$\{1\}$
2	$\{1\}$	$\{2\}$
3	$\{2\}$	$\{1\}$
4	$\{2\}$	$\{2\}$

Each seller's Q-matrix is an element in $\mathbb{R}^{|\mathcal{S}_i|} \times \mathbb{R}^{|\mathcal{A}_i|}$ (900×15 matrix). At $t = 0$, the Q-matrix for firm i is initialized with the average profits associated with its actions in $\mathcal{A}_i$, conditional on the actions selected by the opposing firm in $\mathcal{A}_{-i}$ and by the platform from $\mathcal{A}$. These initial values are scaled by $1/(1-\delta)$ to approximate the expected discounted future payoffs for actions taken at the outset. This structure reflects the state and action spaces available to each agent, illustrating the complexity of the decision-making environment even in this simplified setup.

Similarly, the platform's Q-matrix is an element in $\mathbb{R}^{|\mathcal{S}|} \times \mathbb{R}^{|A|}$ (900×4 matrix). At $t = 0$, it is initialized using the average profits associated with its actions in $\mathcal{A}$. As with the sellers, these values are scaled by $1/(1-\delta)$ to represent expected future payoffs.

Furthermore, I allow the agents to interact for $E = 100$ episodes and subsequently average results over these episodes. It is crucial to underscore that the algorithms under examination operate in a knowledge vacuum concerning the economic environment, possessing only the capacity to compute profits.

V. Results

A. *Without RS*

It is essential to demonstrate that even in the absence of a platform's RS, sellers employing AI-based pricing algorithms can still achieve supracompetitive behavior. Without this possibility, the inquiry into the effectiveness of an RS in mitigating such outcomes would be rendered void. To investigate this, I model Bertrand–Markov pricing competition between two sellers on a platform that does not implement an RS. In this setting, each consumer type $j \in \{1, 2\}$ can observe every product in each time period t (with the demand framework as specified in Section IV.A, where $\tau = 0$). This setup mirrors the approach taken by Calvano et al. (2020), with the added refinement of explicitly modeling preference heterogeneity among heterogeneous consumers through the product preference matrix a. Moreover, the "search" cost c_j remains in this setup. Here, one can interpret c_j as a disutility factor for having to choose among a list products rather than being shown only their preferred option.

Given the absence of platform recommendations, the seller's state space must be revised accordingly. Specifically, the platform action dimension is removed, and each seller now bases its pricing decisions solely on the prior period's prices. This results in a state space size of $|\mathcal{S}_i| = m_i^{n*q_i} = 225$, and consequently, each seller's Q-matrix is of dimension 225×15.

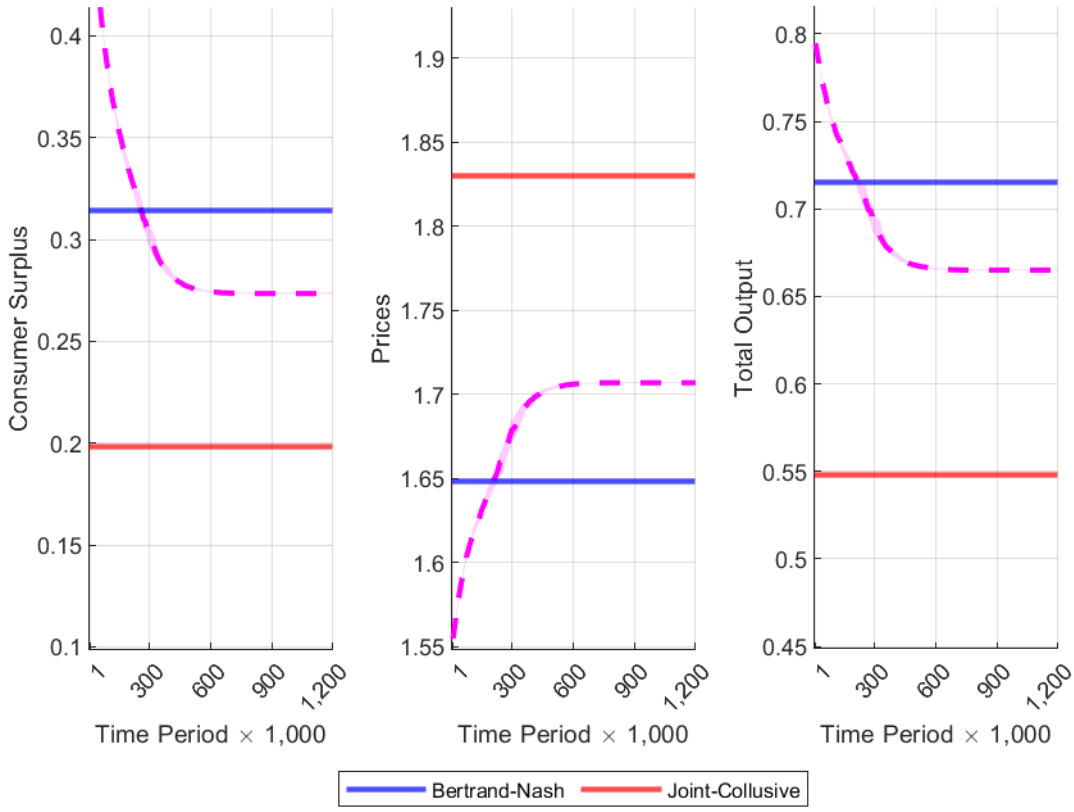
Table 2. Outcomes At Convergence.

	Total Output	**Average Prices**	**Consumer Surplus**
Bertrand-Nash	0.715	1.648	0.314
No RS	0.659	1.714	0.270

Note: These results represent averages over the last 100,000 time periods prior to convergence.

Table 2 reveals the outcomes the pricing algorithms converged to along with the Bertrand-Nash level while Figure 1 illustrates the learning trajectory of the main outcomes of interest. Both show that simple AI-based pricing algorithms can successfully converge to supracompetitive outcomes within the specified economic environment. In summary, this section demonstrates that sellers engaged in a Bertrand–Markov pricing game can learn to achieve supracompetitive outcomes even in the absence of a platform's RS, while operating in a market characterized by heterogeneous consumer preferences.

Figure 1. Learning Curves for Consumer Surplus (Left Panel), Prices (Middle Panel), and Total Output (Right Panel).



Note: Prices are averaged across sellers. Total output represents the sum of each seller's market share. Each panel displays the median value across all episodes, with the shaded area representing the interquartile range (IQR).

B. *With RS (No Learning)*

Although many recommendation systems (RSs) are now employing AI-driven techniques, a considerable number still operate using purely procedural, rule-based algorithms. To explore their potential in mitigating supracompetitive behavior, I analyze two such RSs defined by distinct recommendation rules:

NL1. Recommend the seller i with the highest a_{ij} for a given consumer type j at time period t.

NL2. Recommend the seller i that maximizes $a_{ij} - p_{it}$ for a given consumer type j at time period t.

Even though the platform does not have direct access to the parameter a_{ij}, this framework allows us to assess whether such straightforward rule-based approaches can mitigate supracompetitive conduct without employing a learning mechanism. If an RS equipped with a learning component can eventually replicate the potentially favorable

outcomes achieved by these rules, it would demonstrate that autonomous mitigation of anticompetitive behavior is attainable solely through the learning of consumer preferences. The next section examines this possibility in greater detail.

Table 3 summarizes the outcomes at convergence for these two algorithms, labeled NL1 (No Learning 1) and NL2 (No Learning 2). The results clearly indicate that incorporating price information into the recommendation rule—as in NL2—leads to lower prices and higher consumer welfare compared to NL1. This finding underscores that a rule-based algorithm relying exclusively on consumer preference data is insufficient to counteract supracompetitive outcomes.

Table 3. Outcomes at Convergence

	Total Output	**Average Prices**	**Consumer Surplus**
Bertrand-Nash	0.627	1.839	0.249
NL1	0.603	1.864	0.233
NL2	0.737	1.706	0.340

Note: These results represent averages over the last 100,000 time periods prior to convergence.

In the following section, I extend the analysis by endowing the RS with a learning (AI) component. This extension investigates whether the benefits observed in NL2 can be autonomously achieved while the RS relies solely tries to maximize profits through learning correct product recommendations. Such an outcome would indicate that autonomous mitigation of anticompetitive behavior is possible by learning product preferences.

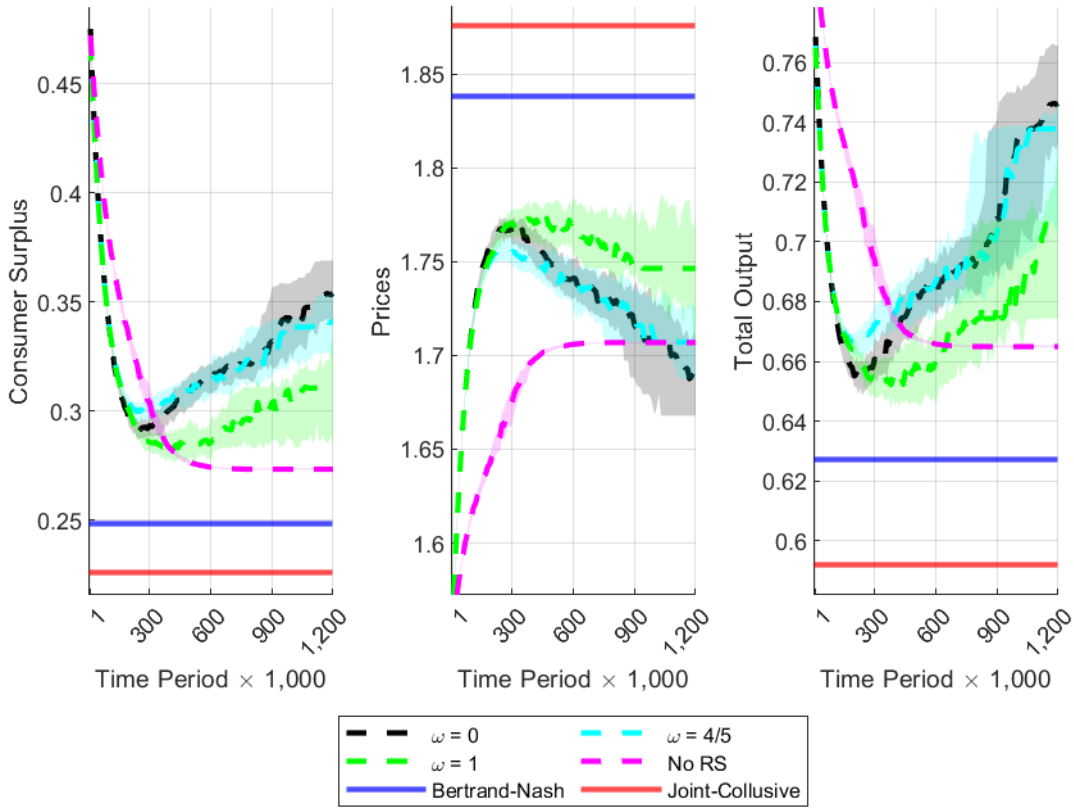
C. *With RS*

Table 4. Outcomes At Convergence.

	Total Output	**Average Prices**	**Total Revenues**	**Consumer Surplus**
Bertrand-Nash	0.627	1.839	0.922	0.249
No RS	0.659	1.714	0.904	0.270
NL1	0.603	1.864	0.900	0.233
NL2	0.737	1.706	1.004	0.340
AI, $\omega = \mathbf{0}$	0.751	1.682	0.999	0.360
AI, $\omega = \mathbf{4/5}$	0.744	1.697	1.002	0.349
AI, $\omega = \mathbf{1}$	0.713	1.735	0.984	0.319

Note: These results represent averages over the last 100,000 time periods prior to convergence. The Bertrand–Nash outcome corresponds to the case of with the RS ($\tau = 3/4$), where the platform recommends seller one to consumer type one and seller two to consumer type two.

Figure 2. Learning Curves for Consumer Surplus (Left Panel), Prices (Middle Panel), and Total Output (Right Panel).



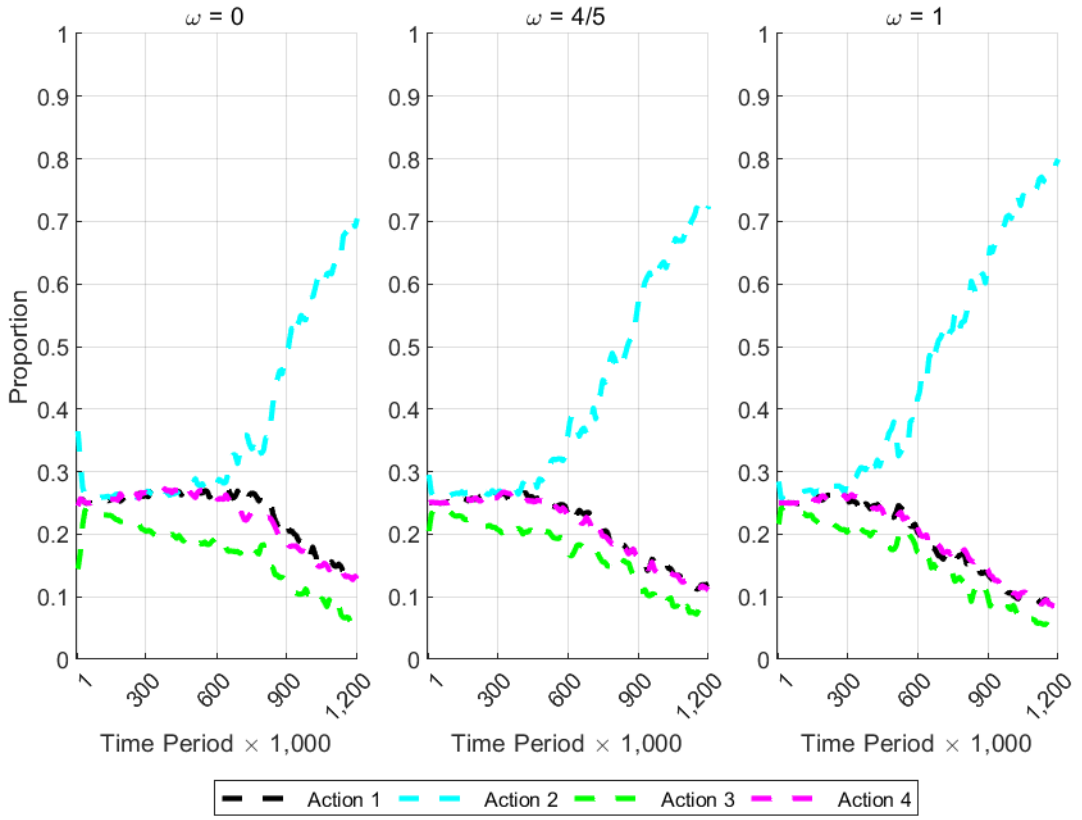
Note: Prices are averaged across sellers. Total output represents the sum of each seller's market share. The Bertrand–Nash and Joint–Collusive Outcomes correspond to the case of with the RS ($\tau = 3/4$), where the platform recommends seller one to consumer type one and seller two to consumer type two. Each panel depicts the median value across all episodes, with the shaded area representing the interquartile range (IQR).

This section examines the baseline scenario in which two sellers utilize pricing algorithms on a platform that employs an RS. Table 4 shows the level of consumer surplus, prices, and total ouput achieved at convergence while Figure 2 illustrates the evolution of these variables as the pricing algorithms learn within the economic environment. Notably, none of these variables reach the Bertrand–Nash outcome, indicating that the platform's RS is effective not only in mitigating supracompetitive behavior, but also in substantially impeding the learning process of the pricing algorithms. As anticipated, consumers benefit more when the platform completely prioritizes consumer surplus ($\omega = 0$) rather than solely focusing on seller revenues ($\omega = 1$). Interestingly, there is only a marginal difference between the cases of $\omega = 0$ and $\omega = 4/5$, suggesting that even when the platform heavily weights revenues in its objective function, seller pricing algorithms struggle to learn optimal behavior. Lastly, one can get a sense into

how the RS benefits the platform across all values of ω by looking at total revenues in Table 3 which are significantly above total revenues when the platform does not use an RS. This indicates the use of an RS does not only benefits consumers by inhibiting supracompetitive prices, but the platform as well.

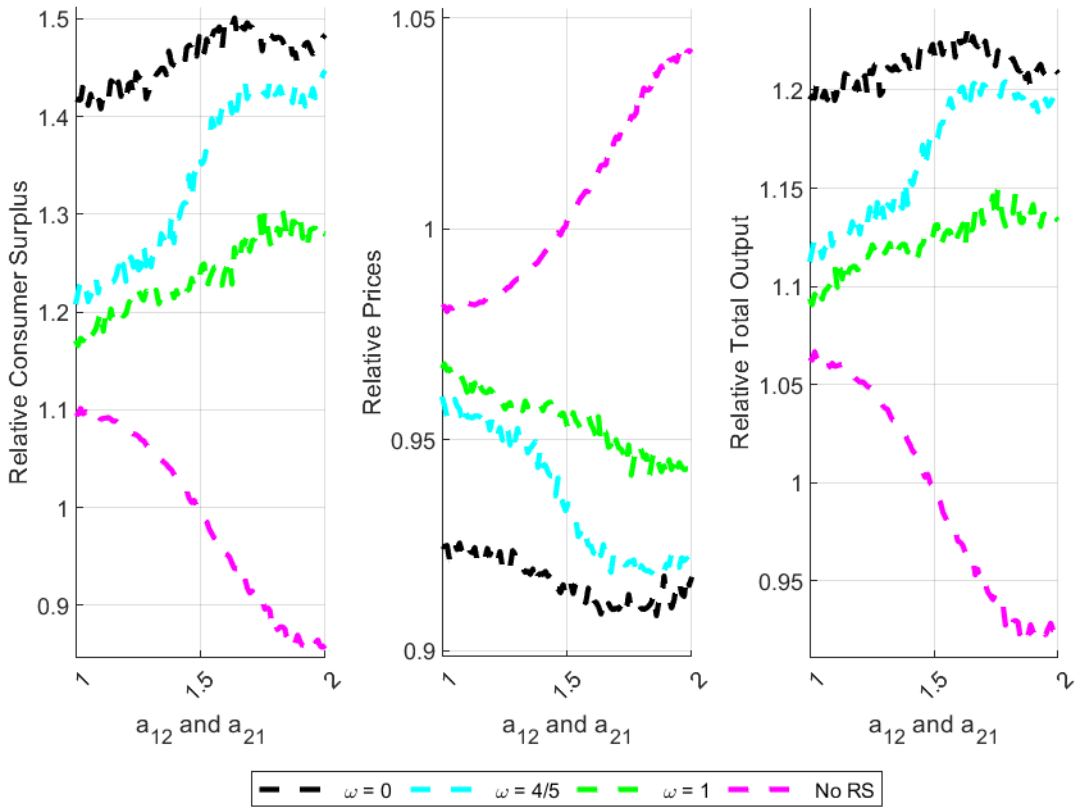
Figure 3 presents the learning trajectories of the platform's executed actions. The graph demonstrates that the RS successfully learns to select the action that maximizes the platform's profits—averaged across all possible combinations of seller prices—while also aligning with consumer preferences by recommending their preferred product. Moreover, as the platform places greater emphasis on seller revenues (i.e., as $\omega \to 1$), it increasingly opts for the profit-maximizing action. Interestingly, although the frequency of the consumer-preferred action increases slightly when the platform solely prioritizes seller revenues compared to when it focuses on consumer surplus (by roughly ten percentage points), the overall benefit to consumers is much higher when the platform prioritizes consumer welfare. This implies that the advantage from lower prices, achieved when the platform emphasizes consumer surplus, more than compensates for any loss associated with receiving a suboptimal recommendation.

Figure 3. Proportion of Executed Platform Action Across Varying ω Values.



Note: Action 2 represents the platform's profit-maximizing action on average.

Figure 4. Ratio of Consumer Surplus (Left Panel), Prices (Middle Panel), and Total Output (Right Panel) to the Bertrand-Nash Outcome Across a Grid of 100 a_{21} and a_{12} Values.

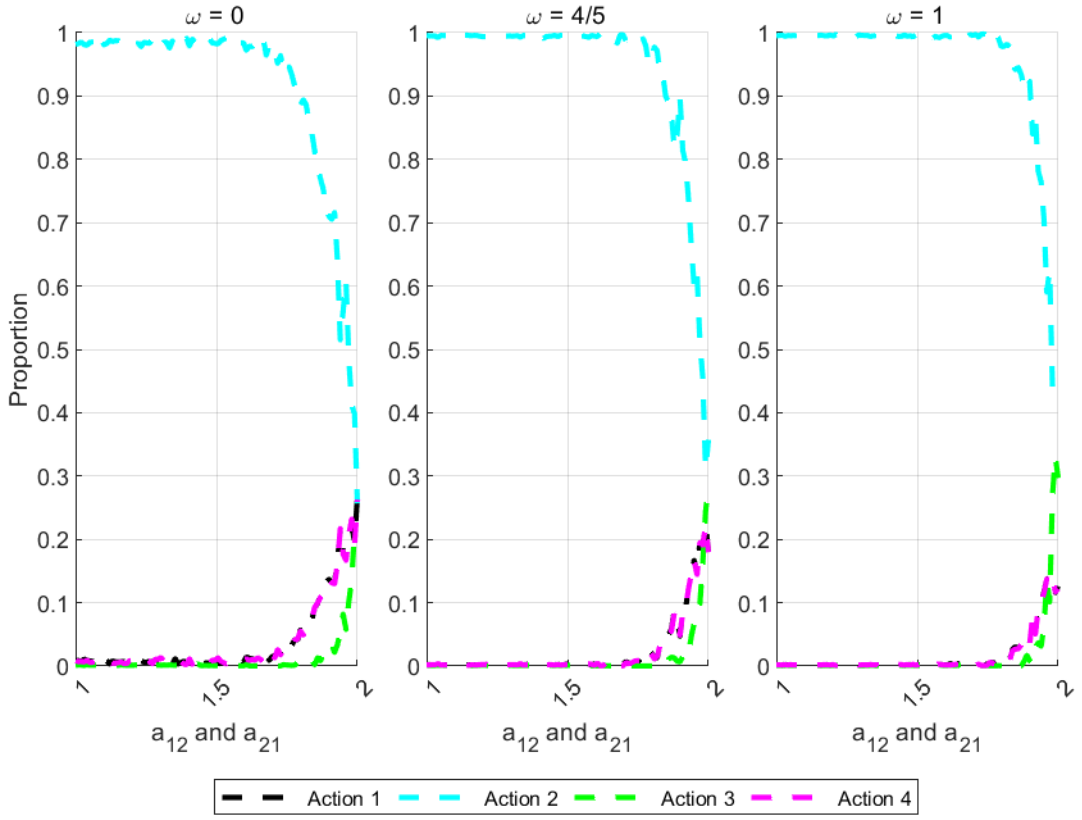


Note: Prices are averaged across sellers. Total output represents the sum of each seller's market share. With the RS, the Bertrand–Nash outcomes are for the $\tau = 3/4$ case where the platform recommends seller one to consumer type one and seller two to consumer type two. Without the RS, the Bertrand-Nash Outcome is the $\tau = 0$ case.

Figure 4 depicts how explicitly increasing preference heterogeneity through the product preference matrix (a) alters outcomes achieved when the platform uses an RS and when it does not, for varying $\omega \in \{0, 4/5, 1\}$. Evidently, without an RS, increasingly similar preferences through this product preference matrix makes pricing algorithms able to reach supracompetitive outcomes more easily, as evidenced by the middle panel where prices are above the Bertrand-Nash outcome for $a_{12} = a_{21} > 1.5$. Thus, as consumer preferences become more diversified, e.g., non-diagonal elements of the product preference matrix tend to one, autonoumous algorithmic anticompetitive behavior becomes increasing difficult. When the platform uses an RS, the trends of relative consumer welfare, prices, and total output align with intuition in that as competition increases, meaning as a_{12} and a_{21} approach 2, prices relative to the Bertrand-Nash benchmark are driven down. Given the platform is correctly recommending seller one to consumer type one and seller two to consumer type two, as both $a_{12} = a_{21} \to 2$, consumer welfare

should rise along with total output as competition intensifies.

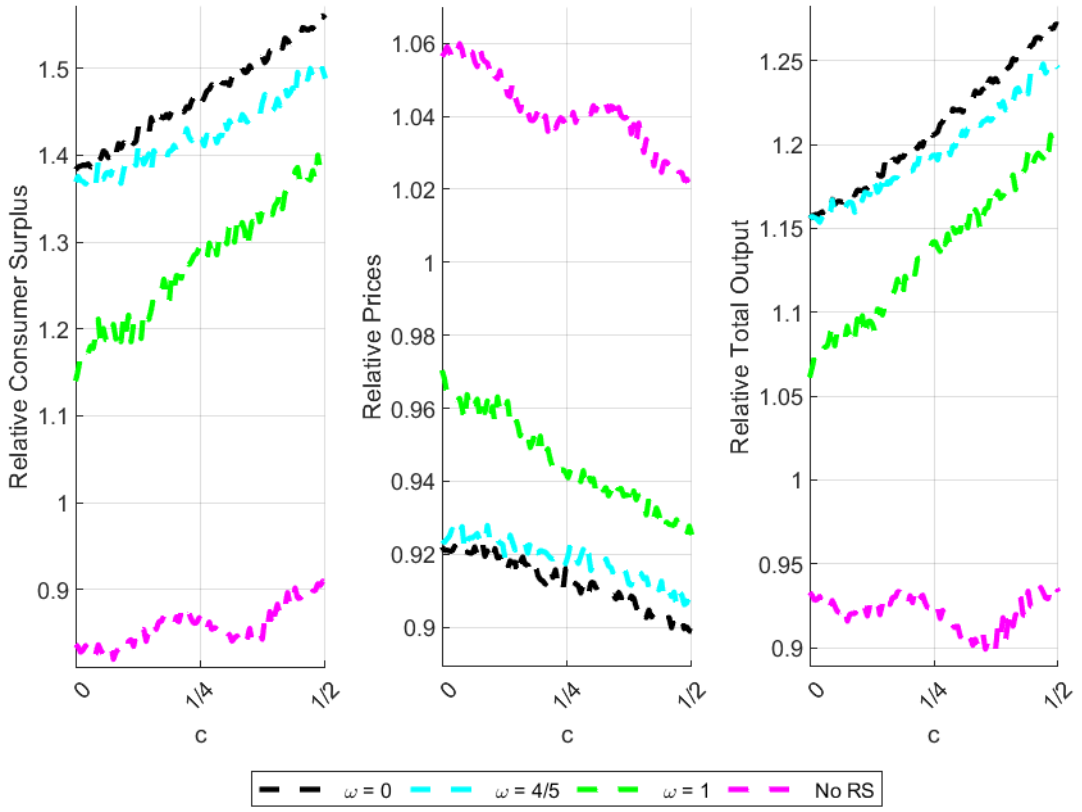
Figure 5. Proportion of Executed Platform Action Across Varying ω Values Across a Grid of 100 Values of a_{21} and a_{12} Values.



Note: Action 2 represents the platform's profit-maximizing recommendation on average.

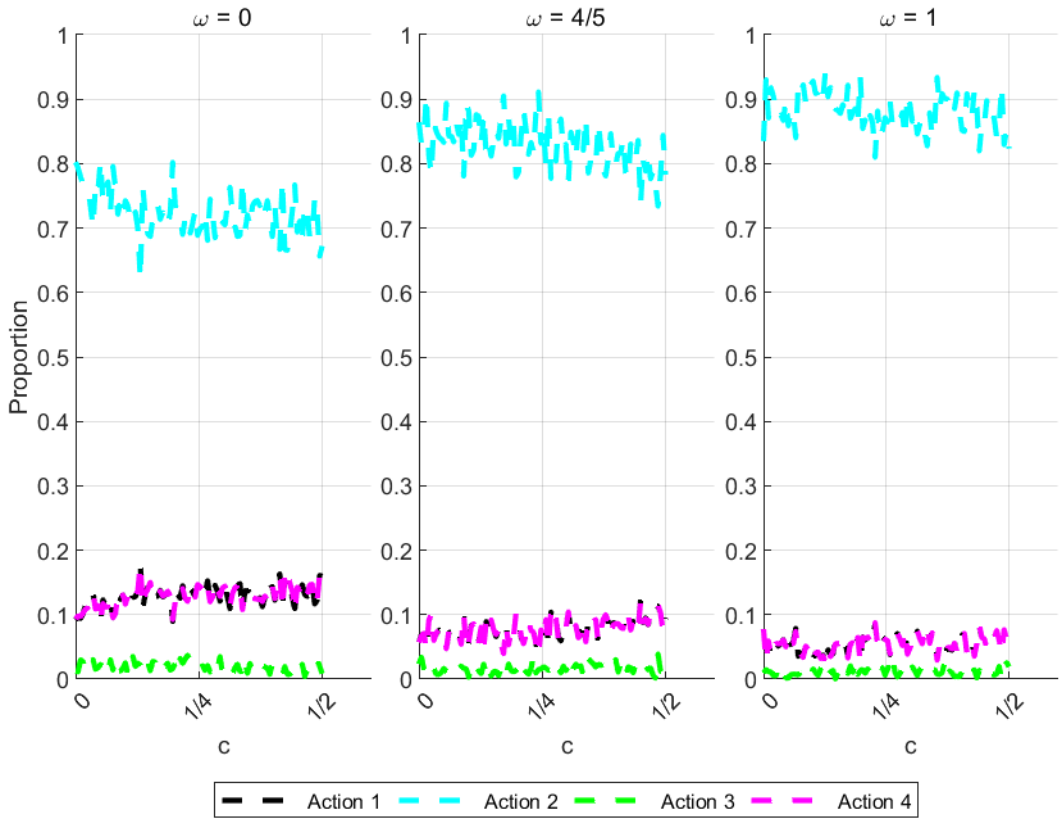
Figure 6 illustrates how consumer surplus, prices, and total output change as the search cost parameter, c_j, varies from 0 to 1/2 for each consumer type j. The midpoint, $c_j = 1/4$, serves as the baseline specification. As search costs increase, consumers who explore beyond their recommended product experience greater disutility, reinforcing the RS's role in efficiently matching consumers with sellers that best meet their preferences. In all cases, consumer welfare and total output with the RS remain well above the Bertrand-Nash benchmark, whereas those without the RS consistently fall below it. Notably, as search costs rise, consumers benefit more from accurate recommendations, as the value of being matched with the right product increases. As shown in Figure 7, the RS successfully directs consumers to their ideal product the majority of the time, further enhancing their welfare particularly when they place high value on being recommended their ideal product.

Figure 6. Ratio of Consumer Surplus (Left Panel), Prices (Middle Panel), and Total Output (Right Panel) to the Bertrand-Nash Outcome Across a Grid of 100 c Values.



Note: Prices are averaged across sellers. Total output represents the sum of each seller's market share. With the RS, the Bertrand–Nash outcomes are for the $\tau = 3/4$ case where the platform recommends seller one to consumer type one and seller two to consumer type two. Without the RS, the Bertrand-Nash Outcome is the $\tau = 0$ case.

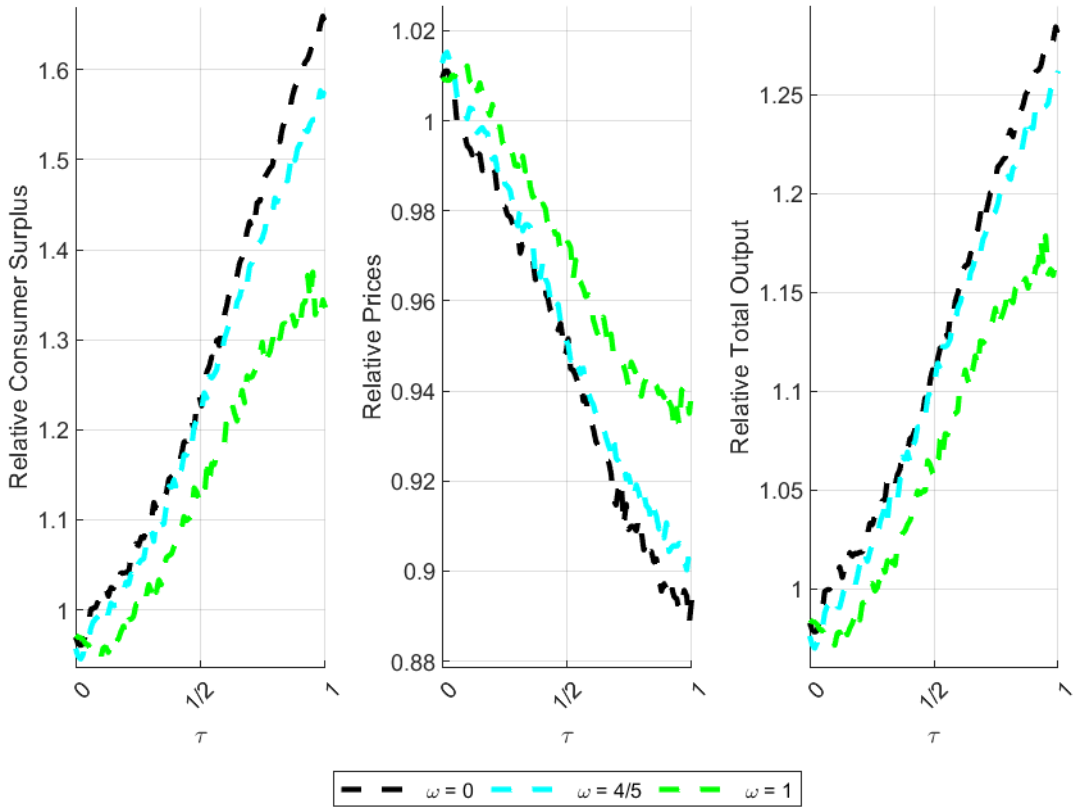
Figure 7. Proportion of Executed Platform Action Across Varying ω Values Across a Grid of 100 Values of Across a Grid of 100 c Values.



Note: Action 2 represents the platform's profit-maximizing recommendation on average.

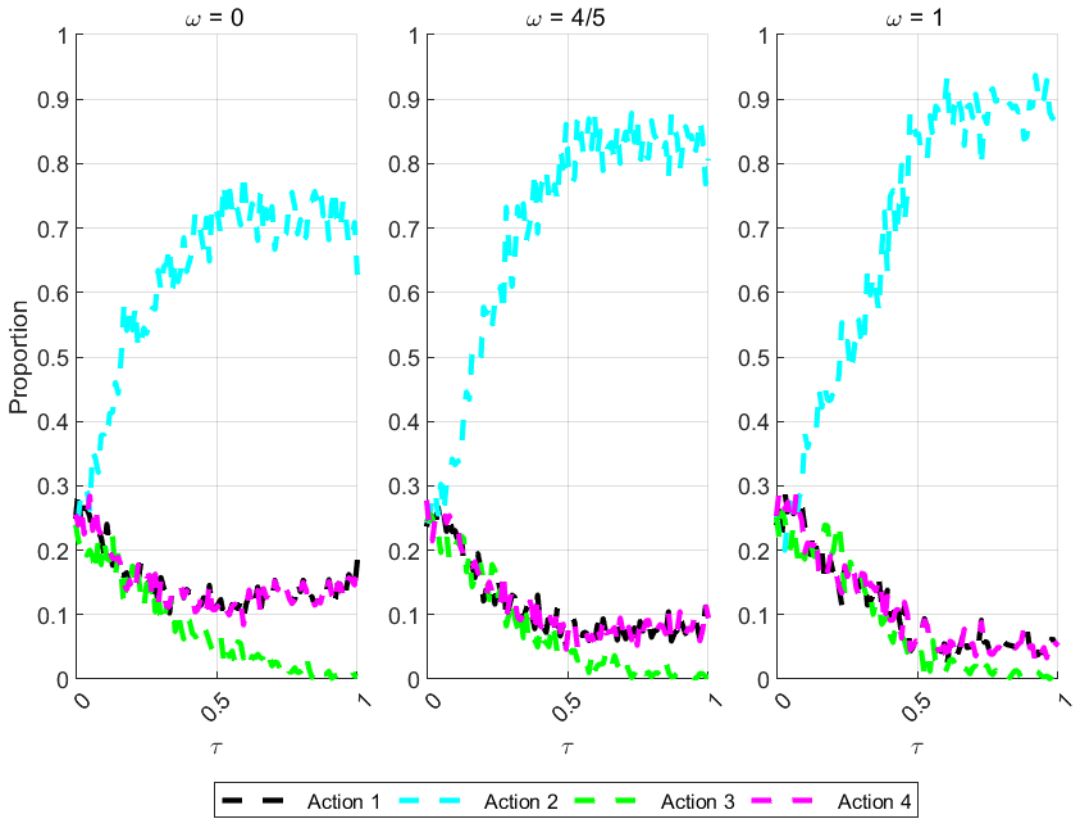
Figure 8 illustrates the evolution of the primary outcome variables relative to the Bertrand-Nash benchmark as $\tau \to 1$. As a larger proportion of consumers follow the recommendation algorithm—opting predominantly for their preferred products, as confirmed by Figure 9—both consumer welfare and total output significantly surpass the Bertrand-Nash levels, while prices concurrently drop below this benchmark. These findings imply that even moderate adherence to product recommendations (approximately above 10%) effectively mitigates autonomous algorithmic anticompetitive behavior, ultimately benefiting consumers and enhancing overall welfare.

Figure 8. Ratio of Consumer Surplus (Left Panel), Prices (Middle Panel), and Total Output (Right Panel) to the Bertrand-Nash Outcome Across a Grid of 100 τ Values.



Note: Prices are averaged across sellers. Total output represents the sum of each seller's market share. Without the RS, the Bertrand-Nash Outcome is the $\tau = 0$ case.

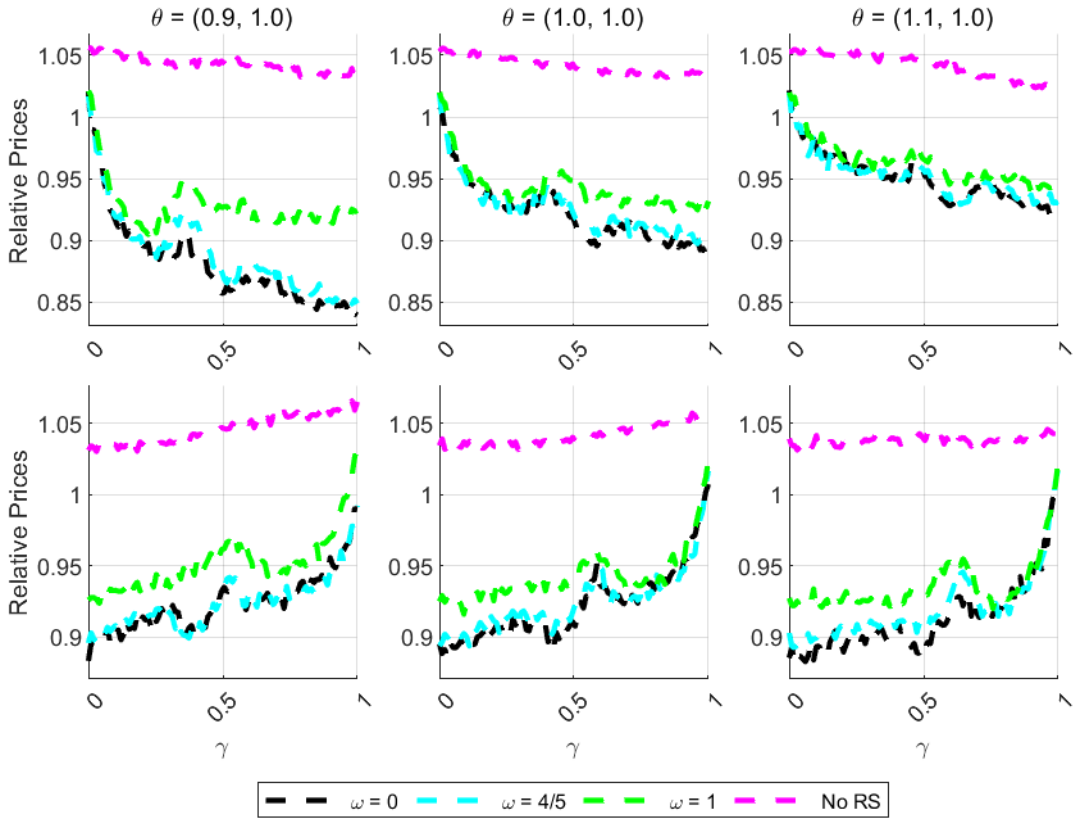
Figure 9. Proportion of Executed Platform Action Across Varying ω Values Across a Grid of 100 Values of τ Values.



Note: Action 2 represents the platform's profit-maximizing recommendation on average.

Figure 10 shows how the ratios of seller one and seller two prices to the Bertrand–Nash outcomes evolve as the share of type one consumers (γ_1) approaches one for varying values of type one's price sensitivity (θ_1). Since each unique (γ_1, θ_1) pair yields a unique equilibrium, this analysis involves computing the Bertrand–Nash outcome for every pair and then evaluating the corresponding ratios. In general, prices remain below the competitive outcome when the platform uses an RS while being above the competitive benchmark without it, except in edge cases where they are approximately equal to the Bertrand–Nash level. As $\gamma_1 \to 1$, the relative price for seller one diverges downward, while the opposite trend is observed for seller two. This is because, with the pricing algorithm's objective being profit maximization, seller one—enjoying a dominant share of consumers—can afford to lower prices while still maintaining high profit levels. Conversely, seller two, which loses nearly all consumer preference as γ_1 approaches one, must increase its prices to compensate for the diminished recommended demand.

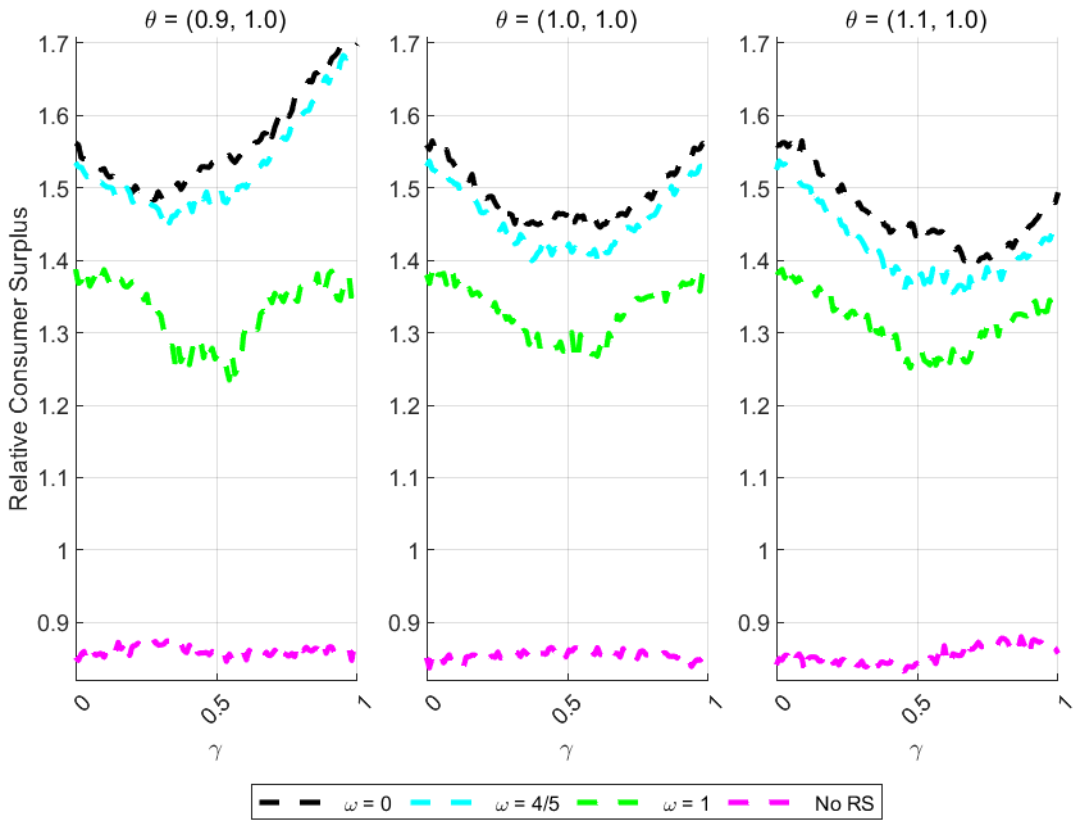
Figure 10. Ratio of Seller One Prices (Top Panel) and Seller Two Prices (Bottom Panel) to the Bertrand–Nash Outcomes Across a Grid of 100 γ Values.



Note: With the RS, the Bertrand–Nash outcomes are for the $\tau = 3/4$ case where the platform recommends seller one to consumer type one and seller two to consumer type two. Without the RS, the Bertrand-Nash Outcome is the $\tau = 0$ case.

Figure 11 extends this analysis to relative consumer surplus. As expected, consumer welfare declines as the price sensitivity of consumer type one increases, particularly when $\gamma_1 \to 1$. Moreover, even when the platform completely disregards consumer welfare ($\omega = 1$), the resulting consumer surplus remains substantially above the competitive benchmark—gains of up to approximately 40%—and these gains increase to roughly 70% when consumers are less price sensitive.

Figure 11. Ratio of Consumer Surplus to the Bertrand–Nash Outcome Across a Grid of 100 γ Values.



Note: With the RS, the Bertrand–Nash outcomes are for the $\tau = 3/4$ case where the platform recommends seller one to consumer type one and seller two to consumer type two. Without the RS, the Bertrand-Nash Outcome is the $\tau = 0$ case.

D. *Endogenous* τ

A critical aspect of my model is the parameter τ, which governs the share of consumers that choose their recommended product versus those that choose to view all available alternatives. Thus far, τ has been set exogenously. In reality, it is likely that consumers who view searching through alternatives as costly will be more likely to select their recommended option and not elect to browse all available options. To that end, I endogenize τ with respect to the search cost c_j for each consumer type j and assume consumers search in a simultaneous fashion.

First, recall the equation for aggregate consumer surplus at time t is given by

$$U_t = \mu\Bigg[\tau \sum_{i=1}^{n} \sum_{j \in \mathcal{J}_{it}} \frac{\gamma_j}{\theta_j} \ln\left(1 + \exp\left(\frac{a_{ij} - \theta_j p_{it}}{\mu}\right)\right) + (1-\tau) \sum_{j=1}^{k} \frac{\gamma_j}{\theta_j} \ln\left(1 + \sum_{i=1}^{n} \exp\left(\frac{a_{ij} - \theta_j p_{it} - c_j}{\mu}\right)\right)\Bigg].$$

Since each consumer type j is only recommend a single firm i in each time period, the expected consumer welfare j gets from its recommendation i in this time period is

$$\mathbb{E}\left[U_{jt}^{rec}\right] = \mu \ln\left(1 + \exp\left(\frac{a_{ij} - \theta_j p_{it}}{\mu}\right)\right),$$

while

$$\mathbb{E}\left[U_{jt}^{search}\right] = \mu \ln\left(1 + \sum_{i'=1}^{n} \exp\left(\frac{a_{i'j} - \theta_j p_{i't} - c_j}{\mu}\right)\right)$$

represents the expected consumer surplus for type j at time t from searching.[23]

Denote c_j^* as the value at which consumer type j is indifferent between following the recommendation and searching. By setting the two above expected consumer welfares equal to each other and solving for c_j^*, it follows that

$$c_j^* = \mu \ln\left(\frac{\sum_{i'=1}^{n} \exp\left(\frac{a_{i'j} - \theta_j p_{i't}}{\mu}\right)}{\exp\left(\frac{a_{ij} - \theta_j p_{it}}{\mu}\right)}\right).$$

Consumers of type j will search more as their indifference point c_j^* decreases, while electing their recommended option as this cutoff rises. I assume $c_j \sim \mathbb{U}[0,1]$, implying that $F\left(c_j^*\right) = \mathbb{P}\left(c_j \leq c_j^*\right) = c_j^*$. Given τ_j is the share of consumers of type j who follow the recommendation, i.e., those consumers with $c_j > c_j^*$, it follows that τ_j is inversely related to c_j^* such that

$$\tau_j = \begin{cases} 1 & \text{if } c_j^* \leq 0 \\ 1 - c_j^* & \text{if } 0 < c_j^* < 1 \\ 0 & \text{if } c_j^* \geq 1. \end{cases}$$

[23] Note, the γ_j/θ_j terms are not shown here because these expressions represent the expected utility of a single type j consumer. The γ_j (population weight) and θ_j (marginal utility of income) only appear when aggregating over types to compute total consumer surplus in money-metric terms.

Table 5. Outcomes At Convergence for Endogenous τ.

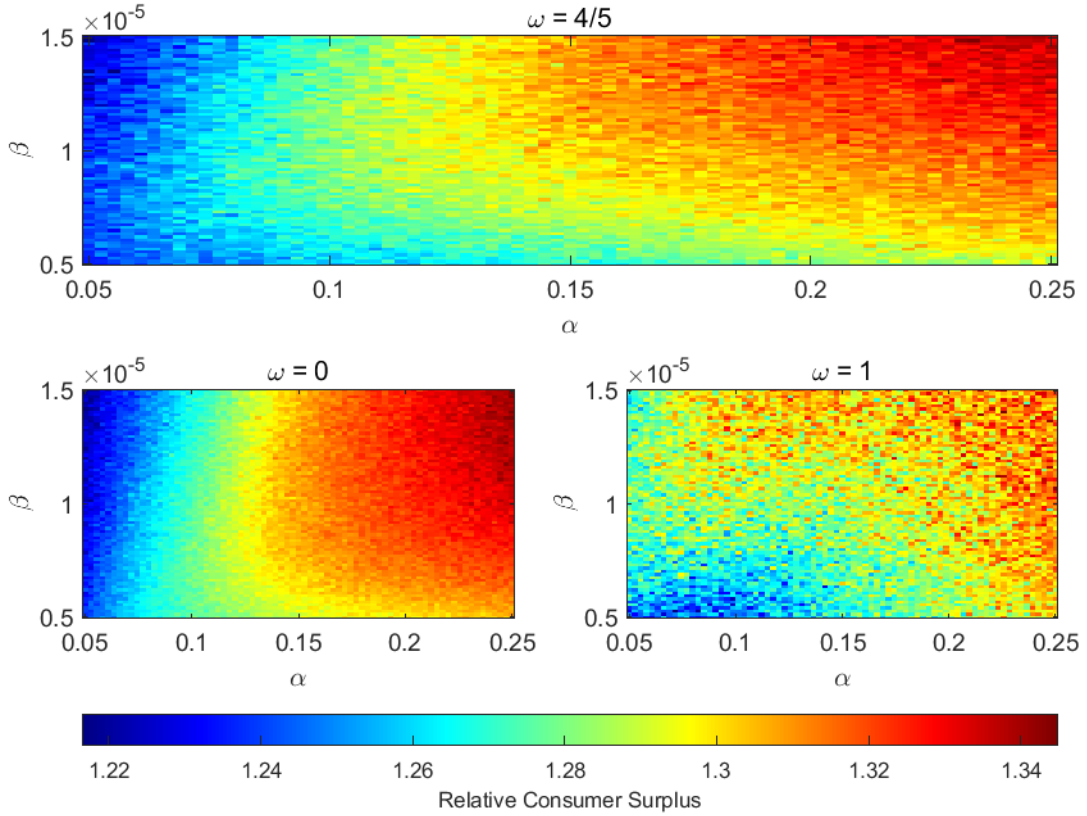
	Total Output	**Average Prices**	**Total Revenues**	**Consumer Surplus**
NL1	0.603	1.864	0.900	0.233
NL2	0.737	1.706	1.004	0.340
AI (Fixed τ), $\omega = 0$	0.751	1.682	0.999	0.360
AI (Fixed τ), $\omega = 4/5$	0.744	1.697	1.002	0.349
AI (Fixed τ), $\omega = 1$	0.713	1.735	0.984	0.319
AI (Endogenous τ), $\omega = 0$	0.765	1.655	0.990	0.393
AI (Endogenous τ), $\omega = 4/5$	0.757	1.674	0.991	0.379
AI (Endogenous τ), $\omega = 1$	0.709	1.732	0.967	0.330

Note: These results represent averages over the last 100,000 time periods prior to convergence.

E. *Learning Parameter Robustness*

A core component of any AI-driven pricing algorithm is the set of learning parameters that shape its behavior. To assess their influence, I evaluate algorithmic performance in the baseline environment by varying these parameters over a grid of 75 (α, β) combinations, representing different learning rates and degrees of experimentation.

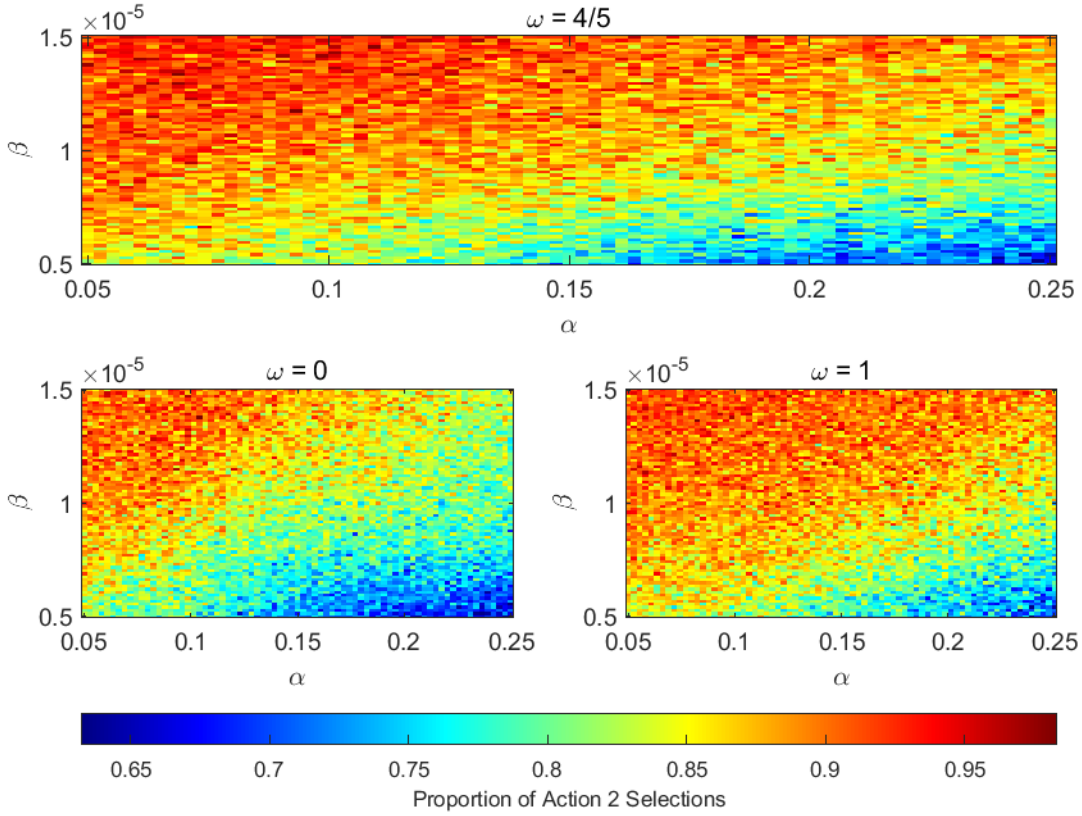
Figure 12. Ratio of Consumer Surplus to the Bertrand–Nash Outcome Across a Grid of (α, β) Pairs.



Note: The Bertrand–Nash outcomes are for the $\tau = 3/4$ case where the platform recommends seller one to consumer type one and seller two to consumer type two. The grid has 75 values of both α and β giving rise to 5,625 unique (α, β) pairs.

Figure 12 reports consumer surplus relative to the Bertrand–Nash benchmark, while Figure 13 shows the frequency with which the platform selects action 2—its profit-maximizing recommendation on average. The results indicate that the platform consistently chooses the optimal action across a wide range of learning parameters. More importantly, consumer surplus remains substantially above the Bertrand–Nash outcome (by at least 22%) in all cases, including when the platform fully prioritizes profits ($\omega = 1$). These findings reinforce the robustness of the main result: the platform's RS mechanism effectively curbs supracompetitive pricing, even when the underlying algorithms vary in how aggressively they learn and explore.

Figure 13. Proportion of Platform Executing Action 2 Across a Grid of (α, β) Pairs.



Note: Action 2 represents the platform's profit-maximizing recommendation on average. The grid has 75 values of both α and β giving rise to 5,625 unique (α, β) pairs.

VI. Conclusion

This paper investigates algorithmic pricing, recommendation systems (RSs), and competition through a model of Bertrand-Markov competition among firms using Q-learning pricing algorithms, alongside an AI-powered recommendation system (RS) strategically influencing consumer product visibility. The analysis provides a core insight into how pricing algorithms can perform when acting on a platform using an AI-based RS. Introducing this RS significantly alters market dynamics by actively mitigating the ability of pricing algorithms to tacitly reach supracompetitive outcomes. The RS, driven by consumer preferences and price information, disrupts the stability of anticompetitive levels, compelling pricing algorithms toward prices even below the competitive benchmark. This occurs even when the platform prioritizes seller profits exclusively, underscoring the robust pro-competitive effect of RSs. Notably, this result holds across diverse consumer heterogeneity, market, and learning parameter conditions, demonstrating the significant pro-competitive effects RSs can product in digital markets.

These findings carry significant policy implications for regulators addressing competitive concerns in such digital economies. The results suggest that regulatory attention should perhaps shift toward overseeing platform-level RSs rather than exclusively targeting sellers' individual pricing algorithms. Enhanced oversight of such RSs could promote competitive pricing behaviors like shown in this paper, thereby improving consumer welfare outcomes. Strengthening oversight of platform RSs may have at least two positive impacts on the market: (1) lower prices and higher consumer welfare resulting from inhibiting autonomous algorithmic supracompetitive behavior and (2) mitigating platform self-preferencing, a topic not thoroughly discussed here, but of growing concern among competition authorities.

Finally, while this paper utilizes reinforcement learning-based RSs, the findings motivate further investigation into more commonly used non-RL-based recommendation systems, such as collaborative filtering algorithms. Given their widespread application, understanding whether these standard RSs similarly mitigate or exacerbate algorithmic anticompetitive conduct would offer valuable insights for policy development. Future research exploring these additional dimensions will be crucial for comprehensive antitrust policy recommendations in increasingly AI-driven marketplaces. Moreover, additional research could extend this model by examining multi-platform competition, vertically integrating the platform into the market, or empirically testing these theoretical predictions with real-world data from digital platforms that use RSs and have sellers operating with pricing algorithms.

References

Abada, **Ibrahim**, **Joseph E. Harrington Jr.**, **Xavier Lambin**, and **Janusz M Meylahn** (2024). "Algorithmic Collusion: Where Are We and Where Should We Be Going?" *SSRN*. DOI: https://dx.doi.org/10.2139/ssrn.4891033.

Agarwal, **Alekh**, **Nan Jiang**, **Sham M. Kakade**, and **Wen Sun** (2022). *Reinforcement Learning: Theory and Algorithms*. 1st. URL: https://rltheorybook.github.io/rltheorybook_AJKS.pdf.

Asker, **John**, **Chaim Fershtman**, and **Ariel Pakes** (2022). "Artificial Intelligence, Algorithmic Design, and Pricing". *American Economic Association* 112, pp. 452–56. DOI: 10.1257/pandp.20221059.

Assad, **Stephanie**, **Robert Clark**, **Daniel Ershov**, and **Lei Xu** (2024). "Algorithmic Pricing and Competition: Empirical Evidence from the German Retail Gasoline Market". *Journal of Political Economy* 132.3. DOI: https://doi.org/10.1086/726906.

Auer, **Peter** (2002). "Using Confidence Bounds for Exploitation-Exploration Trade-offs". *Journal of Machine Learning Research* 3, pp. 397–422. URL: https://dl.acm.org/doi/10.5555/944919.944941.

Banchio, **Martino** and **Giacomo Mantegazza** (2022). "Artificial Intelligence and Spontaneous Collusion". *Social Science Research Network (SSRN)*. DOI: https://dx.doi.org/10.2139/ssrn.4032999.

Banchio, **Martino** and **Andrzej Skrzypacz** (2022). "Artificial Intelligence and Auction Design". DOI: https://doi.org/10.48550/arXiv.2202.05947.

Brasic, **William** (2024). "When Asymmetric Pricing Algorithms Collide". URL: https://williambrasic.com/Research/When_Asymmetric_Pricing_Algorithms_Collide_V2.pdf.

Brown, **Zach Y.** and **Alexander MacKay** (2024). "Algorithmic Coercion with Faster Pricing". *Social Science Research Network (SSRN)*. DOI: https://dx.doi.org/10.2139/ssrn.4380895.

Brown, **Zach Y.** and **Alexander MacKay** (2023). "Competition in Pricing Algorithms". *American Economic Journal: Microeconomics* 15.2, pp. 109–156. DOI: https://doi.org/10.1257/mic.20210158.

Busoniu, **Lucian**, **Bart De Schutter**, and **Robert Babuska** (2008). "A Comprehensive Survey of Multiagent Reinforcement Learning". *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 38.2, pp. 156–172. DOI: https://doi.org/10.1109/TSMCC.2007.913919.

Calder-Wang, **Sophie** and **Gi Heung Kim** (2024). "Coordinated vs Efficient Prices: The Impact of Algorithmic Pricing on Multifamily Rental Markets". URL: https://cowles.yale.edu/sites/default/files/2024-05/Calder-Wang-main.pdf.

Calvano, **Emilio**, **Giacoma Calzolari**, **Vincenzo Denicolò**, and **Sergio Pastorello** (2019). "Algorithmic Pricing: What Implications for Competition Policy?" *Review of Industrial Organization* 55.9, pp. 155–171. DOI: https://doi.org/10.1007/s11151-019-09689-3.

— (2020). "Artificial Intelligence, Algorithmic Pricing, and Collusion". *American Economic Review* 110.10, pp. 3267–3297. DOI: https://doi.org/10.1257/aer.20190623.

— (2023). "Artificial Intelligence, Algorithmic Recommendations, and Competition". *SSRN*. DOI: https://dx.doi.org/10.2139/ssrn.4448010.

Chen, **Le**, **Alan Mislove**, and **Christo Wilson** (2016). "An Empirical Analysis of Algorithmic Pricing on Amazon Marketplace". *Proceedings of the 25th International World Wide Web Conference (WWW 2016)*, pp. 1339–1349. DOI: https://dl.acm.org/doi/10.1145/2872427.2883089.

Chen, **Nan** and **Hsin-Tien Tsai** (2024). "Steering via Algorithmic Recommendations". *RAND Journal of Economics* 55.4, pp. 501–518. DOI: https://doi.org/10.1111/1756-2171.12481.

Devogele, **Sophie** (2023). "Algorithmic Tacit Collusion: A Threat to the Current EU Competition Law Framework?" URL: https://mededingingscongres.nl/wp-content/uploads/2023/10/Thesis-final-version-PDF.pdf.

Ezrachi, **Ariel** and **Maurice E. Stucke** (2017). "Artificial Intelligence and Collusion: When Computers Inhibit Competition". *University of Illinois Law Review* 2017.5, pp. 1775–1810. URL: https://www.illinoislawreview.org/wp-content/uploads/2017/10/Ezrachi-Stucke.pdf.

— (2020). "Sustainable and Unchallenged Algorithmic Tacit Collusion". *Northwestern Journal of Technology and Intellectual Property* 17.2, pp. 217–260. URL: https://scholarlycommons.law.northwestern.edu/njtip/vol17/iss2/2.

Farronato, **Chiara**, **Andrey Fradkin**, and **Alexander MacKay** (2023). "Self-Preferencing At Amazon: Evidence from Search Rankings". *AEA Papers and Proceedings* 113, pp. 239–243. DOI: https://doi.org/10.1257/pandp.20231068.

Fershtman, **Chaim** and **Ariel Pakes** (2012). "Dynamic Games with Asymmetric Information: A Framework for Empirical Work". *Quarterly Journal of Economics* 127.4, pp. 1611–1661. DOI: https://doi.org/10.1093/qje/qjs025.

Fish, **Sara**, **Yannai A. Gonczarowski**, and **Ran Shorrer** (2024). "Algorithmic Collusion by Large Language Models". DOI: https://arxiv.org/pdf/2404.00806.

Fletcher, **Amelia**, **Peter L Ormosi**, and **Rahul Savani** (2023). "Recommender Systems and Supplier Competition on Platforms". *Journal of Competition Law and Economics* 19.3, pp. 397–426. DOI: https://doi.org/10.1093/joclec/nhad009.

Fortin, **John A.** (2021). "Algorithms and Conscious Parallelism: Why Current Antitrust Doctrine is Prepared for the Twenty-First Century Challenges Posed by Dynamic Pricing". *Tulane Journal of Technology and Intellectual Property* 23. URL: https://journals.tulane.edu/TIP/article/view/3648.

Frick, **Kevin Michael** (2023). "Convergence Rates and Collusive Outcomes of Pricing Algorithms". *Social Science Research Network (SSRN)*. DOI: https://dx.doi.org/10.2139/ssrn.4527452.

Friedman, **James** (1971). "A Non-cooperative Equilibrium for Supergames". *Review of Economic Studies* 38.1, pp. 1–12. DOI: https://doi.org/10.2307/2296617.

Gupta, **Kirti** and **Avigail Kifer** (2024). "Algorithms, Artificial Intelligence, and Antitrust: An Overview". *The Antitrust Source*. URL: https://www.cornerstone.com/wp-content/uploads/2024/02/Algorithms-Aritifical-Intelligence-and-Antitrust.pdf.

Harrington, **Joseph E.** (2018). "Developing Competition Law for Collusion by Autonomous Artificial Agents". *Journal of Competition Law and Economics* 14.3, pp. 331–363. DOI: https://doi.org/10.1093/joclec/nhy016.

Hartline, **Jason D.**, **Sheng Long**, and **Chenhao Zhang** (2024). "Regulation of Algorithmic Collusion". *CSLAW'24: Proceedings of the Symposium on Computer Science and Law*, pp. 98–108. DOI: https://doi.org/10.1145/3614407.3643706.

Hettich, **Matthias** (2021). "Algorithmic Collusion: Insights from Deep Learning". *Social Science Research Network (SSRN)*. DOI: https://dx.doi.org/10.2139/ssrn.3785966.

Hovenkamp, **Herbert** (2023). "Antitrust and Self-Preferencing". *Antitrust* 38.1, pp. 5–12. URL: https://www.americanbar.org/content/dam/aba/publications/antitrust/magazine/2023/vol-38-issue-1/antitrust-and-self-preferencing.pdf.

Hu, **Michael** (2023). *The Art of Reinforcement Learning: Fundamentals, Mathematics, and Implementations with Python*. 1st. Berkeley, CA: Apress. ISBN: 9781484296059. URL: https://link.springer.com/book/10.1007/978-1-4842-9606-6.

Johnson, **Justin P.**, **Andrew Rhodes**, and **Matthijs Wildenbeest** (2024). "Algorithmic Steering and Advertising on Platforms".

— (2023). "Platform Design When Sellers Use Pricing Algorithms". *Econometrica* 91.5, pp. 1841–1879. DOI: https://doi.org/10.3982/ECTA19978.

Kittaka, **Yuta**, **Susumu Sato**, and **Yusuke Zennyo** (2023). "Self-Preferencing by Platforms: A Literature Review". *Japan and The World Economy* 66.101191, pp. 1841–1879. DOI: https://doi.org/10.1016/j.japwor.2023.101191.

Klein, **Timo** (2021). "Autonomous Algorithmic Collusion: Q-learning Under Sequential Pricing". *The RAND Journal of Economics* 52.3, pp. 538–558. DOI: https://doi.org/10.1111/1756-2171.12383.

Lee, **Kwok Hao** and **Leon Musolff** (2023). "Entry Into Two-Sided Markets Shaped by Platform-Guided Search". *RR at Econometrica*. URL: https://lmusolff.github.io/papers/Entry_and_Platform_Guided_Search.pdf.

Leisten, **Matthew** (2024). "Algorithmic Competition, with Humans". *Social Science Research Network (SSRN)*. DOI: https://dx.doi.org/10.2139/ssrn.4733318.

MacKay, **Alexander** and **Samuel N. Weinstein** (2022). "Dynamic Pricing Algorithms, Consumer Harm, and Regulatory Response". *Washington University Law Review* 100, pp. 111–274. URL: https://wustllawreview.org/2022/11/25/dynamic-pricing-algorithms-consumer-harm-and-regulatory-response/.

Maskin, **Eric** and **Jean Tirole** (1988). "A Theory of Dynamic Oligopoly, II: Price Competition, Kinked Demand Curves, and Edgeworth Cycles". *Econometrica* 56.3, pp. 571–599. DOI: https://doi.org/10.2307/1911701.

Mazumdar, **Aneesa** (2022). "Algorithmic Collusion: Reviving Section 5 of the FTC Act". *Columbia Law Review* 122.2, pp. 449–488. URL: https://www.jstor.org/stable/10.2307/27114356.

McSweeny, **Terrell** and **Brian O'Dea** (2017). "The Implications of Algorithmic Pricing for Coordinated Effects Analysis and Price Discrimination Markets in Antitrust Enforcement". *Antitrust* 32.1, pp. 75–81. URL: https://www.ftc.gov/system/files/documents/public_statements/1286183/mcsweeny_and_odea_-_implications_of_algorithmic_pricing_antitrust_fall_2017_0.pdf.

Mehra, **Salil K.** (2016). "Antitrust and the Robo-Seller: Competition in the Time of Algorithms". *Minnesota Law Review* 100, pp. 1323–1375. URL: https://scholarship.law.umn.edu/mlr/204.

Miklós-Thal, **Jeanine** and **Catherine Tucker** (2019). "Collusion by Algorithm: Does Better Demand Prediction Facilitate Coordination Between Sellers?" *Management Science* 65.4, pp. 1552–1561. DOI: https://doi.org/10.1287/mnsc.2019.3287.

Musolff, **Leon** (2024). "Algorithmic Pricing, Price Wars and Tacit Collusion: Evidence from E-Commerce". URL: https://lmusolff.com/papers/Algorithmic_Pricing.pdf.

Nazzini, **Renato** and **James Henderson** (2024). "Overcomming the Current Knowledge Gap of Algorithmic "Collusion" and the Fole of Computational Antitrust". *Stanford Computational Antitrust* 4. URL: https://law.stanford.edu/wp-content/uploads/2024/02/Algorithmic-Collusion.pdf.

Rummery, **G.A.** and **Mahesan Niranjan** (1994). "On-Line Q-Learning Using Connectionist Systems". URL: https://www.researchgate.net/publication/2500611_On-Line_Q-Learning_Using_Connectionist_Systems.

Schrepel, **Thibault** (2020). "The Fundamental Unimportance of Algorithmic Collusion for Antitrust Law". *Columbia Law Review*. URL: https://jolt.law.harvard.edu/digest/the-fundamental-unimportance-of-algorithmic-collusion-for-antitrust-law.

Waltman, **Ludo** and **Uzay Kaymack** (2008). "Q-learning Agents in a Cournot Oligopoly Model". *Journal of Economic Dynamics and Control* 32.10, pp. 3275–3293. DOI: https://doi.org/10.1016/j.jedc.2008.01.003.

Watkins, **Christopher J.C.H** and **Peter Dayan** (1992). "Q-learning". *Machine Learning* 8, pp. 279–292. DOI: https://doi.org/10.1007/BF00992698.

Xu, **Xingchen**, **Stephanie Lee**, and **Yong Tan** (2023). "Algorithmic Collusion or Competition: The Role of Platforms' Recommender Systems". URL: https://arxiv.org/abs/2309.14548.

Yang, **Yaodong** and **Jun Wang** (2020). "An Overview of Multi-agent Reinforcement Learning from Game Theoretical Perspective". URL: https://arxiv.org/abs/2011.00583.

VII. Appendix

A. *Baseline Specification*

A..1 *Baseline Parameters*

The product preference matrix a is given by

$$a = \begin{bmatrix} 2 & 1.9 \\ 1.9 & 2 \end{bmatrix}$$

A..2 *Without RS*

Table A.2. Percentage Change from Bertrand-Nash Outcome

	Average	Seller 1	Seller 2
Profits	6.72%	6.69%	6.75%
Revenues	-4.37%	-4.30%	-4.43%
Demand	-7.89%	-7.80%	-7.99%
Prices	4.01%	4.00%	4.02%
CS	-14.14%		

Table A.1. Economic environment parameters

Parameter	**Value**
Number of firms (n)	2
Number of possible prices (m_i)	15
Number of possible recommendations (m)	4
Price step size (ν)	$(2.1 - 1.0)/(m - 1)$
Firm pricing space ($\mathcal{A}_i = \mathcal{A}_{-i}$)	Identical
Firm state space ($\mathcal{S}_i = \mathcal{S}_{-i}$)	Identical
Marginal cost (mc)	1
Inverse index of aggregate demand (a_0)	0
Horizontal differentiation index (μ)	1/4
Consumer share who only see recommendation (τ)	3/4
Platform profit weight (ω)	$\{0,\ 4/5,\ 1\}$
Royalty share (f)	0.2
Number of consumer types (k)	2
Proportion of consumer type 1 (γ_1)	1/2
Price sensitivity (θ_j)	1
Search cost (c_j)	1/4
Discount factor ($\delta_i,\ \delta$)	0.95
Learning rate (β_i)	10^{-5}
Exploration rate ($\alpha_i,\ \alpha$)	0.15
Memory for sellers and platform ($q_i,\ q$)	1

A..3 *With RS*

Table A.3. Percentage Change from Bertrand-Nash Outcome for Different ω Values

	$\omega = 0$			$\omega = 4/5$			$\omega = 1$		
	Average	**Seller 1**	**Seller 2**	**Average**	**Seller 1**	**Seller 2**	**Average**	**Seller 1**	**Seller 2**
Profits	-15.89%	-15.68%	-16.09%	-12.76%	-12.80%	-12.72%	-8.09%	-8.18%	-8.00%
Revenues	8.41%	7.66%	9.16%	8.60%	9.25%	7.95%	6.74%	7.06%	6.43%
Demand	19.85%	18.65%	21.04%	18.66%	19.64%	17.68%	13.73%	14.23%	13.22%
Prices	-8.54%	-8.14%	-8.93%	-7.70%	-8.00%	-7.39%	-5.64%	-5.76%	-5.52%
CS	44.69%			40.35%			28.21%		

Results averaged across $E = 100$ episodes and averaged over the last 100,000 time periods prior to convergence within each episode.

B. *Increased Platform Fee f*

This section gives results upon convergence relative to the Bertrand-Nash level increasing the platform royalty fee from $f = 0.2$ to $f = 0.3$. This percentage royalty fee being increased leads to higher effective marginal cost $mc/(1 - f)$ pushing equilibrium prices upward.[24] So, I increase the seller action space from $m = 15$ equally spaced points between 1 and 2.1 to the same number points equally spaced between 1.3 and 2.4 meaning equilibrium prices are more centered in each seller's action space.[25] Consumer

[24] See the end of the appendix for further justification.

[25] See the next section of the appendix for results with this same action space and $f = 0.2$.

welfare and prices remain below and above the Bertrand-Nash level without the platform's RS, respectively, while the opposite holds when the platform does implement an RS.

B..1 *Without RS*

Table A.4. Percentage Change from Bertrand-Nash Outcome

	Tot/Avg	**Firm 1**	**Firm 2**
Profits	4.50%	4.47%	4.54%
Revenues	-7.33%	-7.28%	-7.39%
Demand	-10.32%	-10.24%	-10.39%
Prices	3.49%	3.48%	3.49%
CS	-15.32%		

B..2 *With RS*

Table A.5. Percentage Change from Bertrand-Nash Outcome for Different ω Values

	$\omega = 0$			$\omega = 4/5$			$\omega = 1$		
	Total/Avg	**Firm 1**	**Firm 2**	**Total/Avg**	**Firm 1**	**Firm 2**	**Total/Avg**	**Firm 1**	**Firm 2**
Profits	-12.36%	-12.24%	-12.47%	-8.12%	-8.32%	-7.92%	-7.56%	-7.53%	-7.58%
Revenues	10.81%	11.00%	10.61%	7.15%	7.10%	7.20%	4.95%	3.00%	6.91%
Demand	18.92%	19.15%	18.69%	12.50%	12.50%	12.50%	9.34%	6.69%	11.98%
Prices	-5.58%	-5.72%	-5.44%	-3.91%	-4.00%	-3.82%	-2.95%	-2.38%	-3.53%
CS	34.68%			22.51%			17.06%		

Results averaged across $E = 100$ episodes and over the last 100,000 time periods prior to convergence within each episode.

C. *Altered Seller Action Space*

Given the Bertrand-Nash prices are relatively close to the upper bound of the baseline pricing grid, I conduct the baseline analysis again using a discrete pricing grid of $m = 15$ equally spaced points between 1.3 and 2.4. Now, the midpoint is roughly equal to competitive benchmark. This results in consumer welfare not reaching levels as high as the baseline specification as prices are converging to slightly higher levels, but consumer surplus is still well above the Bertrand-Nash level across all considered ω values. Thus, autonomous algorithmic supracompetitive behavior is mitigated in this case as well.

C..1 *Without RS*

Table A.6. Percentage Change from Bertrand-Nash Outcome

	Tot/Avg	**Firm 1**	**Firm 2**
Profits	9.45%	9.69%	9.21%
Revenues	-8.42%	-8.11%	-8.72%
Demand	-14.10%	-13.78%	-14.43%
Prices	6.92%	6.86%	6.98%
CS	-23.69%		

C..2 *With RS*

Table A.7. Percentage Change from Bertrand-Nash Outcome for Different ω Values

	$\omega = 0$			$\omega = 4/5$			$\omega = 1$		
	Average	**Seller 1**	**Seller 2**	**Average**	**Seller 1**	**Seller 2**	**Average**	**Seller 1**	**Seller 2**
Profits	-8.82%	-8.93%	-8.71%	-7.95%	-7.69%	-8.22%	-7.99%	-6.79%	-9.19%
Revenues	4.76%	6.17%	3.34%	3.19%	4.22%	2.15%	1.77%	1.79%	1.76%
Demand	11.15%	13.29%	9.01%	8.43%	9.83%	7.03%	6.37%	5.83%	6.91%
Prices	-4.59%	-5.27%	-3.90%	-3.49%	-3.86%	-3.12%	-2.93%	-2.77%	-3.09%
CS	23.59%			18.51%			14.53%		

Results averaged across $E = 100$ episodes and averaged over the last 100,000 time periods prior to convergence within each episode.

D. *Limited State Space*

The baseline analysis with the RS considered the case of both seller's and the platform's state in each period consisting of the prior period seller prices along with the prior period platform's recommendation. Here, I remove the platform recommendation component of the state space so that both the pricing algorithms and the RS make action decisions only based on the prior period seller prices. Results are fairly stable, with consumer welfare and prices being well above and below the Bertrand-Nash benchmark, respectively.

Table A.8. Percentage Change from Bertrand-Nash Outcome for Different ω Values

	$\omega = 0$			$\omega = 4/5$			$\omega = 1$		
	Average	**Seller 1**	**Seller 2**	**Average**	**Seller 1**	**Seller 2**	**Average**	**Seller 1**	**Seller 2**
Profits	-15.17%	-14.15%	-16.19%	-11.73%	-11.73%	-11.72%	-8.84%	-9.54%	-8.13%
Revenues	4.82%	5.76%	3.87%	5.19%	5.24%	5.15%	4.94%	3.75%	6.13%
Demand	14.23%	15.13%	13.32%	13.16%	13.23%	13.10%	11.42%	10.01%	12.84%
Prices	-6.70%	-6.64%	-6.76%	-5.94%	-5.90%	-5.98%	-5.05%	-4.96%	-5.14%
CS	32.96%			29.12%			24.15%		

Results averaged across $E = 100$ episodes and averaged over the last 100,000 time periods prior to convergence within each episode.

E. ϵ-*Greedy RS Action Selection*

The baseline specification considers the platform's RS using the UCB action selection mechanism. Of course, there are a variety of different ways bandit and RL agents can make action selections. To that end, this section considers the RS using ϵ-greedy action selection with experimentation parameter $\beta = 10^{-5}$, meaning both the platform and the sellers employ the same action selection strategy. The results show that while UCB does better in terms of higher consumer welfare and lower prices, the core result of a platform's RS resulting in lower prices and higher consumer welfare relative to the Bertrand-Nash benchmark still holds suggesting this core finding is robust to variations in the action selection mechanism used by the RS.

Table A.9. Percentage Change from Bertrand-Nash Outcome for Different ω Values

	$\omega = 0$			$\omega = 4/5$			$\omega = 1$		
	Total/Avg	**Firm 1**	**Firm 2**	**Total/Avg**	**Firm 1**	**Firm 2**	**Total/Avg**	**Firm 1**	**Firm 2**
Profits	-14.14%	-14.04%	-14.24%	-9.77%	-9.87%	-9.67%	-4.95%	-5.08%	-4.81%
Revenues	7.64%	7.07%	8.20%	6.11%	5.66%	6.57%	3.44%	4.05%	2.83%
Demand	17.89%	17.01%	18.77%	13.59%	12.97%	14.22%	7.39%	8.35%	6.43%
Prices	-7.43%	-6.97%	-7.89%	-5.59%	-5.50%	-5.69%	-3.07%	-3.32%	-2.81%
CS	39.55%			29.16%			15.14%		

Results averaged across $E = 100$ episodes and over the last 100,000 time periods prior to convergence within each episode.

F. *Logit Equilibria*

F..1 *Bertrand-Nash*

Theorem 1. *Assume that the set of consumers firm i is recommended to is $\mathcal{J}_{it} \subseteq \{1, 2, \ldots, k\}$. The Bertrand-Nash equilibrium price for firm i is given by*

$$p_i^* = \frac{mc}{1-f} + \frac{\mu\left(\tau \sum_{j\in\mathcal{J}_{it}} \gamma_j R_{ij}(p_i^*) + (1-\tau)\sum_{j=1}^{k} \gamma_j S_{ij}(p_i^*, p_{-i}^*)\right)}{\left[\tau \sum_{j\in\mathcal{J}_{it}} \gamma_j \theta_j R_{ij}(p_i^*)\left[1 - R_{ij}(p_i^*)\right] + (1-\tau)\sum_{j=1}^{k} \gamma_j \theta_j S_{ij}(p_i^*, p_{-i}^*)\left[1 - S_{ij}(p_i^*, p_{-i}^*)\right]\right]}$$

where

$$R_{ij}(p_i^*) = \frac{\exp\left(\frac{a_{ij} - \theta_j p_i^*}{\mu}\right)}{1 + \exp\left(\frac{a_{ij} - \theta_j p_i^*}{\mu}\right)},$$

$$S_{ij}(p_i^*, p_{-i}^*) = \frac{\exp\left(\frac{a_{ij} - \theta_j p_i^* - c_j}{\mu}\right)}{1 + \sum_{h=1}^{n} \exp\left(\frac{a_{hj} - \theta_j p_h^* - c_j}{\mu}\right)},$$

mc is firm i's marginal cost, and f is the royalty fee paid to the platform.

Proof. Firm i's profit is given by

$$\pi_i = ((1-f)p_i - mc)\, d_i = (1-f)p_i d_i - mcd_i.$$

where

$$d_i = \tau \sum_{j \in \mathcal{J}_{it}} \gamma_j R_{ij}(p_i) + (1-\tau) \sum_{j=1}^{k} \gamma_j S_{ij}(p_i, p_{-i}).$$

The first order condition is

$$\begin{aligned} \frac{\partial \pi_i}{\partial p_i} = 0 &\iff (1-f)d_i + (1-f)p_i \frac{\partial d_i}{\partial p_i} - mc \frac{\partial d_i}{\partial p_i} = 0 \\ &\iff (1-f)d_i + [(1-f)p_i - mc] \frac{\partial d_i}{\partial p_i} = 0. \end{aligned}$$

The derivative of d_i with respect to p_i is

$$\begin{aligned} \frac{\partial d_i}{\partial p_i} &= \frac{\partial}{\partial p_i} \left(\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j R_{ij}(p_i) + (1-\tau) \sum_{j=1}^{k} \gamma_j S_{ij}(p_i, p_{-i}) \right) \\ &= \left(\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j \frac{\partial R_{ij}(p_i)}{\partial p_i} + (1-\tau) \sum_{j=1}^{k} \gamma_j \frac{\partial S_{ij}(p_i, p_{-i})}{\partial p_i} \right). \end{aligned}$$

I will first focus the the derivative of demand from the recommended consumers with respect to price.

$$\begin{aligned} \frac{\partial R_{ij}(p_i)}{\partial p_i} &= \frac{\partial}{\partial p_i} \left(\frac{\exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)}{1 + \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)} \right) \\ &= \frac{-\frac{\theta_j}{\mu} \left(1 + \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)\right) \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right) + \frac{\theta_j}{\mu} \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)^2}{\left(1 + \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)\right)^2} \\ &= \frac{-\frac{\theta_j}{\mu} \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right) - \frac{\theta_j}{\mu} \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)^2 + \frac{\theta_j}{\mu} \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)^2}{\left(1 + \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)\right)^2} \\ &= \frac{-\frac{\theta_j}{\mu} \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)}{\left(1 + \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)\right)^2} \\ &= -\frac{\theta_j}{\mu} \frac{\exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)}{\left(1 + \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)\right)} \frac{1}{\left(1 + \exp\left(\frac{a_{ij} - \theta_j p_i}{\mu}\right)\right)} \\ &= -\frac{\theta_j}{\mu} R_{ij}(p_i) \left[1 - R_{ij}(p_i)\right] \end{aligned}$$

Now, let us focus on the derivative of demand from all consumers in the market with respect to price.

$$
\begin{aligned}
\frac{\partial S_{ij}(p_i, p_{-i})}{\partial p_i} &= \frac{\partial}{\partial p_i}\left(\frac{\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)}{1+\sum_{h=1}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)}\right) \\
&= \frac{-\frac{\theta_j}{\mu}\left(1+\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)+\sum_{h\neq i}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)\right)\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)+\frac{\theta_j}{\mu}\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)^2}{\left(1+\sum_{h=1}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)\right)^2} \\
&= \frac{-\frac{\theta_j}{\mu}\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)-\frac{\theta_j}{\mu}\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)\sum_{h\neq i}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)}{\left(1+\sum_{h=1}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)\right)^2} \\
&= \frac{-\frac{\theta_j}{\mu}\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)-\frac{\theta_j}{\mu}\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)\sum_{h\neq i}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)}{\left(1+\exp\left(\frac{a_{ij}-\theta_j p_i c_j}{\mu}\right)+\sum_{h\neq i}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)\right)^2} \\
&= -\frac{\theta_j}{\mu}\frac{\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)\left[1+\sum_{h\neq i}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)\right]}{\left(1+\exp\left(\frac{a_{ij}-\theta_j p_i c_j}{\mu}\right)+\sum_{h\neq i}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)\right)^2} \\
&= -\frac{\theta_j}{\mu}\frac{\exp\left(\frac{a_{ij}-\theta_j p_i - c_j}{\mu}\right)}{\left(1+\exp\left(\frac{a_{ij}-\theta_j p_i c_j}{\mu}\right)+\sum_{h\neq i}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)\right)}\frac{1+\sum_{h\neq i}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)}{\left(1+\exp\left(\frac{a_{ij}-\theta_j p_i c_j}{\mu}\right)+\sum_{h\neq i}^{n}\exp\left(\frac{a_{hj}-\theta_j p_h - c_j}{\mu}\right)\right)} \\
&= -\frac{\theta_j}{\mu}S_{ij}(p_i, p_{-i})\left[1-S_{ij}(p_i, p_{-i})\right].
\end{aligned}
$$

Putting everything together, it follows that

$$
\begin{aligned}
\frac{\partial \pi_i}{\partial p_i} = 0 &\iff (1-f)d_i + [(1-f)p_i - mc]\frac{\partial d_i}{\partial p_i} = 0 \\
&\iff (1-f)d_i + [(1-f)p_i - mc]\left(\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j \frac{\partial R_{ij}(p_i)}{\partial p_i} + (1-\tau)\sum_{j=1}^{k} \gamma_j \frac{\partial S_{ij}(p_i, p_{-i})}{\partial p_i}\right) = 0 \\
&\iff (1-f)d_i - \frac{1}{\mu}[(1-f)p_i - mc]\left(\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j \theta_j R_{ij}(p_i)\left[1 - R_{ij}(p_i)\right] + (1-\tau)\sum_{j=1}^{k} \gamma_j \theta_j S_{ij}(p_i, p_{-i})\left[1 - S_{ij}(p_i, p_{-i})\right]\right) = 0 \\
&\iff (1-f)p_i - mc = \frac{(1-f)d_i}{\frac{1}{\mu}\left(\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j \theta_j R_{ij}(p_i)\left[1 - R_{ij}(p_i)\right] + (1-\tau)\sum_{j=1}^{k} \gamma_j \theta_j S_{ij}(p_i, p_{-i})\left[1 - S_{ij}(p_i, p_{-i})\right]\right)} \\
&\iff p_i = \frac{mc}{1-f} + \frac{\mu d_i}{\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j \theta_j R_{ij}(p_i)\left[1 - R_{ij}(p_i)\right] + (1-\tau)\sum_{j=1}^{k} \gamma_j \theta_j S_{ij}(p_i, p_{-i})\left[1 - S_{ij}(p_i, p_{-i})\right]} \\
&\iff p_i^* = \frac{mc}{1-f} + \frac{\mu\left(\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j R_{ij}(p_i^*) + (1-\tau)\sum_{j=1}^{k} \gamma_j S_{ij}(p_i^*, p_{-i}^*)\right)}{\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j \theta_j R_{ij}(p_i^*)\left[1 - R_{ij}(p_i^*)\right] + (1-\tau)\sum_{j=1}^{k} \gamma_j \theta_j S_{ij}(p_i^*, p_{-i}^*)\left[1 - S_{ij}(p_i^*, p_{-i}^*)\right]}
\end{aligned}
$$

□

F..2 *Joint-Collusive*

Theorem 2. *Assume that the set of consumers firm i is recommended to is $\mathcal{J}_{it} \subseteq \{1, 2, \ldots, k\}$. The joint-collusive equilibrium price for firm i is given by*

$$
p_i^* = \frac{mc}{1-f} + \frac{\mu\left(\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j R_{ij}(p_i^*) + (1-\tau)\sum_{j=1}^{k} \gamma_j S_{ij}(p_i^*, p_{-i}^*)\right)}{\left[\tau \sum_{j \in \mathcal{J}_{it}} \gamma_j \theta_j R_{ij}(p_i^*)\left[1 - R_{ij}(p_i^*)\right] + (1-\tau)\sum_{j=1}^{k} \gamma_j \theta_j S_{ij}(p_i^*, p_{-i}^*)\left[1 - nS_{ij}(p_i^*, p_{-i}^*)\right]\right]}
$$

where

$$
\begin{aligned}
R_{ij}(p_i^*) &= \frac{\exp\left(\frac{a_{ij} - \theta_j p_i^*}{\mu}\right)}{1 + \exp\left(\frac{a_{ij} - \theta_j p_i^*}{\mu}\right)}, \\
S_{ij}(p_i^*, p_{-i}^*) &= \frac{\exp\left(\frac{a_{ij} - \theta_j p_i^* - c_j}{\mu}\right)}{1 + \sum_{h=1}^{n} \exp\left(\frac{a_{hj} - \theta_j p_h^* - c_j}{\mu}\right)},
\end{aligned}
$$

mc is firm i's marginal cost, and f is the royalty fee paid to the platform.

Proof. Proof follows similarly to the proof of Theorem 1. □