Platform Behavior under Market Shocks: A Simulation Framework and Reinforcement-Learning Based Study

Xintong Wang
Harvard University
Cambridge, MA, USA
xintongw@seas.harvard.edu

Gary Qiurui Ma
Harvard University
Cambridge, MA, USA
qiurui_ma@g.harvard.edu

Alon Eden
The Hebrew University
Jerusalem, Israel
alon.eden@mail.huji.ac.il

Clara Li
Harvard University
Cambridge, MA, USA
clarali@college.harvard.edu

Alexander Trott
Salesforce Research
Palo Alto, CA, USA
alex.trott16@gmail.com

Stephan Zheng
Salesforce Research
Palo Alto, CA, USA
st.t.zheng@gmail.com

David C. Parkes
Harvard University
Cambridge, MA, USA
parkes@eecs.harvard.edu

ABSTRACT

We study the behavior of an economic platform (e.g., Amazon, Uber Eats, Instacart) under shocks, such as COVID-19 lockdowns, and the effect of different regulation considerations. To this end, we develop a multi-agent simulation environment of a platform economy in a multi-period setting where shocks may occur and disrupt the economy. Buyers and sellers are heterogeneous and modeled as economically-motivated agents, choosing whether or not to pay fees to access the platform. We use *deep reinforcement learning* to model the fee-setting and matching behavior of the platform, and consider two major types of regulation frameworks: (1) taxation policies and (2) platform fee restrictions. We offer a number of simulated experiments that cover different market settings and shed light on regulatory tradeoffs. Our results show that while many interventions are ineffective with a sophisticated platform actor, we identify a particular kind of regulation—fixing fees to the optimal, no-shock fees while still allowing a platform to choose how to match buyers and sellers—as holding promise for promoting the efficiency and resilience of the economic system.

KEYWORDS

Platform economy, market shock, fee setting, matching, multi-agent simulation, reinforcement learning, agent-based modeling

ACM Reference Format:
Xintong Wang, Gary Qiurui Ma, Alon Eden, Clara Li, Alexander Trott, Stephan Zheng, and David C. Parkes. 2023. Platform Behavior under Market Shocks: A Simulation Framework and Reinforcement-Learning Based Study. In *Proceedings of the ACM Web Conference 2023 (WWW '23), May 1–5, 2023, Austin, TX, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3543507.3583523

1 INTRODUCTION

Market-driven platforms, such as Amazon, DoorDash, Uber, and TaskRabbit, play an increasingly important role in today's economy, bringing together parties to facilitate trade and presenting new ways to create value. First, they reduce *search cost* by introducing potential matches that were not known before, and second, they reduce *fulfillment cost* by taking care of service or product delivery, thus reducing the effort made to complete transactions.

The importance of the platform-based economy became even more apparent during the COVID-19 pandemic, especially for the restaurant industry. Stay-at-home orders, together with the closure of dine-in channels and caution in regard to visiting brick-and-mortar businesses increased the fulfillment cost of consumers transacting in physical locations. This led to an increasing number of users and restaurants to adopt food-delivery platforms. One study of Uber Eats from February through May 2020 showed a surge in both demand and supply after the shelter-in-place guidance was issued in the U.S. [23]. At the same time, this new prominence gave platforms increased market power. As a demonstration, restaurant commission fees are set by some platforms to 30% per order, leaving traditional restaurants, many of whom no longer have dine-in revenue, with low or even negative margins [1]. In December 2020, the National Restaurant Association reported that more than 110,000 U.S. restaurants—one in six—have permanently closed down since the start of pandemic [3]. To support restaurants, states such as New

This research is funded in part by Defense Advanced Research Projects Agency under Cooperative Agreement HR00111920029. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred. This is approved for public release; distribution is unlimited. Work done while Alon Eden was at Harvard University and Alexander Trott and Stephan Zheng were at Salesforce Research. David Parkes is also currently a Senior Research Scientist with DeepMind. Our code is available at https://github.com/XintongEmilyWang/platform_foundation. Additional experimental results can be found in supplemental material that is published along with the paper.

WWW '23, May 1–5, 2023, Austin, TX, USA
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-9416-1/23/04…$15.00
https://doi.org/10.1145/3543507.3583523

York and California have imposed commission caps and yet platforms responded with countermeasures: the day after Jersey City enforced a 10% cap on fees, Uber Eats added a $3 delivery fee to customers and reduced the delivery radius for restaurants.[1] This speaks to the complexity of the ecosystem, and regulatory policies on delivery platforms have been continuously proposed and debated.

In this work, we use the methods of AI and multi-agent simulation to study a platform-based economy under market shocks, with this as a first step towards reproducing phenomena observed in the real-world economy and as a tool for conducting counterfactual analysis to understand platform behavior in response to different regulations. We develop a multi-agent Gym environment to capture major aspects of the economy in a multi-period setting, with key modeling choices (e.g., epoch-based decision making, user behavior inertia, the discrete-logit choice model) based on the economic literature. Our model further captures a full cycle of market shock, designed to represent the pre-, during, and post-crisis periods.

We formulate the platform's problem as a *partially observable Markov decision process* (POMDP), with both commonly observable components (e.g., shock intensity) and private elements (e.g., buyers' knowledge about sellers, off-platform transactions). We model the platform as a rational agent that uses reinforcement learning (RL) to set fees and match buyer queries to sellers. Buyers and sellers decide whether to join the platform, considering both fees and search and fulfillment costs; on-platform buyers further decide whether to transact with a matched platform seller or off platform.

We conduct extensive simulations to explore a range of settings that differ in market structures (i.e., locations of buyers and sellers in product and preference space), knowledge levels of buyers about sellers, and the cost of off-platform fulfillment. The goal is to use RL to model the optimal behavior of a platform under different regulations, and study the effect of a platform on the efficiency and resilience of the overall economic system. In the absence of any regulation, we find that a revenue-maximizing platform, even while helping to facilitate trades, tends to leverage its increased market power during the shock to raise fees and extract surplus from buyers and sellers. This leads to seller shutdowns and lower post-shock economic welfare, compared with that of the pre-shock period.

As a first kind of regulation, we consider a class of *taxation policies* that impose different tax rates on different categories of profit (e.g., revenue made from user subscriptions, or revenue made from transaction referrals), and study the platform response. We further consider regulations that *cap platform fees* in a particular way. Our results show that either taxation, or capping some subset of fees, simply leads the platform to transfer loss to users by adjusting other fees, demonstrating the power of making use of RL to model the rational behavior of a platform. On the other hand, capping *all* fees and not just a subset has a moderately positive effect on protecting sellers from bankruptcy and promoting a resilient ecosystem. In practice, however, this intervention requires a regulator to have detailed knowledge in setting these caps (and also assumes the platform will follow a particular, myopic matching policy).

The second part of our study allows the platform to retain full flexibility in choosing how to match buyers and sellers while restricting it to keep the same fee structure that it chooses in the absence of shocks. Thus, it requires no special knowledge on the part of a regulator, and gives the platform full flexibility in regard to behavior (matching) that is proprietary and not easy to regulate. We show that under this intervention, a revenue-maximizing platform learns to use matching to retain a more diverse set of sellers on platform, so as to increase its long-term revenue from user registrations. This also helps to promote the efficiency, resilience, and seller diversity in the overall economic system.

The present framework, which introduces a multi-agent Gym environment and uses RL to derive optimal platform responses, provides a tool for understanding regulations and platform economies. Our hope is that this can complement economic theory, which can become analytically intractable in complex, dynamic environments, as well as pure data-driven approaches that cannot answer questions about changing market and agent behaviors. We return to a discussion of the opportunities and outstanding challenges with this kind of AI-based approach to economic study at the end of the paper.

1.1 Related Work

Platform Models. An extensive economic literature focuses on how to establish network effects through fee-setting or subsidizing one side of the market under various forms of platform competition (e.g., single- vs. multi-homing) [2, 5, 24]. To facilitate equilibrium analysis, these models often require simplified assumptions for tractability, e.g., restricting to a single round of platform fee-setting and agent subscribing, adopting a fixed platform matching policy, and assuming homogeneous non-platform actors. Besides related literature that characterizes platform behavior, there are also works on complexity results in regard to setting optimal fees (even for a market with one buyer and two periods) [22] and optimally matching buy queries to sellers (even for a single period) [19].

In contrast, the literature on the role and behavior of economic platforms under shocks is fairly limited. Empirical studies have been conducted to study the impact of COVID-19 on the demand of food-delivery platforms [23], the effect of fee controls on on-demand services [17, 28], as well as the extent to which pandemic has persistently changed customers' purchasing behavior even after the shock calms down, due to habit formation around delivery [21].

RL for Economic System Design. Recent work has demonstrated the effectiveness of using RL for the design and understanding of complex economic systems, including dynamically setting reserve prices in auctions [26], selling user impressions to advertisers [30], designing tax policies [33], optimizing user satisfaction for recommender systems [6, 32], and designing sequential price mechanisms [4]. Many of these works rely on agent-based simulation to model the economic system and conduct counterfactual analysis through the dynamic interactions of agents. For example, Zheng et al. [33] use RL to model a social planner who designs income taxes in multi-period, simulated spatial economies. We are not aware of previous work on the modeling and study of market-based platforms. Most similar to ours is the work of Zhan et al. [32], built on the *RecSim environment* [15], which uses RL to optimize the long-term social welfare of users and content providers in a dynamic, recommender system. Besides the presence of economic shocks, our setting differs in the use of platform fees, the existence of an alternate sales channel (i.e., off platform), and the implication that agents can choose to join or quit the platform.

[1] https://www.protocol.com/delivery-commission-caps-uber-eats-grubhub

2 A MULTI-AGENT PLATFORM MODEL

2.1 Market Environment and Agent Dynamics

The market is populated with heterogeneous buyers $\mathcal{B}$, heterogeneous sellers $\mathcal{S}$, and a single platform p. Following embedding-based representations in recommender systems [25], we adopt a common *latent space* to represent each buyer or seller, $v_b, v_s \in \mathcal{V} \subseteq [0,1]^2$. The first dimension of an agent's *location*, denoted v^0, describes product features (e.g., in the case of food, Italian, Japanese; spicy or not) and the second dimension, denoted v^1, the (normalized) price level (e.g., \$ . . . \$\$\$\$). A seller s sells food at a price v_s^1, of which an ω_s fraction is the production cost. Each buyer b knows a subset of sellers $\mathcal{S}_b \subseteq \mathcal{S}$, and may transact with $s \in \mathcal{S}_b$ without using a platform. Buyers who use the platform are also introduced to additional sellers on platform.

We formulate an *epoch-based* decision problem for agents, similar to Mladenov et al. [19]. There are multiple epochs, indexed k, within an episode. Each epoch has a fixed length of T time steps (e.g., a month of 30 days). We consider a *world transaction friction*, denoted $\mu_k > 0$, which varies by epoch and represents the cost of buyers completing a transaction off-platform (see Eq. (1)). We model *shocks* corresponding to changes in this friction; e.g., during a pandemic, transaction frictions for food-service industry were extremely high, due to fears of sharing indoor spaces and absence of dine-in options.

At the start of an epoch k**.** The platform sets fees, including the buyer and seller *subscription fees*, denoted $P_{\mathcal{B},k} \geq 0$ and $P_{\mathcal{S},k} \geq 0$ respectively, and a per-transaction seller *referral rate* $P_{R,k} \in [0,1]$, the fraction of prices as transaction fee paid by the seller to the platform. We discuss the platform's fee-setting policy in Section 3.1.

Buyers and sellers observe platform fees and the world transaction friction μ_k, and decide whether to pay the subscription fee to use the platform for epoch k. We denote the sets of subscribed buyers and sellers in epoch k as $\mathcal{B}_k$ and $\mathcal{S}_k$. We assume that the platform knows the locations of buyer queries and on-platform sellers in the latent space. This reflects that platforms tend to have good data on the market-relevant properties of sellers and that the combination of a search interface and historical buyer information gives good information on the current demand context of a buyer. Each buyer has a *per-epoch budget*, $\psi_b > 0$, linearly proportional to the buyer's price preference v_b^1. This controls the number of transactions that a buyer will undertake in a given epoch. Beyond fees, the platform matches queries from platform buyers to platform sellers, and in selective treatments, with a matching policy learned through RL (see Section 3.2).

For each time step t ***within an epoch*** k**.** We follow the sequence of "query, match, and transact":
(1) *Query*: a buyer $b \in \mathcal{B}$ is randomly selected to issue a query according to their taste and price preferences (e.g., \$\$\$\$ sushi or \$\$ pizza), $q_{b,t} \sim \mathcal{N}(v_b, \sigma_b^2)$, where σ_b^2 specifies the query variance of b.
(2) *Match*: only for an on-platform buyer b, the platform observes $q_{b,t}$ and matches it to an on-platform seller, denoted $s_{p,t} \in \mathcal{S}_k$.
(3) *Transact*: the buyer b can pick a seller $s \in \{s_{p,t}\} \bigcup \mathcal{S}_b$ if b is on-platform, and $s \in \mathcal{S}_b$ otherwise. A buyer may also choose not to transact if the matching surplus is negative (details in Section 2.2).

We refer to a transaction that is matched via the platform as *a platform transaction*, and otherwise as a *a world transaction.* Even an on-platform buyer can choose a world transaction if this is better than that recommended by the platform. For each world transaction, the buyer suffers a fulfillment cost of μ_k, whereas for each platform transaction, the seller pays a *referral fee*, which is a fraction $P_{R,k}$ of the seller's price.

At the end of an epoch k**.** Buyers and sellers evaluate their surplus from transactions and fees paid to guide future subscription decisions (see Section 2.4). The platform evaluates revenue made through subscription and referral fees to inform adjustments to fees or its matching policy (see Section 3). Each seller has a *shutdown threshold*, $\lambda_s \in \mathbb{N}_{>0}$, and will go bankrupt if they do not obtain positive surplus for a consecutive λ_s epochs. Once shutdown, a seller is unable to engage in transactions for future epochs.

2.2 Transaction-Level Decisions

A buyer with query q and transacting with seller s receives a *matching utility*, $u_{\mathcal{B}}(q,s)$, reflecting matching quality. A buyer with a choice of sellers prefers the one that maximizes immediate *matching surplus*, defined as matching utility minus any transaction friction.

Choice in the world. A buyer b with query $q_{b,t}$ can choose from known sellers whose prices (denoted by v_s^1 for seller s) are within their epoch-budget left at t, denoted $\psi_{b,t}$, i.e., $\mathcal{S}_{b,t} := \{s \in \mathcal{S}_b \ : \ v_s^1 \leq \psi_{b,t}\}$. For $\mathcal{S}_{b,t} \neq \emptyset$, the best world choice is $s_w^* := \mathrm{argmax}_{s \in \mathcal{S}_{b,t}} u_{\mathcal{B}}(q_{b,t}, s)$. Since the friction μ_k could be high enough to prevent a buyer from transacting, we let $s_{b,t}^w$ be s_w^* if $u_{\mathcal{B}}(q_{b,t}, s_w^*) > \mu_k$, and ϕ otherwise, to denote no preferred world seller. Writing $u_{\mathcal{B}}(q_{b,t}, \phi) = 0$, the *world surplus* to buyer b at time t is

$$u_{b,t}^w = \max(u_{\mathcal{B}}(q_{b,t}, s_{b,t}^w) - \mu_k, 0). \tag{1}$$

Choice on the platform. For an on-platform buyer, the *platform surplus*, $u_{b,t}^p$, at time t is $u_{b,t}^p = u_{\mathcal{B}}(q_{b,t}, s_{p,t})$ in the case the platform-recommended seller, $s_{p,t}$, is within remaining budget $\psi_{b,t}$, and zero otherwise. We set $u_{b,t}^p = 0$ if b is off platform.

Overall choice. If no seller provides a positive surplus, the buyer will choose not to transact. Otherwise, a buyer b chooses $s_{b,t}^w$ if it is off-platform, and the most preferred of $s_{b,t}^w$ and $s_{p,t}$ when it is on-platform.[2] We write $s_{b,t}$ to denote the choice of the buyer at time t, and denote a buyer's query, seller options, and transaction as 4-tuple $(q_{b,t}, s_{b,t}^w, s_{p,t}, s_{b,t})$, where $s_{p,t}$ is ϕ for an off-platform buyer.

Surplus from a transaction. The buyer surplus is $r_{b,t} = \max\{u_{b,t}^w, u_{b,t}^p\}$, and zero if the buyer does not transact. We define the *world matching surplus* and *platform matching surplus* respectively as $r_{b,t}^w = u_{b,t}^w \cdot \mathcal{I}_{b,t}^w$ and $r_{b,t}^p = u_{b,t}^p \cdot (1 - \mathcal{I}_{b,t}^w)$, where $\mathcal{I}_{b,t}^w$ is an indicator of whether buyer b transacted in the world or not at time t. A seller when chosen by a buyer cannot decline a transaction. The surplus of seller s (i.e., net profit) is

$$r_{s,t} = \begin{cases} v_s^1(1 - \omega_s - P_{R,k}) & \text{for a platform transaction,} \\ v_s^1(1 - \omega_s) & \text{for a world transaction,} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

We denote $n_{s,k}^p$ the number of transactions completed by seller s via the platform during epoch k, and $n_{s,k}^w$ the number of transactions completed by the seller in the world.

[2] Sellers $s_{p,t}$ and $s_{b,t}^w$ may be the same seller, in which case the buyer will choose to transact via the platform since the world transaction friction is always positive.

2.3 Epoch Surplus and Platform Revenue

Buyer b's *epoch surplus*, $r_{b,k}$, is their total surplus from matching minus any subscription fee paid,

$$r_{b,k} = \sum_{t \in k} r_{b,t} - \mathcal{I}^p_{b,k} P_{\mathcal{B},k} = \sum_{t \in k} r^w_{b,t} + \sum_{t \in k} r^p_{b,t} - \mathcal{I}^p_{b,k} P_{\mathcal{B},k}, \tag{3}$$

where $\mathcal{I}^p_{b,k} \in \{0, 1\}$ indicates whether the the buyer is off- or on-platform during epoch k. Similarly, a seller s's *epoch surplus*, $r_{s,k}$, is their total net profit from transactions minus any fee paid,

$$r_{s,k} = \sum_{t \in k} r_{s,t} - \mathcal{I}^p_{s,k} P_{\mathcal{S},k} = n^w_{s,k} v^1_s (1 - \omega_s) + n^p_{s,k} v^1_s (1 - \omega_s - P_{R,k}) - \mathcal{I}^p_{s,k} P_{\mathcal{S},k}, \tag{4}$$

where $\mathcal{I}^p_{s,k} \in \{0, 1\}$ is an indicator to denote whether the the seller is off- or on-platform. The *total platform revenue* in epoch k is simply the sum of the subscription and referral fee it charges,

$$r_{p,k} = \sum_{b \in \mathcal{B}} \mathcal{I}^p_{b,k} P_{\mathcal{B},k} + \sum_{s \in \mathcal{S}} \left(\mathcal{I}^p_{s,k} P_{\mathcal{S},k} + n^p_{s,k} v^1_s P_{R,k} \right). \tag{5}$$

The *total welfare* of the economy in epoch k is the sum of all buyer and seller surplus and platform revenue.

2.4 Subscription-Level Decisions

At the start of each epoch, each buyer or seller "wakes up" with some probability, reevaluates their current state, and decides whether or not to subscribe to the platform. We provide in the sequel a high-level description as to how buyers and sellers make such decisions, and defer detailed analysis to Appendix A.

Estimating the effect of a subscription decision. Each agent, whether buyer or seller, subscribes by comparing the estimated surplus on and off platform under the newly proposed platform fees and the observed world transaction friction. This is done assuming a unilateral change, i.e., they are the only one who wakes up and makes a different subscription decision, and queries from the agent itself as well as others remain the same as in the previous epoch. For buyers or sellers who were on platform in the past epoch, this estimate means to re-evaluate their transaction decisions under new fees and friction. For agents who were off platform in the past epoch, we assume the platform can provide information (honestly) to facilitate this estimation. This could occur, for example, through a trial period on the platform or by providing an estimate of costs and benefits based on recent history.

Agent-specific decision inertia. Our decision model incorporates *behavior inertia*, capturing an agent's tendency to stick with their current decision (e.g., subscribing to a platform). This kind of inertia has been empirically observed in platform adoption decisions post-pandemic [21], along with other settings, including choosing consumer packaged goods [9, 27] and health and automobile insurance [11, 14]. Following prior models [8, 10, 18], we incorporate inertia as an additive term to an agent's surplus from the current decision. Specifically, we model inertia logarithmically in the number of epochs for which an agent has committed to the same choice, and "reset" this inertia upon a change to the subscription decision. Based on this adjusted surplus, both buyer and seller agents decide whether to subscribe or not according to probabilities calculated from the standard *discrete-choice logit model* [8, 18]. See Appendix A for details.

3 THE PLATFORM'S DECISION MODEL

We formulate the platform's problem as a POMDP [16], with buyers' knowledge about sellers and transactions in the world as private information, and thus not observable to the platform. The platform learns a fee-setting and matching policy based on observations of the decisions of on-platform buyers and sellers. In this work, for reasons of computational tractability, we study platforms that either follow a default, myopic matching policy and learn to set fees or follow fixed, regulated fees and learn how to match.

3.1 Learning to Set Fees under Myopic Matching

We study the sequential decision-making problem of a platform that learns to set fees while using *myopic query matching*—recommending the closest on-platform seller to a query in the latent space, and thus the seller that yields the highest utility to the buyer. Here, the platform decides fees for epoch k based on its experience from epoch $k - 1$. We describe the fee-setting POMDP model:

- The *state* $x_k \in \mathcal{X}$ at the start of epoch k is comprised of
  (1) buyer attributes: the latent location, epoch budget, query distribution, and knowledge of world sellers,
  (2) seller attributes: the latent location, cost fraction, and shutdown threshold,
  (3) agent subscription states: either on- or off-platform for the past epoch, $\mathcal{I}^p_{\mathcal{B},k-1}$ and $\mathcal{I}^p_{\mathcal{S},k-1}$,
  (4) the agent inertia levels: $\chi_{b,k-1}$ and $\chi_{s,k-1}$,
  (5) a sequence of query, seller candidates, and buyer's choices of previous epoch: $Q_{k-1} = \{(q_{b,t}, s^p_{b,t}, s^w_{b,t}, s_{b,t})\}_{t \in k-1, b \in \mathcal{B}}$,
  (6) the shutdown states for sellers: whether a seller has shut down at the end of epoch $k - 1$, denoted $\mathcal{I}_{s,k-1}$,
  (7) the platform fees for the past epoch: $P_{\mathcal{B},k-1}, P_{\mathcal{S},k-1}, P_{R,k-1}$,
  (8) the world transaction friction for the current epoch: μ_k.
- An *action* $a_k = (P_{\mathcal{B},k}, P_{\mathcal{S},k}, P_{R,k})$ defines the fees for the upcoming epoch k. We model a discrete action space $\mathcal{A}$ where fees take discrete values at integer multiples of a tick (or percentage) size.
- For the *state transition* $\mathcal{P} : \mathcal{X} \times \mathcal{A} \to \Delta(\mathcal{X})$, buyers and (viable) sellers follow their choice model to subscribe to the platform (Section 2.4), leading to new subscription states and inertia levels. For each time step $t \in k$, we follow the "query, match, transact" dynamics, which gives a full sequence Q_k. Each viable seller may shut down based on the surplus in epoch k and their shutdown threshold. Fees follow from the actions taken, and the world transaction friction evolves according to a lognormal process. Altogether, this gives a new state $x_{k+1} \sim \mathcal{P}(x_k, a_k)$.
- A *reward* $r_k \sim \mathcal{R}(x_k, a_k)$ is provided to the platform at the end of epoch k, when agent subscription and transaction outcomes are available. The reward can be set to model different platform objectives, integrating considerations that come from regulation.
- The platform's *observation* $o_{k+1} \in \Omega$ consists of the sequence of queries generated by on-platform buyers, as well as their decisions on whether or not to transact via the platform, i.e., $Q^p_k := \{(q_{b,t}, s^p_{b,t}, \mathbb{I}\{s_{b,t} = s^p_{b,t}\})\}_{t \in k, b \in \mathcal{B}_k}$ (but not counterparties in off-platform transactions).

3.2 Learning to Match under Fixed Fees

In a second setting, we study the sequential decision-making problem of a platform that learns how to match buyer queries to sellers when the fees are fixed, for example, due to regulation. Myopic query matching, as described above, favors the buyer side of the market, by directing a query to the buyer's utility-maximizing, on-platform seller. To complement this, we model matching strategies that can choose to benefit other parties in the economy, i.e., sellers or the platform itself. For interpretability, we define a *matching strategy* by two parameters: (1) a *matching utility threshold* $\eta \in [0, 1]$, that specifies the minimum utility that a recommended seller should provide to the buyer, as a fraction of utility from the myopically-optimal match, and (2) a *matching rule*, which directs how to pick a seller amongst those that meet this utility threshold. We consider two rules:

- *The seller-aware matching rule*: Among sellers who meet the utility threshold, match a query to the seller who has the lowest surplus on the platform so far during the epoch,[3] breaking ties in the buyer's favor.
- *The profit-driven matching rule*: Among sellers who meet the utility threshold, match a query to the seller who brings the largest revenue to the platform, breaking ties in the buyer's favor.

As a special case, myopic query matching corresponds to setting the matching threshold $\eta = 1$. Intuitively, the seller-aware rule may be useful in promoting a more diverse set of sellers, by increasing sales to those who have been benefiting less from the platform, whereas the profit-driven rule is at the other end of the spectrum, aiming to maximize the platform's myopic transaction revenue. The goal of the platform is to learn a *matching policy* that chooses a matching strategy for an epoch—a utility threshold and a matching rule, based on an observation as to which buyers and sellers choose to subscribe for the upcoming epoch. To model this, we make several adjustments to the fee-setting POMDP in defining a *matching POMDP*, and defer these details for space reasons to Appendix B.

3.3 Finding the Optimal Policy

Interactions between the platform and buyers and sellers can be considered as a *Stackelberg game*: the platform agent is the *leader*, choosing the fee or matching policies, and buyers and sellers are the *followers*, responding to platform policies. In our simulation, the buyer and seller strategies are a fixed mapping from prior matching experience, platform fees, and world transaction friction to decisions in regard to joining or exiting the platform, and we can handle this Stackelberg structure by modeling the agents within the POMDP environment of the platform (as part of the transition model). Depending on the set-up, the platform learns a fee-setting policy or a matching policy, denoted $\pi(a|o_k)$, to maximize its discounted cumulative reward across different episodes:

$$\max_{\pi} \quad \mathbb{E}_{a\sim\pi, x\sim\mathcal{P}}\left[\sum_{k=0}^{K} \gamma^k r_k\right], \tag{6}$$

where $\gamma \in (0, 1)$ is the discount factor, $K = |\tau|$ is the total number of epochs in an episode, and r_k is based on Eq.(5) whose precise value can further depend on a regulatory structure. Following the success of deep learning to solve POMDPs [12, 13, 31], we use deep RL to learn $\pi(a|o_k; \boldsymbol{\theta})$, specifically parameters $\boldsymbol{\theta}$ that extract sufficient statistics from the observation history and map to actions that maximize the objective (more details in Section 4).

4 PLATFORM BEHAVIOR UNDER SHOCKS AND REGULATORY INTERVENTIONS

We study platform behavior and its effect on the simulated economy under the following regulations: (1) taxation policies that directly change the reward to the platform, (2) fee caps that are enacted through restricting the platform's action space of fee-setting, and (3) fee freezes that are studied along with a platform who can continue to change its matching policy. We first provide specifics on simulation configurations, and then present experimental results.

4.1 Experiment Settings

We follow Section 2 in specifying a set of different market environments, with 10 buyers and 10 sellers, an episode that consists of $K = 12$ epochs, and with each epoch containing $T = 100$ time steps.

Market structure and dynamics. We consider three types of *market structures*, corresponding to distinct latent locations of buyers and sellers (see Appendix C.1, Figure 5):

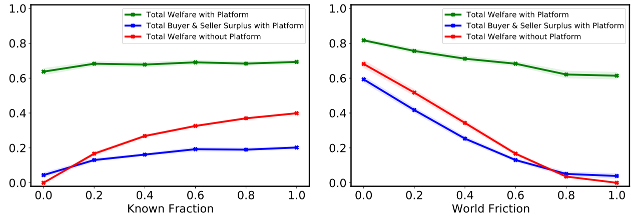
- *Uniform:* $v \sim U[0, 1]^2$, representing an economy with diverse buyer interests and seller attributes.
- *Core-and-Niche:* $v \sim$ truncated Gaussian$(\mu, \sigma^2, 0, 1)$ with $\mu = [0.5, 0.4]$ and $\sigma^2 = [0.2, 0; 0, 0.2]$, representing a "core" of agents around μ, as well as "niche" agents located away from the core.
- *Two-Core:* One group $v \sim$ truncated Gaussian$(\mu_1, \sigma_1^2, 0, 1)$ with $\mu_1 = [0.7, 0.3]$ and $\sigma_1^2 = [0.17, 0; 0, 0.17]$, and another group $v \sim$ truncated Gaussian$(\mu_2, \sigma_2^2, 0, 1)$ with $\mu_2 = [0.3, 0.7]$ and $\sigma_2^2 = [0.17, 0; 0, 0.17]$. The first group is centered around relatively cheap options, and the second around more expensive ones.

We present results of the Core-and-Niche market, and defer the other two market structures to supplemental material that is published along with the paper. For each structure, we also consider markets that vary in the buyer knowledge level, ρ. In particular, each buyer b samples i.i.d. *Bern*(ρ) for each seller, to generate its set of known sellers $\mathcal{S}_b$.

Agent attributes. Within an epoch, buyers arrive in a round robin fashion. A buyer b who arrives at t submits a query around the buyer's latent location, $q_{b,t} \sim \mathcal{N}(v_b, \sigma_b^2)$, with $\sigma_b^2 = [0.02, 0; 0, 0.02]$. When query q is fulfilled by seller s, the buyer receives a matching utility of $u_{\mathcal{B}}(q, s) = \exp(-c\|q - v_s\|_2)$, where we choose $c = 2$ to have matching utilities span $[0, 1]$. We set a buyer's *per-epoch budget* ψ_b to be v_b^1 (their price preferences) times the number of queries it submits within an epoch. Each seller's fractional cost is drawn from $\omega_s \sim U[0.2, 0.4]$. We set the shutdown threshold, $\lambda = 2$, for all sellers. Each agent has an initial preference of staying in the world or joining the platform, with an initial inertia level χ selected uniformly on $\{-2, -1, 0, 1, 2\}$ (intuitively, the more positive/negative the value reflects a stronger preference to stay on/off the platform).

Shocks. We vary the world transaction friction, μ_k, across epochs to model *pre-*, *during*, and *post-shock* stages, fixing the pre- and post-shock stages to each last for three epochs and with low world friction, $\mu_k = 0.1$. The shock stage is controlled by a *shock intensity*, $I \sim U[I_{\min}, I_{\max}]$, which specifies the largest value that will be attained. We sample $\mu_k \sim$ Lognormal$(\mu = 0, \sigma = 0.5)$, and multiply

[3] We assume for simplicity that the platform knows a seller's production cost, and thus can calculate its surplus from transactions. In practice, this idealized seller-aware rule could be modified and defined in terms of the number of platform transactions completed by a seller, or other inferred quantities about seller surplus.



(a) Varying buyers' knowledge level about sellers ρ, with $\mu = 0.6$. **(b) Varying the world transaction friction μ , with $\rho = 0.2$.**

Figure 1: Total welfare and buyer and seller surplus achieved in environments that vary in ρ and μ, with and without a platform under the Core-and-Niche market structure.

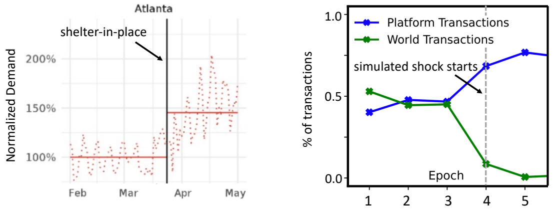


Figure 2: A comparison of empirically-observed demand surge after the shelter-in-place order as shown in [23] (Left) to the increase in the number of on-platform transactions induced by our simulated economic shock (Right).

the values by the intensity I. Figure 6 (red line) in Appendix C.1 shows the average shock schedule for $I \sim U[0.8, 1]$.

Platform action space. Without fee caps, the registration fees, $P_{\mathcal{B},k}$ and $P_{\mathcal{S},k}$, range from 0 to 10, with discrete levels at intervals of 0.2. The seller referral rate, $P_{R,k}$, ranges from 0 to 1, with discrete levels at intervals of 0.1. For matching, the matching utility threshold ranges from 0 to 1, and is discretized at intervals of 0.1.

Implementation details. We sample initial states with a *warm-up epoch*, used in addition to each of the twelve epochs, where the friction is 0.1, the platform charges no fees, and buyers and sellers join the platform based on their initial preferences. The experience gained by buyers and sellers in this warm-up epoch provides a basis for the platform to choose actions, and for the buyers and sellers to form estimates to guide their subscription decisions. We use the *Advantage Actor-Critic (A2C)* [7, 29] to learn the optimal platform policy, and present the average results from training for two different seeds and 100 test episodes. We defer descriptions of the neural network structure and hyperparameters to Appendix C.2.

4.2 Baseline Economic Performance

We first build intuition around our model. We study the value generated by a revenue-maximizing platform across a range of *single-epoch, no-shock* environments that vary in buyers' knowledge level about sellers, ρ, and the world transaction friction, μ. For each market structure, we generate three samples of latent locations of buyers and sellers, and for each latent sample and knowledge level, ρ, we sample ten knowledge structures, specifying which sellers are known by each buyer. We use *Bayesian Optimization* (BO) [20] to find platform fees that maximize the platform's revenue, and conduct controlled experiments with and without a platform.

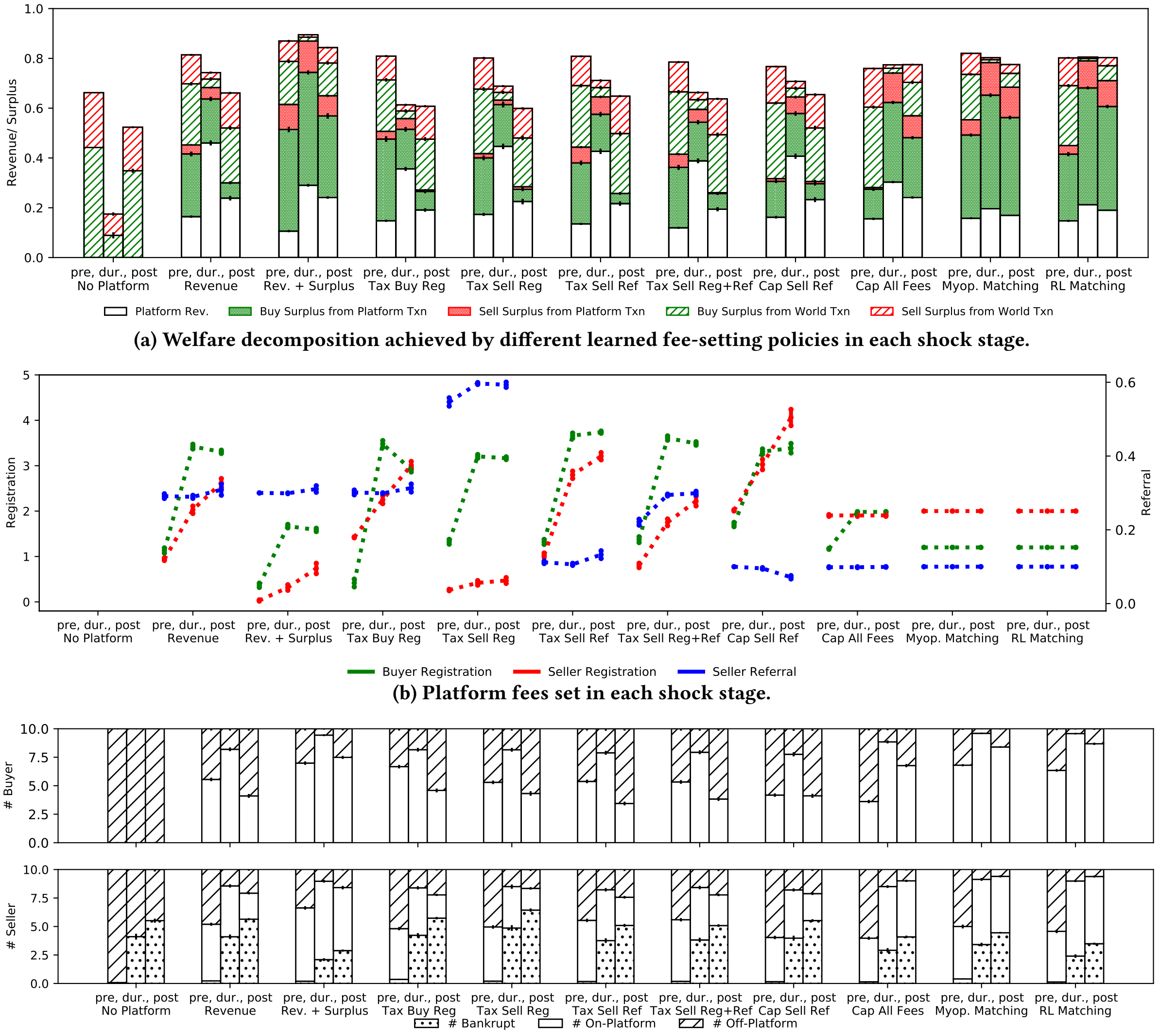
Figure 1 shows the buyer and seller surplus, as well as total welfare (agents' surplus and platform revenue), achieved in Core-and-Niche markets. We normalize surplus by the *ideal welfare* achieved in a setting where buyers knows all sellers and there is no world transaction friction. We validate the simulator by confirming that without a platform (Figure 1, red lines), the total welfare increases as buyers' knowledge about sellers increases and the world friction decreases. Across all environments, a revenue-maximizing platform consistently increases total welfare relative to the absence of a platform, creating value by reducing search costs (i.e., matching buyers to unknown sellers) and facilitating transactions (i.e., reducing off-platform fulfilment costs). We also see that the revenue a platform can extract (i.e., difference between green and blue lines) increases as buyers have less knowledge about sellers, and as the world transaction friction increases.

4.3 Platform Responses to Interventions

In this section, we present our main results, comparing no-intervention with three different kinds of interventions in the presence of market shocks. The headline result is that, whereas taxes and caps imposed on some subset of fees are largely ineffective, there are two forms of interventions under which a revenue-maximizing platform will promote seller diversity and the efficiency and resilience of the economy. These interventions either cap all fees (leaving flexibility to set fees subject to caps), or fix fees to those that a self-interested platform chooses in an environment without shocks but allow the platform to continue to optimize its matching policy. The second intervention may be especially relevant to practice: the knowledge of how to set fees comes from the platform's own behavior, and the platform can continue to flexibly make matching decisions.

Case 1: Platform fee setting in a laissez-faire system. We first consider a platform that is free from any form of regulation and learns to set fees to maximize revenue, while using myopic query matching. In Figure 2, we first illustrate the similarity between the simulated increase in the number of platform transactions and the empirically-observed demand surge in Atlanta after the 2020 stay-in-place order, which is representative of other U.S. cities (i.e., on-platform transactions, or demand, increasing by around 50% due to the shock). This gives a first, basic validation of the economic dynamics in our model.

Figure 3, columns 1-2, compare the outcomes achieved in markets with and without the platform. As a first observation, from Figure 3a, the platform generally improves the overall economic welfare, considering the sum of the revenue to the platform and the surplus to buyers and sellers (i.e., the total heights of the bars). This is especially salient during the shock stage (i.e., central bars), when the world friction is high and very few transactions can generate surplus in the absence of a platform. At the same time, we see two less beneficial outcomes. First, by comparing pre- and post-shock periods (i.e., the left and right bars of the same column), we find that the overall economic welfare falls after the shock, both with and without a platform. This arises as a result of sellers going bankrupt, so that buyers can then only be matched with less-preferred sellers. As further verified in Figure 3c, we see that the number of bankrupt sellers in markets with a platform is almost the same as that in markets without a platform. By comparing the surplus to buyers and sellers with and without the platform, we also find that



(a) Welfare decomposition achieved by different learned fee-setting policies in each shock stage.

(b) Platform fees set in each shock stage.

(c) The number of on- or off-platform agents and bankrupt sellers in each shock stage.

Figure 3: Welfare decomposition, platform fees, and agent states induced by different regulations (Core-and-Niche markets).

the platform reduces the total surplus to buyers and sellers in the post-shock stage relative to without the platform.

We further characterize the type of seller that is most likely to go bankrupt, classifying sellers into three groups: *core sellers* within one standard deviation of the center and with at least two buyers nearby, *niche sellers* that are beyond two standard deviations from the center with at most one buyer nearby, and *cheap sellers* with prices in the lower quartile. As shown in Table 1, cheap sellers are much more likely to go bankrupt in the presence of a platform, whereas niche sellers are more likely to go bankrupt in the absence of a platform. A rational platform learns to raise fees as much as possible during the shock (Figure 3b), leaving sellers with lower margins unable to afford the fees and facing bankruptcy.

Idealized baseline. As an exemplar on the possible effectiveness of regulations, we consider the effect of a *surplus-aware platform* that sets fees to optimize some combination of its own revenue and the on-platform user (buyer and seller) surplus, i.e., $r_{p,k} + \alpha(r^{p}_{b,k} + r^{p}_{s,k})$, and here we choose $\alpha = 0.4$. As shown in Figure 3a (column 3), this platform would restore a comparable level of overall economic welfare after the shock, and lead to fewer bankrupt sellers.

We next consider three possible regulatory interventions.

Case 2: platform fee setting under taxation policies. We study the introduction of taxation on platform profits made from different categories of fees, with tax schemes that charge a 20% tax on profits made from buyer registration fees, seller registration fees, seller referral fees, and all fees charged on sellers, respectively. In experiments, we tried a range of tax rates (i.e., 20%, 40%, 60%). Despite differences in the absolute fee values, they lead to qualitatively similar trends. As shown in Figure 3, columns 4-7, taxation policies in general lead to similar outcomes as those observed in the laissez-faire system (column 2) and the overall economic welfare decreases after the shock due to bankrupt sellers. Specifically, charging taxes on one kind of fee leads a platform to increase other fees, simply transferring the loss to another user group. For example,

Bankrupt freq.	No platform	Rev.-max.	Surplus-aware
All sellers	0.55 (0.02)	0.57 (0.01)	0.29 (0.01)
Core	0.33 (0.03)	0.46 (0.02)	0.28 (0.02)
Niche	0.88 (0.08)	0.45 (0.03)	0.17 (0.02)
Cheap	0.52 (0.04)	0.81 (0.03)	0.43 (0.03)

Table 1: Seller bankrupt frequencies in markets with no platform, a revenue-maximizing platform, and a surplus-aware platform.

Figure 3b compares fees when taxes are imposed on seller referral profits (column 6) to those in a laissez-faire system (column 2). We see that the platform learns to decrease referral rates while raising registration fees during and after the shock, with the effect that on-platform sellers achieve higher surplus (Figure 3a, column 6, solid red) relative to laissez-faire but at the expense of lower on-platform buyer surplus (solid green).

Case 3: platform fee setting under fee caps. As a third case, we first introduce a 10% cap on just the referral fee (a type of intervention that has been adopted in several U.S. states). Similar to taxes imposed on referral profits, the platform responds by raising the registration fees (Figure 3b column 8). This leads more buyers and sellers to stay off platform, especially before the shock, and reduces the on-platform surplus. We also consider the case where all fees are capped, with $P_{\mathcal{B},k} \leq 2.0$, $P_{\mathcal{S},k} \leq 2.0$, and $P_{R,k} \leq 0.1$. As shown in Figure 3b column 9, under these caps, the platform chooses to set the maximum possible fees except for a lower pre-shock buyer registration fee. These caps are able to induce platform policies that benefit the economic system, where the overall welfare is not affected by the shock, with many viable sellers and good buyer matches remaining. At the same time, this intervention requires knowledge on the part of the regulator.

Case 4: platform matching under fixed fees. In this case, we consider a regulatory policy in which the platform is required to keep the same fee structure as it picks, optimally, in a world without an economic shock. In addition, we allow the platform to adapt its matching policy in response to shocks. This case is of interest because the intervention uses only knowledge available to a regulator, and does not assume a particular approach to matching by the platform. This is practically relevant, as matching is proprietary and often hard to regulate due to lack of transparency.

We follow Section 3.2, and consider a platform that uses RL to learn a matching policy but with the fixed fees that are chosen by a revenue-optimizing platform based on the use of BO under the no-shock setting ($P_{\mathcal{B},k} = 1.2$, $P_{\mathcal{S},k} = 2.0$, and $P_{R,k} = 0.1$). As shown in Figure 3, right column, even together with flexibility in regard to choice of matching, the presence of the platform has the effect of helping the economy to preserve a similar level of welfare post-shock as pre-shock, and results in fewer bankrupt sellers. We visualize the matching policy that is learned by the platform in Figure 4. To maximize revenue, the platform generally adopts the seller-aware rule, both before and during the shock, which matches buyers to sellers with lower on-platform surplus. This can be explained as follows: (1) the regulated fees motivate the platform to retain a larger number of sellers through matching in order to generate revenue from registration fees, and (2) the platform has more flexibility in matching as a result of the shock affecting world transactions, and thus can afford to compromise a bit on matching quality without losing buyer transactions. As the shock decays, the platform then learns to increase the matching utility threshold to provide buyers with better matches.

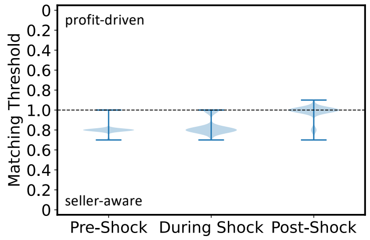


Figure 4: Probability density of learned matching strategy in the pre-, during, and post-shock stages.

	Myopic matching	RL matching
Welfare	973.12 (5.29)	989.44 (4.40)
Revenue	241.63 (3.06)	261.28 (1.59)
Seller shutdown freq.	0.44 (0.02)	0.34 (0.01)
Core	0.30 (0.01)	0.21 (0.01)
Niche	0.23 (0.02)	0.10 (0.02)
Cheap	0.86 (0.02)	0.78 (0.02)

Table 2: Welfare, platform revenue, and seller bankrupt frequencies of markets mediated by a myopic-matching platform and a RL-matching platform.

We further compare the use of RL matching with *myopic query matching* under the same set of fixed fees. As shown in Table 2, the RL-matching platform achieves both higher revenue and total welfare compared to myopic-matching (again, reflecting alignment with the interests of the regulator); the RL-matching platform also learns to substantially reduce the probability of cheap and niche sellers going bankrupt.

5 DISCUSSION

We have introduced a multi-agent simulation framework with which to study a platform-mediated economy in the presence of market shocks, making use of RL to model the decision making of a rational, self-interested platform. Given the prominence of platforms in today's economic systems and complexity of these systems, this work illustrates an important new application of AI methods. We use the framework to study and interpret the effect of several regulations on the efficiency and resilience of the overall economic system under optimal platform responses, suggesting caution in regard to some interventions and giving support to a particular kind of intervention.

Several extensions are of interest, including platform policies that combine fee-setting and matching, platforms with incomplete information about agent queries and locations, buyer preferences and seller offerings that can adapt over time, and achieving greater scale. There also remain several challenges to be addressed before these kinds of AI-based frameworks can guide economic system design in real-world settings. First and foremost is to collect data from existing markets in order to calibrate simulation frameworks, in regard to both environment settings and agent behavior. Second, any learned AI policies should be interpretable to ensure transparency and fairness to all participants in the digital economy.

REFERENCES

[1] Kabir Ahuja, Vishwa Chandra, Victoria Lord, and Curtis Peens. 2021. Ordering in: The rapid evolution of food delivery. *McKinsey & Company* (2021).

[2] Mark Armstrong. 2006. Competition in Two-Sided Markets. *The RAND Journal of Economics* 37, 3 (2006), 668–691. http://www.jstor.org/stable/25046266

[3] National Restaurant Association. 2020. Restaurant Industry in Free Fall; 10,000 Close in Three Months. (2020).

[4] Gianluca Brero, Alon Eden, Matthias Gerstgrasser, David C. Parkes, and Duncan Rheingans-Yoo. 2021. Reinforcement Learning of Sequential Price Mechanisms. In *Thirty-Fifth AAAI Conference on Artificial Intelligence*. 5219–5227.

[5] Bernard Caillaud and Bruno Jullien. 2003. Chicken & Egg: Competition among Intermediation Service Providers. *The RAND Journal of Economics* 34, 2 (2003), 309–328. http://www.jstor.org/stable/1593720

[6] Minmin Chen, Alex Beutel, Paul Covington, Sagar Jain, Francois Belletti, and Ed Chi. 2019. Top-K Off-Policy Correction for a REINFORCE Recommender System. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining*. 456–464.

[7] Thomas Degris, Patrick M. Pilarski, and Richard S. Sutton. 2012. Model-Free reinforcement learning with continuous action in practice. In *2012 American Control Conference (ACC)*. 2177–2182.

[8] Jean-Pierre Dubé, Günter J Hitsch, and Peter E Rossi. 2009. Do switching costs make markets less competitive? *Journal of Marketing research* 46, 4 (2009), 435–445.

[9] Jean-Pierre Dubé, Günter J Hitsch, and Peter E Rossi. 2010. State dependence and alternative explanations for consumer inertia. *The RAND Journal of Economics* 41, 3 (2010), 417–445.

[10] Joseph Farrell and Carl Shapiro. 1988. Dynamic competition with switching costs. *The RAND Journal of Economics* (1988), 123–137.

[11] Benjamin R Handel. 2013. Adverse selection and inertia in health insurance markets: When nudging hurts. *American Economic Review* 103, 7 (2013), 2643–82.

[12] Matthew J. Hausknecht and Peter Stone. 2015. Deep Recurrent Q-Learning for Partially Observable MDPs. *CoRR* abs/1507.06527 (2015).

[13] Nicolas Heess, Jonathan J. Hunt, Timothy P. Lillicrap, and David Silver. 2015. Memory-based control with recurrent neural networks. *CoRR* abs/1512.04455 (2015).

[14] Elisabeth Honka. 2014. Quantifying search and switching costs in the US auto insurance industry. *The RAND Journal of Economics* 45, 4 (2014), 847–884.

[15] Eugene Ie, Chih-Wei Hsu, Martin Mladenov, Vihan Jain, Sanmit Narvekar, Jing Wang, Rui Wu, and Craig Boutilier. 2019. RecSim: A Configurable Simulation Platform for Recommender Systems. *CoRR* abs/1909.04847 (2019).

[16] Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. 1998. Planning and acting in partially observable stochastic domains. *Artificial Intelligence* 101, 1 (1998), 99–134.

[17] Zhuoxin Li and Gang Wang. 2021. Regulating Powerful Platforms: Evidence from Commission Fee Caps in On-Demand Services. https://ssrn.com/abstract=3871514

[18] Alexander MacKay and Marc Remer. 2021. Consumer Inertia and Market Power. https://ssrn.com/abstract=3380390

[19] Martin Mladenov, Elliot Creager, Omer Ben-Porat, Kevin Swersky, Richard S. Zemel, and Craig Boutilier. 2020. Optimizing Long-term Social Welfare in Recommender Systems: A Constrained Matching Approach. In *Proceedings of the 27th International Conference on Machine Learning*. 6987–6998.

[20] Fernando Nogueira. 2014. Bayesian Optimization: Open source constrained global optimization tool for Python. https://github.com/fmfn/BayesianOptimization

[21] E. Shin Oblander and Daniel McCarthy. 2022. Persistence of Consumer Lifestyle Choices: Evidence from Restaurant Delivery During COVID-19. https://ssrn.com/abstract=3836262

[22] Christos H. Papadimitriou, George Pierrakos, Christos-Alexandros Psomas, and Aviad Rubinstein. 2016. On the Complexity of Dynamic Mechanism Design. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, Robert Krauthgamer (Ed.). SIAM, 1458–1475.

[23] Manav Raj, Arun Sundararajan, and Calum You. 2021. COVID-19 and Digital Resilience: Evidence from Uber Eats. https://ssrn.com/abstract=3625638

[24] Jean-Charles Rochet and Jean Tirole. 2003. Platform Competition in Two-Sided Markets. *Journal of the European Economic Association* 1, 4 (06 2003), 990–1029.

[25] Ruslan Salakhutdinov and Andriy Mnih. 2007. Probabilistic Matrix Factorization. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*. 1257–1264.

[26] Weiran Shen, Binghui Peng, Hanpeng Liu, Michael Zhang, Ruohan Qian, Yan Hong, Zhi Guo, Zongyao Ding, Pengjun Lu, and Pingzhong Tang. 2020. Reinforcement Mechanism Design: With Applications to Dynamic Pricing in Sponsored Search Auctions. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence*. AAAI Press, 2236–2243.

[27] Matthew Shum. 2004. Does advertising overcome brand loyalty? Evidence from the breakfast-cereals market. *Journal of Economics & Management Strategy* 13, 2 (2004), 241–272.

[28] Michael Sullivan. 2022. Price Controls in a Multi-Sided Market. https://m-r-sullivan.github.io/assets/papers/food_delivery_cap.pdf

[29] Richard S. Sutton and Andrew G. Barto. 1998. *Reinforcement Learning: an Introduction.* MIT Press.

[30] Pingzhong Tang. 2017. Reinforcement mechanism design. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. 5146–5150.

[31] Daan Wierstra, Alexander Foerster, Jan Peters, and Jürgen Schmidhuber. 2007. Solving Deep Memory POMDPs with Recurrent Policy Gradients. In *Artificial Neural Networks – ICANN 2007*, Joaquim Marques de Sá, Luís A. Alexandre, Włodzisław Duch, and Danilo Mandic (Eds.). 697–706.

[32] Ruohan Zhan, Konstantina Christakopoulou, Ya Le, Jayden Ooi, Martin Mladenov, Alex Beutel, Craig Boutilier, Ed Chi, and Minmin Chen. 2021. Towards Content Provider Aware Recommender Systems: A Simulation Study on the Interplay between User and Provider Utilities. In *Proceedings of the Web Conference 2021*. 3872–3883.

[33] Stephan Zheng, Alexander Trott, Sunil Srinivasa, David C. Parkes, and Richard Socher. 2022. The AI Economist: Taxation policy design via two-level deep multiagent reinforcement learning. *Science Advances* 8, 18 (2022), eabk2607.

A SUBSCRIPTION-LEVEL DECISION

Calculations for Subscription Effect Estimation. We first describe the subscription effect estimates for each type of the agents. To recap, these estimates, denoted ξ_{k+1}, are conducted based on the new platform fees and world transaction friction, under the same set of queries submitted or received in the last epoch (i.e., epoch k) and the same subscription decisions made by other agents.

- **A buyer b, on-platform in epoch k.** We denote the estimate of epoch surplus if buyer b does not subscribe to the platform as $\xi^{w}_{b,k+1}$. For this, given each query of b in epoch k, the choice in the world under new friction μ_{k+1} (Section 2.2) is reevaluated
$$\xi^{w}_{b,k+1} = \sum_{t\in k} u^{w}_{b,t}(\mu_{k+1}) = \sum_{t\in k} \max\{u_{\mathcal{B}}(q_{b,t}, s^{*}_{w}) - \mu_{k+1}, 0\},$$
where we denote $u^{w}_{b,t}(\mu)$ the world matching surplus for buyer b at t under the world friction μ, i.e., $u^{w}_{b,t}(\mu) = \max\{u_{\mathcal{B}}(q_{b,t}, s^{*}_{w}) - \mu, 0\}$. Similarly, we denote $\xi^{p}_{b,k+1}$ the estimate of epoch surplus if buyer b remains on platform. The new friction may affect the choice between a platform seller and a world seller. This surplus is estimated based on updated decisions under μ_{k+1}, with
$$\xi^{p}_{b,k+1} = -P_{\mathcal{B},k+1} + \sum_{t\in k} \max\{u^{w}_{b,t}(\mu_{k+1}), u^{p}_{b,t}\}.$$
- **A buyer b, off-platform in epoch k.** The surplus estimate of remaining off-platform depends on the new friction, i.e., $\xi^{w}_{b,k+1} = \sum_{t\in k} u^{w}_{b,t}(\mu_{k+1})$. For $\xi^{p}_{b,k+1}$, we need to estimate the epoch surplus if b subscribes to the platform. We assume that buyer b can observe platform-recommended sellers, e.g., this can be from trial periods to gain platform experience or a platform's estimate based on past orders. For a query sequence $\{q_{b,t}\}_{t\in k}$, denote the corresponding best platform-recommended sellers as $\{s^{p*}_{b,t}\}$. The estimated surplus if subscribing to the platform is
$$\xi^{p}_{b,k+1} = -P_{\mathcal{B},k+1} + \sum_{t\in k} \max\{u^{w}_{b,t}(\mu_{k+1}), u_{\mathcal{B}}(q_{b,t}, s^{p*}_{b,t})\}.$$
- **A seller s, on-platform in epoch k.** For $\xi^{w}_{s,k+1}$, we need to estimate the epoch surplus if s does not subscribe to the platform, by reasoning about (1) how many more world transactions would happen if s is not on the platform, and (2) how buyers' transaction decisions may be affected by the epoch $k+1$ world friction. To facilitate a precise estimate, we assume that seller s can observe the sequence of query and seller candidate tuples $\{(q_{b,t}, s, s^{p}_{b,t})\}_{t\in k}$ in which seller s is chosen as the best world option. Given $q_{b,t}$, $\mathcal{S}_k \backslash s$, and a fixed platform matching strategy used in epoch k, we denote the updated, best-platform seller as $s^{p*}_{b,t}$. For this modified sequence $\{(q_{b,t}, s, s^{p*}_{b,t})\}$, we consider the choices buyers will make under the new friction μ_{k+1}, and estimate the number of transactions seller s will receive without being on the platform, denoted $n^{w'}_{s,k}$. Thus, the estimated surplus if seller s is off platform is $\xi^{w}_{s,k+1} = n^{w'}_{s,k} v^{1}_{s}(1-\omega_s)$.
For $\xi^{p}_{s,k+1}$, we reason about how the new friction affects the number of transactions. Given the sequence $\{(q_{b,t}, s^{w}_{b,t}, s)\}_{t\in k}$ where s is picked as the best platform seller, we consider each buyer's choice and estimate the number of platform visits under μ_{k+1}, denoted $n^{p^{*}}_{s,k}$. Given the sequence $\{(q_{b,t}, s, s^{p}_{b,t})\}$ where s is picked as the best world seller, we estimate the number of world transactions under μ_{k+1} (even if seller s chooses to stay on the platform), denoted $n^{w^{*}}_{s,k}$. Thus, the estimated surplus that seller s would receive by subscribing to the platform is
$$\xi^{p}_{s,k+1} = -P_{\mathcal{S},k+1} + n^{p^{*}}_{s,k} v^{1}_{s}(1-\omega_s - P_{R,k+1}) + n^{w^{*}}_{s,k} v^{1}_{s}(1-\omega_s).$$
- **A seller s, off-platform in epoch k.** For $\xi^{p}_{s,k+1}$, we reason about the surplus from subscribing to the platform, e.g., by asking the platform for an estimate on the number of platform transactions. Given the sequence of queries on the platform, i.e., $\{q_{b,t}\}_{b\in\mathcal{B}_k, t\in k}$, the platform can update the matches it would suggest s also on-platform by following its matching strategy. Denote the sequence of query and updated seller matches with tuples $\{(q_{b,t}, s^{w}_{b,t}, s^{p^{*}}_{b,t})\}_{b\in\mathcal{B}_k, t\in k}$. With μ_{k+1}, we estimate the numbers of platform transactions $n^{p^{*}}_{s,k}$ and world transactions $n^{w^{*}}_{s,k}$. The estimated surplus if seller s subscribes to the platform is
$$\xi^{p}_{s,k+1} = -P_{\mathcal{S},k+1} + n^{p^{*}}_{s,k} v^{1}_{s}(1-\omega_s - P_{R,k+1}) + n^{w^{*}}_{s,k} v^{1}_{s}(1-\omega_s).$$
Seller s also adjusts the surplus that they would get by remaining off platform, reasoning about how buyers' transaction decisions will be affected by the new world friction. Given the sequence of query and matched seller tuples $\{(q_{b,t}, s, s^{p}_{b,t})\}_{b\in\mathcal{B}, t\in k}$ in which seller s was chosen as the best world seller, we re-evaluate each buyer's choice to get $n^{w'}_{s,k}$. The estimated off-platform surplus is
$$\xi^{w}_{s,k+1} = n^{w'}_{s,k} v^{1}_{s}(1-\omega_s).$$

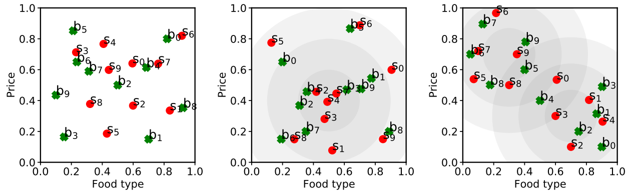
Calculations for Agent-Specific Decision Inertia. There is a rich body of literature that establishes *decision inertia*, modeling the presence of such inertia across different markets (see Section 2.4). In our setting an agent, either a buyer or seller, in subscription state $\mathcal{I}^{p}_{k} \in \{0,1\}$, is prone to stay in the same state in epoch $k+1$, due to habit formation, loyalty, or inattention. We treat buyers and sellers in the same way and illustrate the concept with a buyer b for simplicity. Each buyer starts with an initial preference in regard to adopting the platform or not, denoted by an integer $\chi_{b,0} \sim U[-\chi, \chi]$ for some integer χ. If the initial $\chi_{b,0}$ is positive, the buyer subscribes at the warm-up epoch, otherwise they do not. The inertia $\chi_{b,k}$ maps into an additive bonus to either the surplus for joining the platform or remaining in the world through the functional form:

$$\sigma^{p}_{b,k+1} := \mathcal{I}^{p}_{b,k} \log\left(\chi_{b,k}\right), \qquad \sigma^{w}_{b,k+1} := (1-\mathcal{I}^{p}_{b,k}) \log\left(-\chi_{b,k}\right). \quad (7)$$

That is, the longer an agent sticks to their decision, the larger the bias term gets, which increases in a concave way (logarithmically) over time. Such interpretation of decision inertia as an additive bonus is common in the literature [8, 10, 18]. Based on this adjusted utility, agents decide whether to subscribe or not according to probabilities inferred by the standard *discrete-choice logit* model [8, 18], where the probability of subscribing to the platform is

$$\delta^{p}_{b,k+1} := \frac{\exp\left(\xi^{p}_{b,k+1}+\sigma^{p}_{b,k+1}\right)}{\exp\left(\xi^{p}_{b,k+1}+\sigma^{p}_{b,k+1}\right)+\exp\left(\xi^{w}_{b,k+1}+\sigma^{w}_{b,k+1}\right)}.$$

The inertia is updated after each decision in the following way. If $\chi_{b,k} > 0$ and the buyer subscribes, then $\chi_{b,k+1} := \chi_{b,k} + 1$, and otherwise $\chi_{b,k+1} := -1$ (i.e., it resets for off-platform). Similarly if $\chi_{b,k} < 0$ and the buyer decides to stay off-platform, then $\chi_{b,k+1} := \chi_{b,k} - 1$, and otherwise reset $\chi_{b,k+1} := 1$.



(a) Uniform. (b) Core-and-Niche. (c) Two-Core.

Figure 5: Three types of latent locations of buyers and sellers.

B THE PLATFORM MATCHING POMDP

Based on the fee-setting POMDP (Section 3.1), we make the following adjustments to define the matching POMDP:

- We define $x_k \in \mathcal{X}$ for epoch k as the state of the market after the platform has set fees and agents have chosen to subscribe, but still before the first query is submitted. That is, the state includes agent-subscription states $\mathcal{I}^p_{\mathcal{B},k}$ and $\mathcal{I}^p_{\mathcal{S},k}$ and platform fees, i.e., $P_{\mathcal{B},k}, P_{\mathcal{S},k}, P_{R,k}$, for the upcoming epoch k. We also include the matching utility threshold adopted in the previous epoch, η_{k-1}, to state x_k. All other elements of the state remain the same.
- Here, the platform's action a_k chooses (i) the matching utility threshold for epoch k, and (ii) the matching rule for the epoch, whether seller-aware or profit-driven. For the threshold, we consider η_k that takes discrete values in $[0, 1]$.
- Different from the fee-setting POMDP, the state transition for platform matching starts with buyer queries: (1) a buy agent generates a query, (2) if the buyer is on platform, the platform recommends a seller based on η_k and its matching rule, and (3) the buyer selects a seller with whom to transact (Section 2.2). This gives the full sequence of queries, Q_k. At the end of epoch k, buyers and sellers observe the new fees $P_{\mathcal{B},k+1}$, $P_{\mathcal{S},k+1}$ and $P_{R,k+1}$, as given by a fixed fee schedule, and decide whether or not to subscribe to the platform for the next epoch.
- The reward $r_k \sim \mathcal{R}(x_k, a_k)$ of the platform for epoch k includes both the referral fees from epoch k and the subscription fees collected from buyers and sellers, reflecting the decision in regard to whether or not to join the platform for epoch $k+1$. This avoids delayed reward for the platform, as the registration decisions, and thus registration fees for the next epoch, are influenced by the platform's matching strategy (i.e., action a_k) during epoch k.
- The platform's observability of information in the state follows the same as for the fee-setting POMDP.

C DEFERRED MATERIALS FOR SECTION 4

C.1 Market Environments

Figure 6 red line plots the average shock of two-hundred simulation runs. Each run includes a pre-shock stage (epoch 1-3) with $\mu = 0.1$, a shock stage (epoch 4-9) where we sample the shock intensity $I \sim U[0.8, 1]$ and the world transaction frictions according to Section 4.1, and a post-shock stage with $\mu = 0.1$ (epoch 9-12).

C.2 Implementation Details

Observation features. Following our POMDP formulation, we make the following observations available to the platform:

- On-platform buyers and sellers, represented by two binary vectors, and their latent locations,

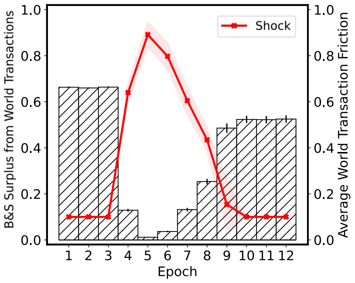


Figure 6: Buyer and seller surplus in markets without a platform across the 12 epochs with a full cycle of economic shock.

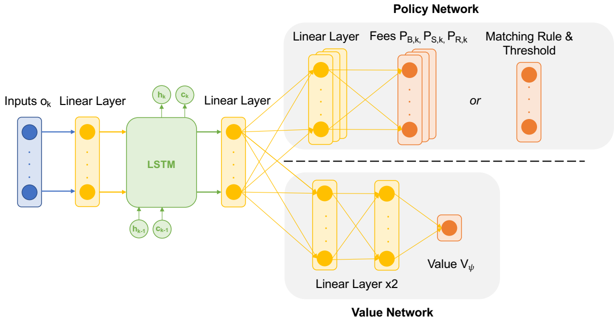


Figure 7: Neural network structure for the platform policies.

- Summary statistics of on-platform agents, including the number of platform transactions and platform surplus accumulated so far within an epoch,
- The platform matching and transaction matrices between *on-platform* buyers and sellers for the past epoch,
- The platform fees, the matching rule and utility threshold (if learn matching), and the current epoch's world friction.

Neural network structure and training parameters. Based on preliminary explorations, we design the actor and critic to share a fully-connected layer, LSTM cells of size 128, and again a fully-connected layer to recover sufficient statistics of the history, using this to in effect infer the knowledge structure of buyers and the demand elasticity of agents to platform fees. Each network also has its own two fully-connected layers. The critic outputs the value $V_\psi(o)$ of an observation o, and the actor gives policy π_θ for an observation o. For the fee-setting actor, this includes three separate output layers, with each returning a vector of probabilities for one type of platform fee. For the matching actor, this is a vector of probabilities over the matching utility thresholds that are applicable to both matching rules. Figure 7 illustrates the neural network structure we implement. Besides the policy gradient loss, we apply *entropy regularization* to the policy network to encourage exploration. The respective losses for the policy network and the value network are, $\mathcal{L}_\pi = -\log \pi(a_k|o_k;\boldsymbol{\theta})(R_k - V_\psi(o_k)) - \beta\mathcal{H}(\pi(A_k|o_k;\boldsymbol{\theta}))$ and $\mathcal{L}_V = (R_k - V_\psi(o_k))^2$, where $\mathcal{H}$ denotes the entropy over learned action probabilities.

We tune the platform agent with various combinations of learning rates $\{0.0001, 0.0005, 0.001\}$, batch sizes $\{4, 16, 32, 64, 128\}$, and entropy weights $\{0.001, 0.01, 0.05\}$, and select hyperparameters that maximize the objective function. We report detailed training parameters in the supplemental material.