Competing Bandits:
The Perils of Exploration under Competition*

Guy Aridor † Yishay Mansour ‡ Aleksandrs Slivkins § Zhiwei Steven Wu¶

First version: July 2020
This version: December 2022

Abstract

Most online platforms strive to learn from interactions with users, and many engage in *exploration*: making potentially suboptimal choices for the sake of acquiring new information. We study the interplay between *exploration* and *competition*: how such platforms balance the exploration for learning and the competition for users. Here users play three distinct roles: they are customers that generate revenue, they are sources of data for learning, and they are self-interested agents which choose among the competing platforms.

We consider a stylized duopoly model in which two firms face the same multi-armed bandit problem. Users arrive one by one and choose between the two firms, so that each firm makes progress on its bandit problem only if it is chosen. Through a mix of theoretical results and numerical simulations, we study whether and to what extent competition incentivizes the adoption of better bandit algorithms, and whether it leads to welfare increases for users. We find that stark competition induces firms to commit to a "greedy" bandit algorithm that leads to low welfare. However, weakening competition by providing firms with some "free" users incentivizes better exploration strategies and increases welfare. We investigate two channels for weakening the competition: relaxing the rationality of users and giving one firm a first-mover advantage. Our findings are closely related to the "competition vs. innovation" relationship, and elucidate the first-mover advantage in the digital economy.

Keywords: Competition vs. innovation, exploration vs. exploitation, multi-armed bandits, regret.
JEL Codes: D83, L15, O31

---

*All theoretical results are from Mansour et al. (2018), and all numerical simulations are from Aridor et al. (2019) (which was published as a 2-page abstract in ACM EC 2019). This manuscript features a unified and streamlined presentation, expanded related work, and revised background materials (*e.g.,* Appendices A,B are new).

We thank Ian Ball, Yeon-Koo Che, Bar Light, Sven Rady, Sara Shahanaghi and Glen Weyl for helpful comments and conversations. We also thank the audience of the conference and seminar talks on this work. All errors are our own.

†Northwestern University Kellogg, Department of Marketing. Email: guy.aridor@kellogg.northwestern.edu.

‡Tel Aviv University, Department of Computer Science, and Google Tel Aviv. Email: mansour.yishay@gmail.com.
Supported in part by Israel Science Foundation grant number 993/17.

§Microsoft Research New York City. Email: slivkins@microsoft.com.

¶Carnegie Mellon University, Pittsburgh, PA Email: zstevenwu@cmu.edu.
Research done when Z.S. Wu was an intern and a postdoc at Microsoft Research NYC.

Contents

1 Introduction

Learning from interactions with users is ubiquitous in modern customer-facing platforms, from product recommendations to web search to content selection to fine-tuning user interfaces. Many platforms purposefully implement *exploration*: making potentially suboptimal choices for the sake of acquiring new information. Online platforms routinely deploy A/B tests, and are increasingly adopting more sophisticated exploration methodologies based on *multi-armed bandits*, a standard and well-studied framework for exploration and making decisions under uncertainty. This trend has been stimulated by two factors: almost-zero cost of deploying iterations of a product (provided an initial infrastructure investment), and the fact that many online platforms primarily compete on product quality, rather than price (*e.g.,* because they are supported by ads or cheap subscriptions).

In this paper, we study the interplay between *exploration* and *competition*.[1] Platforms that engage in exploration typically need to compete against one another. Most importantly, platforms compete for users, who benefit them in two ways: generating revenue and providing data for learning. This creates a tension: while exploration may be essential for improving the service tomorrow, it may degrade the service quality *today*, in which case some of the users can leave and there will be fewer users to learn from. This may create a "data feedback loop" when the platform's performance further degrades relative to competitors who keep learning and improving from *their* users, and so forth. Taken to the extreme, such dynamics may cause a "death spiral" effect when the vast majority of customers eventually switch to competitors.

The main high-level question we ask is: **Whether and how does competition between platforms incentivize the adoption of better exploration algorithms?** This translates into a number of more concrete questions. While it is commonly assumed that better technology always helps, is this so under competition? Does increased competition lead to higher consumer welfare? How significant are the data feedback loops and how they relate to the anti-trust considerations? We offer a mix of theoretical results and numerical simulations, in which we study complex interactions between platforms' learning dynamics and users' self-interested behavior. Prior work on exploration vs. competition targets technically very different models of competition which are not amenable to our high-level question (as we discuss in Section 2).

Our model: competition game. We consider a stylized duopoly model in which two firms (*principals*) compete for users (*agents*). Principals compete on quality rather than on prices, and engage in exploration in order to learn which actions lead to high quality products. Agents arrive sequentially. A new agent arrives and chooses a principal (more on this below). The principal selects an action which affects the quality of service provided to this agent, *e.g.,* a list of web search results. The agent experiences this action and the resulting reward from this action is observed by the principal. Each principal only observes its own users. Principals commit to their strategies in advance, so as to maximize their market share.

In more detail, the principal-side model is as follows. Each principal faces a basic and well-studied version of the multi-armed bandit problem, where each reward is drawn independently from a fixed, action-specific distribution. Each principal's pure strategy is a multi-armed bandit algorithm, which dynamically adjusts to the observed rewards. However, it is oblivious to all signals on competition (such as the market share or the competitor's choices or rewards), even when such signals are available. This modeling choice reflects the reality of industrial applications, which

[1] *I.e.,* we add *competition* to the standard exploration-exploitation tradeoff studied in multi-armed bandits.

follow a huge body of knowledge in machine learning; more on this in Section 3.1. Due to similar practical considerations, we expect the actual strategy choice to reflect the competition game only in a crude, qualitative way. Hence, basic outcomes under competition are worth studying *per se*, not only as a stepping stone to equilibrium characterization.

To flesh out the meaning of *better* exploration algorithms, as per the main question, we draw on the literature from machine learning. We consider algorithm's performance *in isolation*: in a standalone exploration problem without competition. Rewards are not discounted with time, and we focus on big, qualitative differences in asymptotic regret rates. One baseline is algorithms that do not purposefully explore, and instead make myopically optimal decisions; we call them *greedy algorithms*. In isolation, they are known to perform poorly for a wide variety of problem instances.

The agent-side model is as follows. When an agent arrives, she forms a reward estimate for each principal, and then chooses a principal using these reward estimates according to some fixed decision rule. Modeling the reward estimates is subtle, as one needs to specify how the agents know and interpret the principals' algorithms and the algorithms' past performance. We consider two extremes for this issue: Bayesian agents that know the algorithms but do not observe the past performance, and frequentist agents that observe each algorithm's recent average performance ("reputation score") but have no prior knowledge or beliefs on the algorithms. The Bayesian model follows a standard rational "template" in theoretical economics, whereas the frequentist model is more realistic. We find that the former is amenable to theoretical analysis, and the latter to simulations. Our main findings are similar for both.

Throughout our results, we investigate two channels for weakening the competition: relaxing the rationality of users (via their decision rule) and giving one firm a first-mover advantage.

Theoretical results in a Bayesian model. We consider a Bayesian model (called the *Bayesian-choice model*), where agents have a common Bayesian prior on reward distributions, know the principals' algorithms and their own arrival times, but do not observe the previous agents' choices or rewards. Each agent computes Bayesian-expected rewards for both principals, and uses them as reward estimates to decide which principal to choose. Our results depend crucially on agents' decision rule:

(i) The most obvious decision rule maximizes the reward estimate; we call it `HardMax`. We find that it is not conducive to adopting better algorithms: each principal's dominant strategy is to choose the greedy algorithm. Further, if the tie-breaking is probabilistically biased in favor of one principal, the latter can always prevail in competition.

(ii) We dilute the `HardMax` agents with a small fraction of "random agents" who choose a principal uniformly at random. (They can be interpreted as consumers that are oblivious to the principals' reputation.) We call this model `HardMax&Random`. Then better algorithms help in a big way: when two algorithms compete against one another, a sufficiently better algorithm is guaranteed to win all non-random agents after an initial learning phase. There is a caveat, however: any algorithm can be defeated by interleaving it with the greedy algorithm. Consequently, a better algorithm may sometimes lose in competition, and a pure Nash equilibrium typically does not exist.

(iii) We further soften the decision rule so that the selection probabilities vary smoothly in terms of the reward estimates. We call it `SoftMax`, a more realistic middle ground between `HardMax` and random agents.In the most technical result of the paper, we find that a sufficiently better algorithm prevails under much weaker assumptions.

Numerical simulations in a frequentist model. We then consider a frequentist model (called the *reputation-choice model*), where agents observe signals about the principals' past performance and make their decisions naively, without invoking any prior knowledge or beliefs. The performance signals are aggregated as a scalar *reputation score* for each principal, modeled as a sliding window average of its rewards. Thus, agents' decision rule depends only on the two reputation scores. While this model uses more realistic assumptions about agent choices and allows us to characterize outcomes in finite samples, we provide numerical simulations to characterize the outcomes of interest. This allows us to refine and expand the results from the Bayesian model in several ways:

(i) We find that the greedy algorithm often wins under the `HardMax` decision rule, with a strong evidence of the "death spiral" effect mentioned earlier. As predicted by the theory, better algorithms prevail under `HardMax&Random` with enough "random" users.

(ii) Focusing on `HardMax`, we investigate the first-mover advantage as a different channel to vary the intensity of competition: from the first-mover to simultaneous entry to late-arriver. We find that the first-mover is incentivized to choose a more advanced exploration algorithm, whereas the late-arriver is often incentivized to choose the "greedy algorithm" (more so than under simultaneous entry). Consumer welfare is higher under early/late arrival than under simultaneous entry. We frame these results in terms of an inverted-U relationship.

(iii) However, the greedy algorithm is sometimes *not* the best strategy under high levels of competition.[2] We revisit algorithms' performance in a standalone bandit problem, *i.e.,* without competition. We find that the most natural performance measure does not explain this phenomenon, and suggest a new, more nuanced one that does.

(iv) We decompose the first-mover advantage into two distinct effects: free data to learn from (*data advantage*), and a more definite, and possibly better reputation compared to an entrant (*reputation advantage*), and run additional experiments to separate and compare them. We find that either effect alone leads to a significant advantage under competition. The data advantage is larger than reputation advantage when the incumbent commits to a more advanced bandit algorithm. Finally, we find an "amplification effect" of the data advantage: even a small amount thereof gets amplified under competition, causing a large difference in eventual market shares.

Economic Interpretations. Our findings are consistent with Schumpeter's inverted-U relationship between competition and innovation, whereby too little or too much competition is bad for innovation, but intermediate levels of competition tend to be better (Schumpeter, 1942; Aghion et al., 2005; Vives, 2008). We interpret innovation as the adoption of better exploration algorithms,[3] and control the severity of the competition by varying the agents' decision rule from `HardMax` (cut-throat competition) to `HardMax&Random` to `SoftMax` and all the way to the uniform selection. Another, technically different inverted-U relationship zeroes in on the `HardMax&Random` model.

Our model also speaks to policy discussions on regulating data-intensive digital platforms (Furman et al., 2019; Scott Morton et al., 2019), and particularly to the ongoing debate on the role of data in the digital economy. One fundamental question in this debate is whether data can serve a similar role as traditional "network effects", creating scenarios when only one firm can function in the market (Rysman, 2009; Jullien and Sand-Zantman, 2019). The death spiral/amplification effects mentioned above have a similar flavor: a relatively small performance loss due to exploration (resp.,

[2]In our theoretical results on `HardMax`, the greedy algorithm is always the best strategy, mainly because it is aware of the Bayesian prior (whereas in the simulations the prior is not available).

[3]Adoption of exploration algorithms tends to require substantial R&D effort in practice, even if the algorithms are well-known and/or similar technologies already exist elsewhere (*e.g.,* see Agarwal et al., 2017).

data advantage) gets amplified under competition and causes the firm to be starved of users (resp., take over most of the market). However, a distinctive feature of our approach is that the network effects arise endogenously.

Our results highlight that understanding the performance of learning algorithms in isolation does not necessarily translate to understanding their impact in competition, precisely due to the fact that competition leads to the endogenous generation of observable data. Approaches such as Lambrecht and Tucker (2015); Bajari et al. (2018); Varian (2018) argue that the diminishing returns to scale and scope of data in isolation mitigate such data feedback loops, but ignore the differences induced by learning in isolation versus under competition. Explicitly modeling the interaction between learning technology and data creation allows us to speak on how data advantages are characterized and amplified by the increased *quality* of data gathered by better learning algorithms, not just the quantity thereof. In particular, we find that incumbency is good for innovation and welfare, *and* creates a barrier to entry, all due to data feedback loops.

Significance. Our results have a dual purpose: shed light on real-world implications of some typical scenarios, and investigate the space of models for describing the real world. As an example for the latter: while the `HardMax` model with simultaneous entry is arguably the most natural model to study *a priori*, our results elucidate the need for more refined models with "free exploration" (*e.g.,* via random agents or early entry). On a technical level, we connect a literature on regret-minimizing bandits in machine learning and that on competition in economics.

The two technical parts of the paper, Bayesian/theoretical and frequentist/experimental, are on equal footing. While one does not provide direct experimental (resp., theoretical) justification for the other, they yield consistent conclusions, and present two complementary but different approaches to attack the same problem. Our theory takes a Bayesian perspective, standard in economic theory, and discovers several strong asymptotic results. Much of the difficulty, both conceptual and technical, is in setting up the model and the theorems. In particular, it was crucial to interpret the results and intuitions from the literature on multi-armed bandits so as to formulate meaningful and productive assumptions on bandit algorithms and Bayesian priors. The numerical simulations for the frequentist model provide a more nuanced and "non-asymptotic" perspective. In essence, we look for substantial effects within relevant time scales. (In fact, we start our investigation by determining what time scales are relevant in the context of our model.) The central challenge is to capture a huge variety of bandit algorithms and bandit problem instances with only a few representative examples, and arrive at findings that are consistent across the entire space.

The Bayesian model is suitable for analysis and the frequentist model for simulations, *but not vice versa*. A natural implementation of the Bayesian model requires running time quadratic in the number of rounds,[4] which precludes numerical simulations at a sufficient scale. The frequentist model features an intricate feedback loop between algorithms' performance, their reputations and agents' choices, which simplifies the simulations but does not appear analytically tractable.

[4] *E.g.,* this is because at each round t, one needs to recompute, and integrate over, a discrete distribution with t possible values, namely the number of agents that have chosen principal 1 so far.

2 Related work

Exploration. Multi-armed bandits (*MAB*) is an elegant and tractable abstraction for tradeoff between *exploration* and *exploitation*: essentially, between acquisition and usage of information. MAB problems have been studied for many decades by researchers from computer science, operations research, statistics and economics, generating a vast and multi-threaded literature. The most relevant thread concerns the basic model of regret-minimizating bandits with stochastic rewards and no auxiliary structure (which is the problem faced by each principal in our model), see Appendix A for background. This basic model has been extended in many different directions, with a considerable amount of work on each: *e.g.,* payoffs with a specific structure (*e.g.,* combinatorial, linear, convex or Lipschitz), payoff distributions that change over time, and auxiliary payoff-relevant signals. Dedicated monographs (Bubeck and Cesa-Bianchi, 2012; Slivkins, 2019; Lattimore and Szepesvári, 2020) cover the work on regret-minimizing formulations (which mainly comes from computer science). The classic book (Gittins et al., 2011) focuses on the Markovian formulations, which predate regret-minimization. Connections to economics are detailed in books (Cesa-Bianchi and Lugosi, 2006; Slivkins, 2019) and surveys (Bergemann and Välimäki, 2006; Hörner and Skrzypacz, 2017). Industrial applications are discussed in (Agarwal et al., 2017).

A monopolistic bandit algorithm may interact with self-interested parties, leading to a tension between exploration and incentives. This tension has been studied in several scenarios: incentivized exploration in recommendation systems (starting from Kremer et al. (2014); Che and Hörner (2018), see Slivkins (2019, Ch. 11)), dynamic auctions (Bergemann and Said, 2011), pay-per-click ad auctions (*e.g.,* Babaioff et al., 2014; Devanur and Kakade, 2009), coordinating search and matching (Kleinberg et al., 2016), and human computation (*e.g.,* Ho et al., 2016; Ghosh and Hummel, 2013). Unlike this work, we focus on incentives created in a competition.

Exploration and competition. Several papers consider exploration algorithms in scenarios when the explorer is not a monopolist. The technical models are very different, and not amenable to the high-level question articulated in the Introduction.

Bergemann and Välimäki (1997, 2000) and Keller and Rady (2003) study the interplay of exploration and competition for users when the competing firms experiment with *prices* (whereas in our model the firms experiment with design alternatives). All three papers consider environments with fixed product quality and dynamic strategies that respond to competition, and analyze Markov-perfect equilibria. In contrast, we consider a one-shot game where firms commit to algorithms for their bandit problem and the goal is to learn the best product alternative. This results in the nature of exploration being fundamentally different relative to these papers and, as such, we focus on different outcomes of interest relative to these papers.

In the line of work on *strategic experimentation* (starting from Bolton and Harris (1999); Keller et al. (2005), see Hörner and Skrzypacz (2017) for a survey), agents explore and learn over time in a shared environment. Thus, we have exploration algorithms which interact with each other strategically, *e.g.,* each agent prefers to free-ride on someone else's exploration. However, this work is all about cooperation (or lack thereof), rather than competition.

Several papers study competition between two principals who run algorithms but do not interact, directly or indirectly, until the very end of the game. Akcigit and Liu (2016) consider a "research competition" between two firms racing towards a big discovery. Each firm deploys a bandit algorithm with two arms, corresponding to safe and risky lines of research. The firms do not interact until one of them makes the discovery and wins the game. In the "dueling algorithms" framework of Immorlica et al. (2011), each principals runs an algorithm for the same problem. All inputs are observable at once, and principals' payoffs are binary (win/lose). Ben-Porat and Tennenholtz (2019) study competition between "offline" machine learning algorithms. In comparison, we study a "product competition" in which the two firms interact continuously (via the customers' choices), accrue rewards incrementally, and compete for individual customers.

A long line of work from electrical engineering and computer science, starting from Lai et al. (2008); Liu and Zhao (2010), focuses on competition for resources, not competition for consumers. Specifically, this literature targets an application to *cognitive radios*, where multiple radios transmit simultaneously in a shared medium and compete for bandwidth. Each radio chooses channels over time using a multi-armed bandit algorithm. This work studies a repeated game between bandit algorithms, and focuses on designing algorithms which work well in this game.

Competition. The competition vs. innovation relationship and the inverted-U shape thereof have been introduced in a classic book (Schumpeter, 1942), and remained an important theme in the literature ever since (*e.g.,* Aghion et al., 2005; Vives, 2008). This literature treats innovation as R&D that improves the products and, R&D costs aside, is a priori beneficial for the firm. In contrast, we focus on innovation in *exploration technology* which systematically improves the firm's products and crucially depends on data generated by the firm's customers. In particular, we find that such innovation may potentially hurt the firm. We recover the inverted-U relationship purely through the reputational consequences of exploration, whereas prior work relies on costs and profits.

The literature on learning-by-doing vs. competition (*e.g.,* Fudenberg and Tirole, 1983; Dasgupta and Stiglitz, 1988; Cabral and Riordan, 1994) studies firms that learn while competing against each other, so that a firm attracting more consumers reduces its production costs. Our model differs in several important respects. First, firms learn to improve product quality rather than to reduce production costs. Second, the firms' current actions have consequences (via reputation and/or data collected by the algorithm) that directly impact consumer choices in the future. Third, we endogenize the technology behind learning-by-doing by explicitly considering bandit algorithms.

A line of work on *platform competition* (starting with Rysman (2009), see Weyl and White (2014) for a survey) concerns competition between firms that improve as they attract more users. This literature is not concerned with *innovation*, and typically models network effects exogenously, whereas they are endogenous in our model. A nascent literature studies network effects in data-intensive markets (Prufer and Schottmüller, 2017; Hagiu and Wright, 2020), but typically models learning as a reduced-form function of past consumer history and focuses on the role of prices.

Schmalensee (1982); Bagwell (1990) investigate how buyer uncertainty about product quality can serve as a barrier to entry for late arrivers; we find a similar effect with "reputation advantage". De Corniere and Taylor (2020) note the role of data as a barrier to entry in online markets; we find a similar effect with "data advantage". Kerin et al. (1992) overview other channels through which first-mover advantage can affect competition.

While we use first-mover advantage and agents' decision rule, classic "market competitiveness" measures, such as the Lerner Index or the Herfindahl-Hirschman Index (Tirole, 1988), are not applicable to our setting, as they rely on ex-post observable market attributes such as prices or market shares (which are, resp., absent and endogenous for us).

Choice models. Stochastic choice models similar to ours are widely used in economics. "Random agents" (a.k.a. noise traders) can side-step the "no-trade theorem" (Milgrom and Stokey, 1982), a famous impossibility result in financial economics. They play a similar role in our model, side-stepping the dominance of the greedy algorithm. Moreover, `SoftMax` subsumes the logit choice rule, a standard behavioral model with strong empirical and microeconomic foundations (*e.g.,* Mosteller and Nogee, 1951; Luce, 1959; Matějka and McKay, 2015). Choice models similar to `SoftMax` are used to explain horizontal product differentiation (*e.g.,* Hotelling, 1929; Perloff and Salop, 1985).

3 Our model in detail

Principals and agents. There are two principals and T agents. We denote them, resp., principal $i \in \{1, 2\}$ and agent $t \in [T]$, where $[T] := \{1, 2, \ldots, T\}$.

In each round $t \in [T]$, the following interaction takes place. Agent t arrives and chooses a principal $i_t \in \{1, 2\}$. The principal chooses action $a_t \in A$, where A is a fixed set of actions.[5] The agent experiences this action and receives an associated reward $r_t \in \{0, 1\}$, which is then observed by the principal. We posit *stochastic rewards*: whenever a given action $a \in A$ is chosen, the reward is an independent draw from Bernoulli distribution with mean μ_a. In particular, the mean rewards μ_a, as well as the action set A, are the same for both principals and all rounds. The mean rewards are initially not known to anybody. The principals are completely unaware of the rounds when the opponent is chosen. Thus, each principal follows the protocol of *multi-armed bandits* (henceforth, *MAB*). That is: in each round when it is chosen, the principal picks an action from A and observes a reward for this action (and nothing else).

Each principal i commits to an MAB algorithm $\texttt{alg}_i$ before round 1, and uses this algorithm throughout the game. The algorithm proceeds in time-steps: each time it is called, it outputs an arm from A, and inputs a reward for this action. The algorithm is called only in game rounds when principal i is chosen. When the distinction between algorithm's time-steps and game rounds is unclear from the context, we will refer to them as, resp., *local steps/rounds* and *global rounds*.

Agent response. Each agent t forms a reward estimate $\texttt{EST}_i(t) \in [0, 1]$ for each principal i. (What these estimates are, and how much the agents know in order to form them, depends on the Bayesian vs. frequentist model variant.) The reward estimates determine the choice of the principal. Specifically, agent t chooses principal 1 with probability

$$p_t = f_{\texttt{resp}}\left(\texttt{EST}_1(t) - \texttt{EST}_2(t)\right), \tag{1}$$

where $f_{\texttt{resp}} : [-1, 1] \to [0, 1]$ is the *response function*, same for all agents. We assume that $f_{\texttt{resp}}$ is monotonically non-decreasing, is larger than $1/2$ on the interval $(0, 1]$, and smaller than $1/2$ on the interval $[-1, 0)$. We consider three variants for $f_{\texttt{resp}}$, depicted in Figure 1:

- `HardMax`: $f_{\texttt{resp}}$ equals 0 on the interval $[-1, 0)$ and 1 on the interval $(0, 1]$. In words, a `HardMax` agent deterministically chooses a principal with a higher reward estimate.

- `HardMax&Random`: $f_{\texttt{resp}}$ equals ϵ_0 on the interval $[-1, 0)$ and $1 - \epsilon_0$ on the interval $(0, 1]$, for some constant $\epsilon_0 \in (0, \frac{1}{2})$. In words, each agent is a `HardMax` agent with probability $1 - 2\epsilon_0$, and makes a random choice otherwise.

[5] We use 'action' and 'arm' interchangeably, as common in the literature on multi-armed bandits.

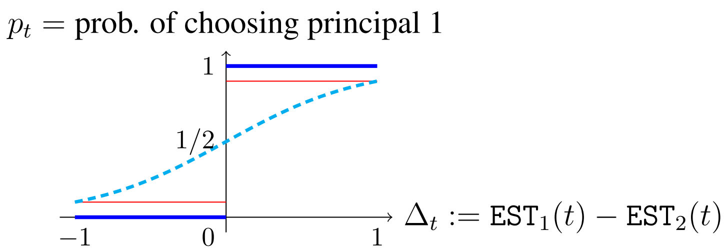

Figure 1: The models for $f_{\texttt{resp}}$: `HardMax` is thick blue, `HardMax&Random` is red, and `SoftMax` is dashed.

- `SoftMax`: $f_{\texttt{resp}}$ lies in $[\epsilon_0, 1-\epsilon_0]$, breaks ties fairly, and has a bounded derivative around 0 (see Definition 4.15 for a formal definition). Intuitively, $f_{\texttt{resp}}$ is a smoothed version of `HardMax&Random`, without a sharp threshold therein.

`HardMax&Random` and `SoftMax` agents can be interpreted in several ways. First, they make mistakes, due to lack of awareness or interest. Second, they give some chance to the non-preferred principal, due to curiosity or a behavioral effect like probability matching. Third, they can be realized as a distribution over more "basic" agent types. Indeed, the `HardMax&Random` distribution is a mixture of `HardMax` and "random agents" (which choose a principal uniformly at random). The latter can be interpreted as consumers that are completely oblivious to principals' reputation. One can obtain a `SoftMax` response function using agent types that choose a principal i with a largest reward estimate $\texttt{EST}_i$, unless $|\texttt{EST}_1 - \texttt{EST}_2|$ is upper-bounded by some parameter θ, in which case they choose uniformly. Then, we obtain `SoftMax` as a mixture of random agents and these "θ-`HardMax`" agents, for a suitable distribution over θ.

Bayesian vs. frequentist variants. We consider two model variants, Bayesian and frequentist (we use them, resp., for theoretical results and numerical simulations). The main difference between the two concerns the agents' reward estimates $\texttt{EST}_i(t)$.

In the *Bayesian-choice model*, the mean reward vector $\mu = (\mu_a : \; a \in A)$ is drawn from a common Bayesian prior $\mathcal{P}_{\texttt{mean}}$. Each agent knows its global round t, but not the performance signals such as the current market shares. Her reward estimates are defined as posterior mean rewards: $\texttt{EST}_i(t) = \texttt{PMR}_i(t) := \mathbb{E}\left[\, r_t \mid i_t = i \,\right]$ for each principal i, where the agent knows t, the principals' algorithms, $\mathcal{P}_{\texttt{mean}}$, and $f_{\texttt{resp}}$.

In the *reputation-choice model*, agents' reward estimate for a given principal is the average reward of the last M agents that chose this principal. We call it *reputation score*, and interpret it as the current reputation. To make it meaningful initially, each principal enjoys a "warm start": additional T_0 agents arrive before the game starts, and interact with the principal as described above.

Competition game. Some of our results explicitly study the game between the two principals, termed the *competition game*. We model it as a simultaneous-move game: before the first agent arrives, each principal commits to an MAB algorithm. Principals are risk-neutral; their utility is defined as their market share, *i.e.,* the number of agents they attract. Thus, they aim to select the algorithm that maximizes their expected market share.[6]

[6]The immediate goal of principals' MAB algorithms is (still) to maximize agents' rewards, so as to attract agents. Besides, it is unclear how to maximize market share directly within our model. Note that in extensions (Section 4.6) the principals' utility can also depend on rewards.

The distinction between a pure and mixed strategy in this game is worth clarifying. For each principal, a pure strategy commits to a particular MAB algorithm. A mixed strategy chooses a particular MAB algorithm at random from a mixture, and then commits to this algorithm. While a mixed strategy induces a randomized MAB algorithm, it differs from a pure strategy with the same algorithm in that the realization of the mixture is revealed to the agents.

Extensions. The Bayesian-choice model admits several extensions, detailed in Section 4.6. First, all/most results extend to arbitrary reward distributions, allow reward-dependent utility, and carry over to a more general version of multi-armed bandits. Second, agents could have beliefs on $(\texttt{alg}_1, \texttt{alg}_2, \mathcal{P}_{\texttt{mean}}, f_{\texttt{resp}})$ that need not be correct; then, all results carry over with respect to these beliefs. Third, we can handle a limited amount of non-stationarity in $f_{\texttt{resp}}$ for the HardMax&Random and SoftMax decision rules. Finally, the main result on HardMax extends to time-discounted utilities.

"Non-strategic" exploration strategies. We focus on a realistic scenario when the exploration strategies available to the principals are "non-strategic" in nature. Even though the principals play a multi-step game, they do not react to each other's moves or to the agents' strategic choices. This is how industry approaches exploration algorithms, for several reasons. First, "non-strategic" exploration is well-studied in machine learning, and yet it remains a very complex and actively studied subject in research. Even the seemingly simple algorithms are not straightforward to deploy in practice, and require a substantial investment in infrastructure (*e.g.,* see the discussions in Agarwal et al. (2017) and Slivkins (2019, Chapter 8.7)). Responding to the competition represents another layer of complexity which has not been previously studied in this context, to the best of our knowledge, let alone made even remotely practical. Second, the competitor's exploration strategy is typically not public, and understanding its exploration behavior via observations appears challenging even as a research problem. Third, while the principals could potentially react to the market share or the reputation scores, baking these signals into one's exploration strategy runs the risk of over-interpreting our competition model, as they may change for exogenous reasons. Alternatively, one could use such signals, as well as the intuitions coming from this paper, to guide the platform's decisions regarding exploration.

Bandit algorithms. Our treatment of bandit algorithms is very standard in machine learning — the primary community where these algorithms are designed and studied — but perhaps less standard in economic theory. The main tenets are as follows.

(i) Algorithms are designed for vanishing regret without time-discounting, and compared theoretically based on their asymptotic regret rates (rather than Bayesian-optimal time-discounted reward, as in Gittins index, a more standard economic perspective). Indeed, non-discounted, regret-minimizing formulations has been prevalent in the bandits literature over the past two decades (Slivkins, 2019; Lattimore and Szepesvári, 2020), and better correspond to practical deployments (*e.g.,* Agarwal et al., 2017).

(ii) A key distinction is between no exploration (the "greedy" algorithm), fixing the exploration schedule in advance ("exploration-separating" algorithms, *e.g.,* the epsilon-greedy algorithm), and adapting exploration to the past observations (*e.g.,* Thompson Sampling). Accordingly, there's a stark 3-way distinction in asymptotic regret rates. In particular, our numerical results use a standard, representative algorithm from each of the three classes.

Self-contained background regarding the two tenets can be found in Appendix A.

3.1 Discussion: stylized model

Our models are stylized in several important respects. Firms compete only on the quality of service, rather than, say, pricing or the range of products. Agents are myopic: they do not worry about how their actions impact their future utility.[7] On the machine learning side, we focus on qualitative distinctions described above, rather than state-of-art algorithms for realistic applications.

For the Bayesian-choice model, agents do not observe any signals about the principals' past performance, making agents' behavior independent of a particular realization of the prior. This allows us to summarize each learning algorithm via its Bayesian-expected rewards, not worrying about its detailed performance on particular realizations of the prior. Such summarization is essential for formulating lucid and general analytic results, let alone proving them.

For the reputation-choice model, various performance signals available to the users, from personal experience to word-of-mouth to consumer reports, are abstracted as persistent "reputation scores" reflecting the current reputation, and further simplified to average performance over a sliding time window. The reputation scores directly account for competition, allowing the users to have no direct information on the algorithms deployed or the bandit problem faced by the firms. The latter property makes our model amenable to numerical simulations.

4 Theoretical results: the Bayesian-choice model

In this section, we present our theoretical results for the Bayesian-choice model. While we provide intuition and proof sketches, the detailed proofs are deferred to Appendix D.

4.1 Preliminaries

Let $\texttt{rew}_i(n)$ denote the agent's realized reward observed by principal i at local step n, *i.e.,* the reward collected by algorithm $\texttt{alg}_i$ in this local step. For a global round t, let $n_i(t)$ denote the number of global rounds before t in which principal i is chosen. We will use the fact that $\texttt{PMR}_i(t) := \mathbb{E}\left[\, r_t \mid i_t = i \,\right] = \mathbb{E}\left[\, \texttt{rew}_i(n_i(t)+1) \,\right].$

Assumptions. We make two mild assumptions on the prior. First, each arm a can be best:

$$\forall a \in A: \quad \Pr\left[\, \mu_a > \mu_{a'} \quad \forall a' \in A \setminus \{a\} \,\right] > 0. \tag{2}$$

Second, posterior mean rewards are pairwise distinct given any feasible history h:[8]

$$\mathbb{E}[\mu_a \mid h] \neq \mathbb{E}[\mu_{a'} \mid h] \quad \forall a, a' \in A. \tag{3}$$

In particular, *prior* mean rewards are pairwise distinct: $\mathbb{E}[\mu_a] \neq \mathbb{E}[\mu'_a]$ for any $a, a' \in A$.

In Appendix C, we provide two examples for which property (3) is 'generic', in the sense that it can be enforced almost surely by a small random perturbation of the prior. The two examples concern, resp., Beta priors and priors with a finite support, and focus on priors $\mathcal{P}_{\texttt{mean}}$ that are independent across arms.

[7] So, agents do not attempt to learn over time, game future agents, or manipulate the principals' learning algorithms. This is arguably typical in practice, in part because one agent's influence tends to be small.

[8] The *history* of an MAB algorithm at a given step t comprises actions a_s and rewards r_s in all previous steps $s < t$. The history is *feasible* if for each s, reward r_s is in the support of the reward distribution for a_s.

MAB algorithms. We consider two baseline algorithms. The main one, called `BayesGreedy`, chooses an arm a with the largest posterior mean reward $\mathbb{E}[\mu_a \mid \cdot]$ given all information currently available to the algorithm. A more primitive baseline, called `StaticGreedy`, chooses an arm a with the largest prior mean reward $\mathbb{E}[\mu_a]$, and uses this arm in all rounds.

We characterize the inherent quality of an MAB algorithm in terms of its *Bayesian Instantaneous Regret* (henceforth, BIR), a standard notion from machine learning:

$$\texttt{BIR}_i(n) := \mathbb{E}[\, \max_{a \in A} \mu_a - \texttt{rew}_i(n) \,]. \tag{4}$$

We are primarily interested in how fast BIR decreases with n. (We treat the number of arms as a constant.) Intuitively, (much) better MAB algorithms tend to have a (much) smaller BIR, see Appendix A for background. An algorithm is called *Bayesian-monotone* if it can only get better over time, in the Bayesian sense: namely, if $\mathbb{E}[\texttt{rew}_i(\cdot)]$ is non-decreasing, and therefore $\texttt{BIR}(\cdot)$ is non-decreasing. This is a mild assumption, see Appendix B.

4.2 `HardMax` response function

We consider agents with `HardMax` response function, and show that principals are not incentivized to *explore*, *i.e.,* to deviate from `BayesGreedy`. The core technical result is that if one principal adopts `BayesGreedy`, then the other principal loses all agents as soon as he deviates therefrom. We make this formal below.

Definition 4.1. One MAB algorithm *deviates* from another at (local) step n if there is a set H of histories over the previous local steps such that both algorithms lead to H with positive probability, and choose different distributions over arms given any history $h \in H$. If $n = n_0$ is the smallest such step, we say alg deviates from alg′ *starting from* step n_0.

Theorem 4.2. *Assume* `HardMax` *response function with fair tie-breaking:* $f_{\texttt{resp}}(0) = 1/2$. *Assume that* $\texttt{alg}_1$ *is* `BayesGreedy`*, and* $\texttt{alg}_2$ *deviates from* `BayesGreedy` *starting from some (local) step* $n_0 < T$. *Then all agents in global rounds* $t \geq n_0$ *select principal* 1.

`BayesGreedy` is a weakly dominant strategy in the competition game, and a unique Nash equilibrium. This is because `BayesGreedy` receives, in expectation, at least half of the agents before global round n_0, and all agents after that; both are the best possible against $\texttt{alg}_2$. Moreover, `BayesGreedy` guarantees at least $T/2$ agents in expectation, and any other strategy can receive strictly less, *e.g.,* if the opponent chooses `BayesGreedy`.

Likewise, consider any mixed Nash equilibrium. On the one hand, each principal can guarantee exactly half of the market share by mimicking the strategy of the opponent. On the other hand, if one principal's mixed strategy is not `BayesGreedy`, then the opponent can grab more than a half of the market share by switching to `BayesGreedy`. It follows that (`BayesGreedy`,`BayesGreedy`) is a unique Nash equilibrium, whether pure or mixed.

Corollary 4.3. `BayesGreedy` *is a weakly dominant strategy in the competition game. The game has a unique mixed Nash equilibrium: both principals choose* `BayesGreedy`.

The proof of Theorem 4.2 relies on two key lemmas: that deviating from `BayesGreedy` implies a strictly smaller Bayesian-expected reward, and that `HardMax` implies a "sudden-death" property: if one agent chooses principal 1 with certainty, so do all subsequent agents. We re-use both lemmas in later results, so we state them in sufficient generality.

Lemma 4.4. *Assume that* $\mathtt{alg}_1$ *is* BayesGreedy*, and* $\mathtt{alg}_2$ *deviates from* BayesGreedy *starting from some (local) step* $n_0 < T$*. Then* $\mathbb{E}[\mathtt{rew}_1(n_0)] > \mathbb{E}[\mathtt{rew}_2(n_0)]$*. The lemma holds for any response function* $f_{\mathtt{resp}}$ *(as it only considers the stand-alone performance of each algorithm).*

Lemma 4.5. *Consider* HardMax *response function with* $f_{\mathtt{resp}}(0) \geq \frac{1}{2}$*. Suppose* $\mathtt{alg}_1$ *is Bayesian-monotone, and* $\mathtt{PMR}_1(t_0) > \mathtt{PMR}_2(t_0)$ *for some global round* t_0*. Then* $\mathtt{PMR}_1(t) > \mathtt{PMR}_2(t)$ *for all subsequent rounds* t*.*

The sudden-death property in Lemma 4.5 holds because principal 1 appears at least as good or better to the next agent (by the Bayesian-monotonicity property), whereas principal 2 appears the same as before. Principal 2 needs new data in order to improve, and it does not receive new data unless the response function is randomized.

The remainder of the proof of Theorem 4.2 uses the conclusion of Lemma 4.4 to derive the precondition for Lemma 4.5, *i.e.,* goes from $\mathbb{E}[\mathtt{rew}_1(n_0)] > \mathbb{E}[\mathtt{rew}_2(n_0)]$ to $\mathtt{PMR}_1(n_0) > \mathtt{PMR}_2(n_0)$. The subtlety one needs to deal with is that the principal's "local" round corresponding to a given "global" round is a random quantity due to the random tie-breaking.

Biased tie-breaking. The HardMax model is very sensitive to tie-breaking between the principals. If ties are broken deterministically in favor of principal 1, this principal can get all agents no matter what the other principal does, simply by using StaticGreedy.

Theorem 4.6. *Assume* HardMax *response function with* $f_{\mathtt{resp}}(0) = 1$ *(ties are always broken in favor of principal* 1*). If* $\mathtt{alg}_1$ *is* StaticGreedy*, then all agents choose principal* 1*.*

Proof Sketch. Agent 1 chooses principal 1 because of the tie-breaking rule. Since StaticGreedy is trivially Bayesian-monotone, all the subsequent agents choose principal 1 by an induction argument similar to the one in the proof of Lemma 4.5. □

A more challenging scenario is when the tie-breaking is biased in favor of principal 1, but not deterministically so: $f_{\mathtt{resp}}(0) > \frac{1}{2}$. Then this principal also has a "winning strategy" no matter what the other principal does. Specifically, principal 1 can get all but the first few agents, under a mild assumption that BayesGreedy deviates from StaticGreedy.

Theorem 4.7. *Assume* HardMax *response function with* $f_{\mathtt{resp}}(0) > \frac{1}{2}$ (i.e., *tie-breaking is biased in favor of principal* 1*). Assume the prior* $\mathcal{P}$ *is such that* BayesGreedy *deviates from* StaticGreedy *starting from some step* n_0*. Suppose that principal* 1 *runs a Bayesian-monotone MAB algorithm that coincides with* BayesGreedy *in the first* n_0 *steps. Then all agents* $t \geq n_0$ *choose principal* 1*.*

Thus, Principal 1 can use BayesGreedy, or any other Bayesian-monotone MAB algorithm that coincides with BayesGreedy in the first few steps. The proof re-uses Lemmas 4.4 and 4.5, which do not rely on fair tie-breaking.

4.3 HardMax with random agents

Consider the HardMax&Random response model, *i.e.,* HardMax mixed with "random agents". Informally, we find that *a much better algorithm wins big*. In more detail, a principal with asymptotically better BIR wins by a large margin: after a "learning phase" of constant duration, all agents choose this principal with maximal possible probability $f_{\texttt{resp}}(1)$. For example, a principal with $\texttt{BIR}(n) \leq \tilde{O}(n^{-1/2})$ prevails over one with $\texttt{BIR}(n) \geq \Omega(n^{-1/3})$.

To state this result, we need to express a property that $\texttt{alg}_1$ eventually catches up and surpasses $\texttt{alg}_2$, even if initially it receives only a fraction of traffic. We assume that both algorithms run indefinitely and do not depend on the time horizon T; call such algorithms *T-oblivious*. In particular, their BIR at a given step does not depend on T. Then this property can be formalized as follows:

$$(\forall \epsilon > 0) \qquad \texttt{BIR}_1(\epsilon n) \,/\, \texttt{BIR}_2(n) \to 0. \tag{5}$$

In fact, a weaker version suffices: denoting $\epsilon_0 = f_{\texttt{resp}}(-1)$, for some constant n_0 we have

$$(\forall n \geq n_0) \qquad \texttt{BIR}_1\left(\nicefrac{1}{2}\, \epsilon_0\, n \right) \,/\, \texttt{BIR}_2(n) < \nicefrac{1}{2}. \tag{6}$$

If this holds, we say that $\texttt{alg}_1$ *BIR-dominates* $\texttt{alg}_2$ starting from (local) step n_0.

We also need a mild technical assumption that $\texttt{BIR}_2(\cdot)$ is not extremely small:

$$(\exists m_0\ \forall n \geq m_0) \qquad \texttt{BIR}_2(n) > 4\, e^{-\epsilon_0\ n/12}. \tag{7}$$

Theorem 4.8. *Fix a* HardMax&Random *response function* $f_{\texttt{resp}}$. *Suppose algorithms* $\texttt{alg}_1$, $\texttt{alg}_2$ *are Bayesian-monotone and T-oblivious, and (7) holds. If* $\texttt{alg}_1$ BIR-dominates $\texttt{alg}_2$ *starting from step n_0, then each agent $t \geq \max(n_0, m_0)$ chooses principal 1 with probability $f_{\texttt{resp}}(1) = 1 - \epsilon_0$ (which is the largest possible probability for this response function).*

To conclude that a (much) better algorithm prevails in equilibrium, we consider a version of the competition game in which the principals are restricted to choosing from a given set of MAB algorithms; the algorithms in this set are called *feasible*.

Corollary 4.9. *Fix a* HardMax&Random *response function* $f_{\texttt{resp}}$ *with fair tie-breaking:* $f_{\texttt{resp}}(0) = \nicefrac{1}{2}$. *Consider the competition game in which all feasible MAB algorithms are T-oblivious, Bayesian-monotone, and satisfy (7) for some fixed m_0. Suppose some feasible algorithm* alg *BIR-dominates all other feasible algorithms, starting from some local step n_0. Then, for any sufficiently large time horizon T,* alg *is a weakly dominant strategy for each principal, and* $(\texttt{alg}, \texttt{alg})$ *is a unique mixed Nash equilibrium.*

This corollary is geared towards a fairly realistic scenario when the principals choose among a small number of *types* of MAB algorithms (*e.g.,* Epsilon-Greedy vs. Thompson Sampling), rather than small tweaks within each type. We make no positive prediction when a few feasible algorithms are good, but no one dominates the others. Next we show that such positive predictions are essentially impossible.

Counterpoint: A little greedy goes a long way

Given any Bayesian-monotone MAB algorithm alg other than BayesGreedy, we design a modified algorithm which "mixes in" some greedy choices (and consequently learns at a slower rate), yet prevails over alg in the competition game. Thus, we have a counterpoint to "much better algorithms win": even under HardMax&Random, a slower-learning algorithm may lose in competition. A similar counterpoint to Corollary 4.9 states that non-greedy algorithms cannot be chosen in a pure

Nash equilibrium. This is consistent with Theorem 4.8, because the BIR-dominance condition required therein does not hold here.

The modified algorithm, called the *greedy modification* of alg with *mixing parameter* $p \in (0,1)$, is defined as follows. Suppose alg deviates from BayesGreedy starting from some (local) step n_0. The modified algorithm coincides with BayesGreedy for the first $n_0 - 1$ steps. In each step $n \geq n_0$, alg is invoked with probability $1-p$, and with the remaining probability p does the "greedy choice": chooses an action with the largest posterior mean reward given the current information collected by alg. The data from the "greedy choice" steps are not recorded.[9] This completes the specification.

We find that the greedy modification prevails in competition if p is small enough. We focus on *symmetric* response functions: ones with $f(x) + f(-x) = 1$ for any $x \in [0,1]$.

Theorem 4.10. *Consider a symmetric* HardMax&Random *response function* $f_{\texttt{resp}}$. *Suppose* $\texttt{alg}_1$ *is Bayesian-monotone, and deviates from* BayesGreedy *starting from some step* n_0. *Let* $\texttt{alg}_2$ *be the greedy modification of* $\texttt{alg}_1$ *with mixing parameter* $p > 0$ *such that* $(1-\epsilon_0)(1-p) > \epsilon_0$, *where* $\epsilon_0 = f_{\texttt{resp}}(-1)$ *is the baseline selection probability. Then each agent* $t \geq n_0$ *chooses principal* 2 *with probability* $1 - \epsilon_0$ *(which is the largest possible).*

Moreover, the greedy modification preserves Bayesian-monotonicity:

Lemma 4.11. *The greedy modification of any Bayesian-monotone algorithm is Bayesian-monotone, for any mixing parameter.*

Thus, the greedy modification is a pure strategy in the competition game restricted to Bayesian-monotone MAB algorithms, and it is beneficial in competition. Consider a pure Nash equilibrium of this game. If one principal chooses a non-greedy algorithm, then the opponent could guarantee strictly more than half of the market share via the greedy deviation. This is a contradiction, because the first principal can always guarantee exactly half of the market share by mimicking the opponent. Therefore, both principals must choose BayesGreedy.

Corollary 4.12. *Fix a symmetric* HardMax&Random *response function* $f_{\texttt{resp}}$. *Consider the competition game in which algorithms are feasible if and only if they are Bayesian-monotone. Then:*

*(a) the only possible pure Nash equilibrium is (*BayesGreedy, BayesGreedy*).*

(b) If BayesGreedy *satisfies* $\texttt{BIR}(n) \cdot n^{\gamma} \to \infty$ *for some* $\gamma > 1/2$, *then there are no pure Nash equilibria, for any sufficiently large time horizon* T.

Let us clarify part (b) of the corollary. The stated precondition is a fairly mild form of inefficiency. A more typical scenario for BayesGreedy is a learning failure, with positive-constant BIR in each round. *E.g.,* this happens when the Bayesian prior $\mathcal{P}_{\texttt{mean}}$ is independent across arms, and the prior on each μ_a has a strictly positive density on $[0,1]$ (see Corollary 11.9 in Slivkins (2019)). (Even) if the precondition holds, BayesGreedy is dominated by any Bayesian-monotone, T-oblivious algorithm with $\texttt{BIR}(n) = \tilde{O}(t^{-1/2})$. One such algorithm is Thompson Sampling (Sellke and Slivkins, 2021). By Theorem 4.8, it would be a profitable deviation from the (BayesGreedy, BayesGreedy) profile if the time horizon T is large enough.

[9] In other words: the algorithm proceeds as if the "greedy choice" steps have never happened. While it is usually more efficient to consider all available data, this modification simplifies analysis.

Remark 4.13. While Corollary 4.12 does not restrict *mixed* Nash equilibria, this equilibrium concept appears somewhat dubious for the competition game. Indeed, the underlying premise for mixed Nash equilibria is to allow each principal to respond to the competitor's mixed strategy. However, one could argue that the principal could instead respond to the competitor's *pure* strategy directly, because the latter is revealed via commitment.

Finally, let us argue how the greedy modification can degrade the algorithm in some precise sense. Formulating this claim precisely is somewhat subtle, as per below. We also prove Lemma 4.11 as a by-product of this analysis.

Claim 4.14. *Let* $\texttt{alg}_1$ *be any Bayesian-monotone algorithm, and let* $\texttt{alg}_2$ *be its greedy modification, with an arbitrary mixing parameter* $p \in (0,1)$*. Let* $\texttt{alg}_{gr}$ *be a hypothetical algorithm which at each step* n *outputs the "Bayesian-greedy choice" based on the data collected by* $\texttt{alg}_1$ *in the first* $n-1$ *steps. Let* $\texttt{BIR}^{gr}(n)$ *be the* $\texttt{BIR}$ *of this algorithm. Suppose there exists a convex, decreasing function* $f : \mathbb{R}^+ \to [0,1]$ *and parameter* $q \in (1-p, 1)$ *such that for any sufficiently large step* n *it holds that*

$$\texttt{BIR}^{gr}(n) \geq f(n) > \texttt{BIR}_1(n/q). \tag{8}$$

Then for any sufficiently large step n *we have* $\texttt{BIR}_1(n) > \texttt{BIR}_2(n)$.

Proof. Let M_n be the number of times $\texttt{alg}_1$ is invoked in the first n steps of $\texttt{alg}_2$. Let $\texttt{reg}_2(n) = n \cdot \max_a \mu_a - \texttt{rew}_2(n)$ be the (frequentist) instantaneous regret of $\texttt{alg}_2$. Then

$$\mathbb{E}\left[\, \texttt{reg}_2(n) \mid M_n = m \,\right] = (1-p) \cdot \texttt{BIR}_1(m) + p \cdot \texttt{BIR}^{gr}(m).$$

$$\texttt{BIR}_2(n) = \mathbb{E}\left[\, (1-p) \cdot \texttt{BIR}_1(M_n) + p \cdot \texttt{BIR}^{gr}(M_n) \,\right]. \tag{9}$$

Using (8),(9) and Jensen's inequality, for any $q \in (1-p, 1)$ and any large enough step n we have

$$\texttt{BIR}_2(n) \geq \mathbb{E}\left[\, \texttt{BIR}^{gr}(M_n) \,\right] \geq \mathbb{E}\left[\, f(M_n) \,\right] \geq f\left(\, \mathbb{E}[M_n] \,\right) > f(qn) > \texttt{BIR}_1(n). \qquad \square$$

Lemma 4.11 follows from Eq. (9). Indeed, the lemma asserts that $\texttt{BIR}_2(n)$ is non-decreasing, which follows from Eq. (9) because both $\texttt{alg}_1$ and $\texttt{alg}_{\texttt{gr}}$ are Bayesian-monotone. The latter follows from the "informational monotonicity" of the "greedy step": it can only get better with more information, see Lemma B.1.

4.4 `SoftMax` response function

For the `SoftMax` model, we derive a "better algorithm wins" result under a much weaker version of BIR-dominance. This is the most technical part of the paper.

We start with a formal definition of `SoftMax`:

Definition 4.15. A response function $f_{\texttt{resp}}$ is `SoftMax` if the following conditions hold:

- $f_{\texttt{resp}}(\cdot)$ is bounded away from 0 and 1: $f_{\texttt{resp}}(\cdot) \in [\epsilon, 1-\epsilon]$ for some $\epsilon \in (0, \frac{1}{2})$,
- fair tie-breaking: $f_{\texttt{resp}}(0) = \frac{1}{2}$.
- the response function $f_{\texttt{resp}}(\cdot)$ is "smooth" around 0:

$$\exists \text{ constants } \delta_0, c_0, c_0' > 0 \qquad \forall x \in [-\delta_0, \delta_0] \qquad c_0 \leq f'_{\texttt{resp}}(x) \leq c_0'. \tag{10}$$

Remark 4.16. This definition is fruitful when c_0 and c'_0 are close to $\frac{1}{2}$. Throughout, we assume that $\texttt{alg}_1$ is better than $\texttt{alg}_2$, and obtain results parameterized by c_0. By symmetry, one could assume that $\texttt{alg}_2$ is better than $\texttt{alg}_1$, and obtain similar results in terms of c'_0.

For the sake of intuition, let us derive a version of Theorem 4.8, with the same assumptions and a similar proof. The conclusion is much weaker, though: we can only guarantee that each agent $t \geq n_0$ chooses principal 1 with probability slightly larger than $\frac{1}{2}$. This is essentially unavoidable in a typical case when both algorithms satisfy $\texttt{BIR}(n) \to 0$.

Theorem 4.17. *Assume* SoftMax *response function. Suppose algorithms* $\texttt{alg}_1$, $\texttt{alg}_2$ *satisfy the assumptions in Theorem 4.8. Then each agent* $t \geq n_0$ *chooses principal* 1 *with probability*

$$\Pr[i_t = 1] \geq \tfrac{1}{2} + \tfrac{c_0}{4}\, \texttt{BIR}_2(t). \tag{11}$$

To prove this theorem, we follow the steps in the proof of Theorem 4.8 to derive $\texttt{PMR}_1(t) - \texttt{PMR}_2(t) \geq \texttt{BIR}_2(t)/2 - \exp(-\epsilon_0\, t/12)$. This is at least $\texttt{BIR}_2(t)/4$ by (7). Then Eq. (11) follows by the smoothness condition (10).

Let us relax the notion of BIR-dominance so that the constant multiplicative factors in (6), namely $\epsilon_0/2$ and $\frac{1}{2}$, are replaced by constants that can be arbitrarily close to 1.

Definition 4.18. Let $\texttt{alg}_1$, $\texttt{alg}_2$ be T-oblivious MAB algorithms. Say that $\texttt{alg}_1$ *weakly BIR-dominates* $\texttt{alg}_2$ if there are absolute constants $\beta_0, \alpha_0 \in (0, 1/2)$ and $n_0 \in \mathbb{N}$ such that

$$(\forall n \geq n_0) \quad \frac{\texttt{BIR}_1((1-\beta_0)\, n)}{\texttt{BIR}_2(n)} < 1 - \alpha_0. \tag{12}$$

Now we are ready to state the main result for SoftMax:

Theorem 4.19. *Assume the* SoftMax *response function. Suppose algorithms* $\texttt{alg}_1$, $\texttt{alg}_2$ *are Bayesian-monotone and* T*-oblivious, and* $\texttt{alg}_1$ *weakly-BIR-dominates* $\texttt{alg}_2$*. Posit mild technical assumptions:* $\texttt{BIR}_1(n) \to 0$ *and that* $\texttt{BIR}_2$ *cannot be extremely small, namely:*

$$(\exists m_0\ \forall n \geq m_0) \qquad \texttt{BIR}_2(n) \geq \tfrac{4}{\alpha_0}\ \exp\left(-\tfrac{1}{12}\, n\, \min\{\epsilon_0, 1/8\}\right). \tag{13}$$

Then there exists some t_0 *such that each agent* $t \geq t_0$ *chooses principal* 1 *with probability*

$$\Pr[i_t = 1] \geq \tfrac{1}{2} + \tfrac{1}{4}\, c_0\, \alpha_0\, \texttt{BIR}_2(t). \tag{14}$$

Proof Sketch. The main idea is that even though $\texttt{alg}_1$ may have a slower rate of learning in the beginning, it will gradually catch up and surpass $\texttt{alg}_2$. We distinguish two phases. In the first phase, $\texttt{alg}_1$ receives a random agent with probability at least $f_{\texttt{resp}}(-1) = \epsilon_0$ in each round. Since $\texttt{BIR}_1$ tends to 0, the difference in BIRs between the two algorithms is also diminishing. Due to the SoftMax response function, $\texttt{alg}_1$ attracts each agent with probability at least $1/2 - O(\beta_0)$ after a sufficient number of rounds. Then the game enters the second phase: both algorithms receive agents at a rate close to $\frac{1}{2}$, and the fractions of agents received by both algorithms — $n_1(t)/t$ and $n_2(t)/t$ — also converge to $\frac{1}{2}$. At the end of the second phase and in each global round afterwards, the counts $n_1(t)$ and $n_2(t)$ satisfy the weak BIR-dominance condition, in the sense that they both are larger than n_0 and $n_1(t) \geq (1-\beta_0)\, n_2(t)$. At this point, $\texttt{alg}_1$ actually has smaller BIR, which reflected in the PMRs eventually. Accordingly, from then on $\texttt{alg}_1$ attracts agents at a rate slightly larger than $\frac{1}{2}$. We prove that the "bump" over $\frac{1}{2}$ is at least on the order of $\texttt{BIR}_2(t)$. □

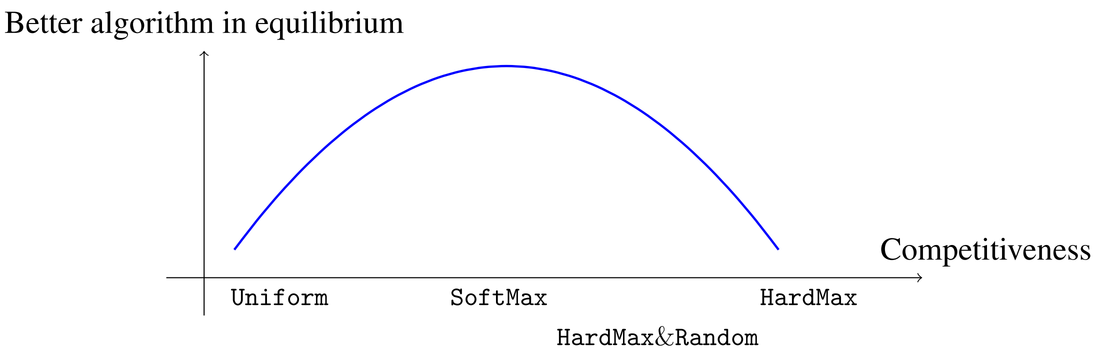

Figure 2: The stylized inverted-U relationship in the "main story".

It follows that a weakly-BIR-dominating algorithm prevails in equilibrium.

Corollary 4.20. *Consider the competition game in which all feasible algorithms are Bayesian-monotone, T-oblivious, and satisfy $\sum_{m=1}^{n} \texttt{BIR}(m) \rightarrow_n \infty$.*[10] *Suppose some feasible algorithm* `alg` *weakly-BIR-dominates all others. Then, for any sufficiently large time horizon T,* `alg` *is a weakly dominant strategy for each principal, and* $(\texttt{alg}, \texttt{alg})$ *is a unique mixed Nash equilibrium.*

4.5 Economic implications

We frame our contributions in terms of the relationship between *competitiveness* (as expressed by the "hardness" of the response function $f_{\texttt{resp}}$), and adoption of better algorithms.

Main story. Our main story concerns the finite competition game between the two principals where one allowed algorithm `alg` is "better" than the others. We track whether and when `alg` is chosen in an equilibrium. We vary *competitiveness* by changing the response function from `HardMax` (very competitive environment) to `HardMax&Random` to `SoftMax` (less competition). Our conclusions are as follows:

- Under `HardMax`, no innovation: `BayesGreedy` is chosen over `alg`.
- Under `HardMax&Random`, some innovation: `alg` is chosen as long as it BIR-dominates.
- Under `SoftMax`, more innovation: `alg` is chosen as long as it weakly-BIR-dominates.

These conclusions follow from Corollaries 4.3, 4.9 and 4.20, respectively. Recall that weak-BIR-dominance is a weaker condition, so that a better algorithm is chosen in a broader range of scenarios. We also consider the uniform choice between the principals, which entails the least amount of competition and (when principals optimize market share) provides no incentives to innovate.[11] Thus, we have an inverted-U relationship, see Figure 2.

Secondary story. Let us zoom in on the symmetric `HardMax&Random` model. Competitiveness within this model are controlled by the baseline probability $\epsilon_0 = f_{\texttt{resp}}(-1)$, which varies smoothly

[10]This is a very mild non-degeneracy condition, see Appendix A for background.

[11]However, if principals' utility is aligned with agents' welfare, then a monopolist principal is incentivized to choose the best possible MAB algorithm (namely, to minimize cumulative Bayesian regret $\texttt{BReg}(T)$). Accordingly, monopoly would result in better social welfare than competition, as the latter is likely to split the market and cause each principal to learn more slowly. This is a well-known effect of economies of scale.

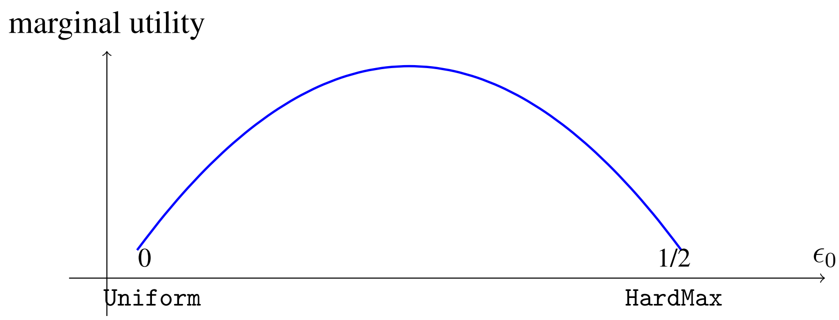

Figure 3: The stylized inverted-U relationship from the "secondary story"

between the two extremes: `HardMax` ($\epsilon_0 = 0$, tough competition) and the uniform choice ($\epsilon_0 = \frac{1}{2}$, no competition). Principals' utility is the number of agents.

We consider the marginal utility of switching to a better algorithm. Suppose initially both principals use some algorithm `alg`, and principal 1 ponders switching to another algorithm `alg`' which BIR-dominates `alg`. What is the marginal utility ΔU of this switch?

- if $\epsilon_0 = 0$ then ΔU can be negative if `alg` is `BayesGreedy`.
- if ϵ_0 is near 0 then only a small ΔU can be guaranteed, as it may take a long time for $\texttt{alg}'$ to "catch up" with `alg`, and hence less time to reap the benefits.
- if ϵ_0 is medium-range, then ΔU is large, as $\texttt{alg}'$ learns fast and gets most agents.
- if ϵ_0 is near $\frac{1}{2}$, the algorithm matters less, so ΔU is small.

These findings can also be organized as an inverted-U relationship, see Figure 3.

4.6 Extensions

Our theoretical results can be extended beyond the basic model in Section 3.

Reward-dependent utility. Except for Corollary 4.20, our results allow a more general notion of principal's utility that can depend on both the market share and agents' rewards. Namely, each principal i collects $U_i(r_t)$ units of utility in each global round t when she is chosen (and 0 otherwise), where $U_i(\cdot)$ is some fixed non-decreasing function with $U_i(0) > 0$.

Time-discounted utility. Theorem 4.2 and Corollary 4.3 holds under a more general model which allows time-discounting: namely, the utility of each principal i in each global round t is $U_{i,t}(r_t)$ if this principal is chosen, and 0 otherwise, where $U_{i,t}(\cdot)$ is an arbitrary non-decreasing function with $U_{i,t}(0) > 0$.

Arbitrary reward distributions. Bernoulli rewards can be extended to arbitrary reward distributions. For each arm $a \in A$ there is a parametric family $\psi_a(\cdot)$ of reward distributions, parameterized by the mean reward. Whenever arm a is chosen, the reward is drawn independently from distribution $\psi_a(\mu_a)$. The prior $\mathcal{P}_{\texttt{mean}}$ and the distributions $(\psi_a(\cdot) \colon a \in A)$ constitute the (full) Bayesian prior on rewards.

Beliefs. Instead of knowing the principals' algorithms $(\texttt{alg}_1, \texttt{alg}_2)$, the Bayesian prior $\mathcal{P}_{\texttt{mean}}$, and the response function $f_{\texttt{resp}}$, agents could have beliefs on these objects that need not be correct. If agents have common "point beliefs" on these objects, then all our results carry over with respect to these beliefs.

Limited non-stationarity in $f_{\texttt{resp}}$. Different agents can have different response functions. For HardMax&Random, our results carry over if each agent t has a HardMax&Random response function $f_{\texttt{resp}}$ with parameter $\epsilon_t \geq \epsilon_0$. For SoftMax, different agents can have different response functions that satisfy Definition 4.15 with the same parameters.

MAB extensions. Our results carry over, with little or no modification of the proofs, to much more general versions of MAB, as long as it satisfies the i.i.d. property. In each round, an algorithm can see a *context* before choosing an action (as in *contextual bandits*) and/or additional feedback other than the reward after the reward is chosen (as in, *e.g., semi-bandits*), as long as the contexts are drawn from a fixed distribution, and the (reward, feedback) pair is drawn from a fixed distribution that depends only on the context and the chosen action. The Bayesian prior $\mathcal{P}$ needs to be a more complicated object, to make sure that PMR and BIR are well-defined. Mean rewards may also have a known structure, such as Lipschitzness, convexity, or linearity; such structure can be incorporated via $\mathcal{P}$. All these extensions have been studied extensively in the literature on MAB, and account for a substantial segment thereof; see (Slivkins, 2019; Lattimore and Szepesvári, 2020) for background.

BIR **can depend on T.** Many MAB algorithms are parameterized by the time horizon T, and their regret bounds include $\operatorname{polylog}(T)$. In particular, a typical regret bound for BIR is

$$\texttt{BIR}(n \mid T) \leq \operatorname{polylog}(T) \cdot n^{-\gamma} \quad \text{for some } \gamma \in (0, \tfrac{1}{2}]. \tag{15}$$

We write $\texttt{BIR}(n \mid T)$ to emphasize the dependence on T. Accordingly, BIR-dominance can be redefined: there exists a number T_0 and a function $n_0(T) \in \operatorname{polylog}(T)$ such that

$$(\forall T \geq T_0,\ n \geq n_0(T)) \quad \texttt{BIR}_1(\epsilon_0 n/2 \mid T) \,/\, \texttt{BIR}_2(n \mid T) < 1/2. \tag{16}$$

Weak BIR-dominance extends similarly. Theorem 4.8 and 4.17 easily carry over.

5 Numerical simulations: the reputation-choice model

In this section we present our numerical simulations. As discussed in the Introduction, we focus on the reputation-choice model, whereby each agent chooses the firm with a maximal reputation score, modeled as a sliding window average of its rewards. While we experiment with various MAB instances and parameter settings, we only report on selected, representative experiments. Additional plots and tables are provided in Appendix E. Unless noted otherwise, our findings are based on and consistent with all these experiments.

5.1 Experiment setup

Challenges. An "atomic experiment" is a competition game between a given pair of bandit algorithms, in a given competition model, on a given multi-armed bandit problem (and each such experiment is run many times to reduce variance). Accordingly, we have a three-dimensional space of atomic experiments one needs to run and interpret: {pairs of algorithms} x {competition models} x {bandit problems}, and we are looking for findings that are consistent across this entire space. It is essential to keep each of the three dimensions small yet representative. In particular, we need to capture a huge variety of bandit algorithms and bandit instances with only a few representative examples. Further, we need a succinct and informative summarization of results within one atomic experiment and across multiple experiments (*e.g.,* see Table 1).

Competition model. All experiments use `HardMax` response function (without mentioning it), except Section 5.5 where we use `HardMax&Random` agents. In some of our experiments, one firm is the "incumbent" who enters the market before the other ("late entrant"), and therefore enjoys a *first-mover advantage*. Formally, the incumbent enjoys additional X rounds of the "warm start". We treat X as an exogenous element of the model, and study the consequences for a fixed X.

MAB algorithms. In abstract terms, we posit three types of technology, from "low" to "medium" to "high". Concretely, we consider three essential classes of bandit algorithms: ones that never explicitly explore (*greedy algorithms*), ones that explore without looking at the data (*exploration-separating algorithms*), and ones where exploration gradually zooms in on the best arm (*adaptive-exploration algorithms*). In the absence of competition, these classes are fairly well-understood: greedy algorithms are terrible for a wide variety of problem instances, exploration-separated algorithms learn at a reasonable but mediocre rate across all problem instances, and adaptive-exploration algorithms are optimal in the worst case, and exponentially improve for "easy" problem instances (see Appendix A).

We look for qualitative differences between these three classes under competition. We take a representative algorithm from each class. Our pilot experiments indicate that our findings do not change substantially if other representative algorithms are chosen. We use `BayesGreedy` (`BG`) algorithm as in Section 4.1, `BayesEpsilonGreedy` (`BEG`) from the "exploration-separating" algorithms, and `ThompsonSampling` (`TS`) from the "adaptive-exploration" algorithms.[12] For ease of comparison, all three algorithms are parameterized with the same "fake" Bayesian prior: namely, the mean reward of each arm is drawn independently from a $\texttt{Beta}(1, 1)$ distribution. Recall that Beta priors with 0-1 rewards form a conjugate family, which allows for simple posterior updates.

MAB instances. We consider bandit problems with $K = 10$ arms and Bernoulli rewards. The *mean reward vector* $(\mu(a) : \; a \in A)$ is initially drawn from some distribution, termed *MAB instance*. We consider three MAB instances:

1. *Needle-In-Haystack*: one arm (the "needle") is chosen uniformly at random. This arm has mean reward $.7$, and the remaining ones have mean reward $.5$.

2. *Uniform instance*: the mean reward of each arm is drawn independently and uniformly from $[1/4, 3/4]$.

3. *Heavy-Tail instance*: the mean reward of each arm is drawn independently from $\texttt{Beta}(.6, .6)$ distribution (which is known to have substantial "tail probabilities").

We argue that these MAB instances are (somewhat) representative. Consider the "gap" between the best and the second-best arm, an essential parameter in the literature on MAB. The "gap" is fixed in Needle-in-Haystack, spread over a wide spectrum of values under the Uniform instance,

[12] In each round t, `ThompsonSampling` computes a Bayesian posterior on μ_a for each arm a and draws an independent sample $\tilde{\mu}_{a,t}$ from this posterior; it chooses an arm which maximizes $\tilde{\mu}_{a,t}$.

`BayesEpsilonGreedy` proceeds as follows. In each round, with probability ϵ it explores by choosing an arm from the full set of arms uniformly at random. With the remaining probability, it "exploits" by choosing an arm with maximal posterior mean reward given the current data. We use $\epsilon = 5\%$ throughout. Our pilot experiments show that choosing a different ϵ does not qualitatively change the results.

and is spread but focused on the large values under the Heavy-Tail instance. We also ran smaller experiments with versions of these instances, and achieved similar qualitative results.

Simulation details. For each MAB instance we draw $N = 1000$ mean reward vectors independently from the corresponding distribution. We use this same collection of mean reward vectors for all experiments with this MAB instance. For each mean reward vector we draw a table of realized rewards (*realization table*), and use this same table for all experiments on this mean reward vector. This ensures that differences in algorithm performance are solely due to differences in the algorithms in the different experimental settings.

More specifically, the realization table is a 0-1 matrix W with K columns which correspond to arms, and $T + T_{\max}$ rows, which correspond to rounds. Here $T_{\max}$ is the maximal duration of the "warm start" in our experiments, *i.e.,* the maximal value of $X + T_0$. For each arm a, each value $W(\cdot, a)$ is drawn independently from Bernoulli distribution with expectation $\mu(a)$. Then in each experiment, the reward of this arm in round t of the warm start is taken to be $W(t, a)$, and its reward in round t of the game is $W(T_{\max} + t, a)$.

For the reputation scores, we fix the sliding window size $M = 100$. We found that lower values induced too much random noise in the results, and increasing M further did not make a qualitative difference. Unless otherwise noted, we used $T = 2000$.

Terminology. A particular instance of the competition game is specified by the MAB instance and the game parameters, as described above. Recall that firms are interested in maximizing their expected market share at the end of the game. Thus, for a given instance of the game and a given firm, algorithm Alg1 *(weakly) dominates* algorithm Alg2 if Alg1 provides a larger (or equal) expected final market share than Alg2, no matter that the opponent does. An algorithm is a (weakly) dominant strategy for the firm if it (weakly) dominates the other two algorithms.

5.2 Performance in Isolation

We start with a pilot experiment in which we investigate each algorithm's performance "in isolation": in a stand-alone MAB problem without competition. We focus on reputation scores generated by each algorithm. We confirm that algorithms' performance is ordered as we'd expect: `ThompsonSampling` > `BayesEpsilonGreedy` > `BayesGreedy` for a sufficiently long time horizon. For each algorithm and each MAB instance, we compute the mean reputation score at each round, averaged over all mean reward vectors. We plot the *mean reputation trajectory*: how this score evolves over time. We also plot the trajectory for instantaneous rewards (*not* averaged over the previous time-periods), which provides a better view into algorithm's performance at a given time.[13] Figure 4 shows these trajectories for the Needle-in-Haystack instance; for other MAB instances the plots are similar. We summarize this finding as follows:

Finding 1. *The mean reputation trajectories and the instantaneous reward trajectories are arranged as predicted by prior work:* `ThompsonSampling` > `BayesEpsilonGreedy` > `BayesGreedy` *for a sufficiently long time horizon T.*

[13] For "instantaneous reward" at a given time t, we report the average (over all mean reward vectors) of the mean rewards at this time, instead of the average of the *realized* rewards, so as to decrease the noise.

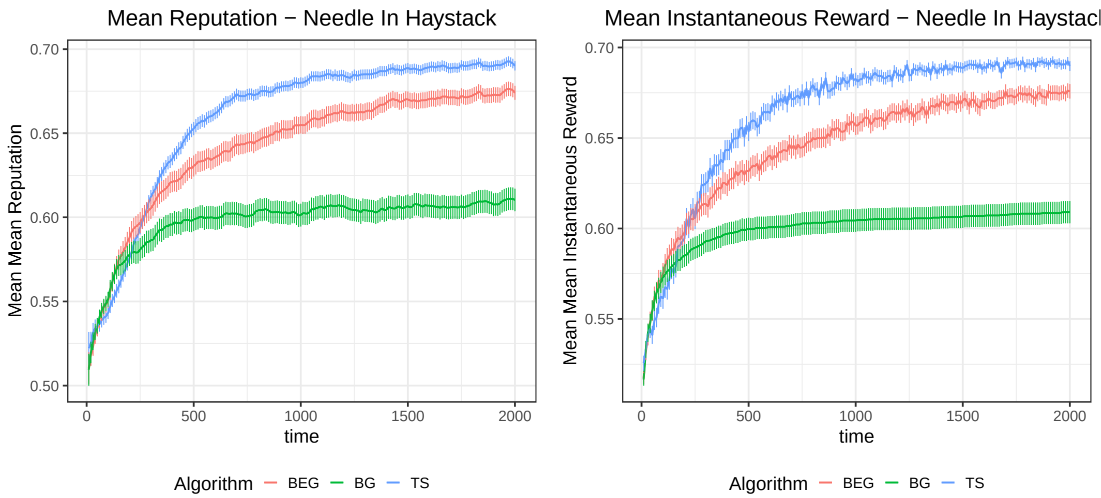

Figure 4: Mean reputation trajectory (left) and mean instantaneous reward trajectory (right) for Needle-in-Haystack. The shaded area shows 95% confidence intervals.

We also use Figure 4 to choose a reasonable time-horizon for the subsequent experiments, as $T = 2000$. The idea is, we want T to be large enough so that algorithms performance starts to plateau, but small enough such that algorithms are still learning.

The mean reputation trajectory is probably the most natural way to represent an algorithm's performance on a given MAB instance. However, we found that the outcomes of the competition game are better explained with a different "performance-in-isolation" statistic that is more directly connected to the game. Consider the performance of two algorithms, Alg1 and Alg2, "in isolation" on a particular MAB instance. The *relative reputation* of Alg1 (vs. Alg2) at a given time t is the fraction of mean reward vectors/realization tables for which Alg1 has a higher reputation score than Alg2. The intuition is that agent's selection in our model depends only on the comparison between the reputation scores.

This angle allows a more nuanced analysis of reputation costs vs. benefits under competition. Figure 5 (left) shows the relative reputation trajectory for `ThompsonSampling` vs `BayesGreedy` for the Uniform instance. The relative reputation is less than $\frac{1}{2}$ in the early rounds, meaning that `BayesGreedy` has a higher reputation score in a majority of the simulations, and more than $\frac{1}{2}$ later on. The reason is the exploration in `ThompsonSampling` leads to worse decisions initially, but allows for better decisions later. The time period when relative reputation vs. `BayesGreedy` dips below $\frac{1}{2}$ can be seen as an explanation for the competitive disadvantage of exploration. Such period also exists for the Heavy-Tail instance. However, it does not exist for the Needle-in-Haystack instance, see Figure 5.[14]

Finding 2. *Exploration can lead to relative reputation vs.* `BayesGreedy` *going below $\frac{1}{2}$ for some initial time period. This happens for some MAB instances but not for some others.*

Definition 5.1. For a particular MAB algorithm, a time period when relative reputation vs. `BayesGreedy` goes below $\frac{1}{2}$ is called *exploration disadvantage period*. An MAB instance is called *exploration-disadvantaged* if such period exists.

[14]We see two explanations for this: `ThompsonSampling` identifies the best arm faster for the Needle-in-Haystack instance, and there are no "very bad" arms to make exploration expensive in the near term.

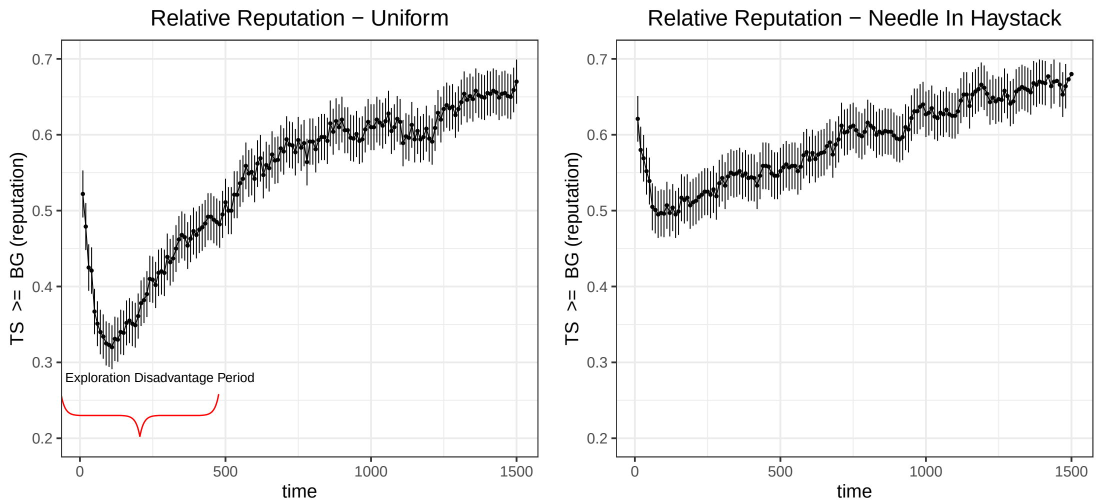

Figure 5: Relative reputation trajectory for `ThompsonSampling` vs `BayesGreedy`, on Uniform instance (left) and Needle-in-Haystack instance (right). Shaded area display 95% confidence intervals. The relative reputation at time t is the fraction of mean reward vectors for which, at time t, `ThompsonSampling` has a higher reputation score than `BayesGreedy`.

Note that Uniform and Heavy-tail instances are exploration-disadvantaged, but Needle-in-Haystack instance is not.

5.3 Competition vs. Better Algorithms

Our main experiments concern the duopoly game defined in Section 3. As the "intensity of competition" varies from monopoly to "incumbent" to "simultaneous entry" to "late entrant", we find a stylized inverted-U relationship as in Section 4.5. We look for equilibria in the duopoly game, where each firm's choices are limited to `BayesGreedy`, `BayesEpsilonGreedy` and `ThompsonSampling`. We do this for each "intensity level" and each MAB instance, and look for findings that are consistent across MAB instances. We break ties towards less advanced algorithms, as they tend to have lower adoption costs (Agarwal et al., 2017). `BayesGreedy` is then the dominant strategy under monopoly.

Simultaneous entry. The basic scenario is when both firms are competing from round 1. A crucial distinction is whether an MAB instance is exploration-disadvantaged:

Finding 3. *Under simultaneous entry:*

(a) (`BayesGreedy`,`BayesGreedy`) *is the unique pure-strategy Nash equilibrium for exploration-disadvantaged MAB instances with a sufficiently small "warm start".*

(b) This is not necessarily the case for MAB instances that are not exploration-disadvantaged. In particular, `ThompsonSampling` *is a weakly dominant strategy for Needle-in-Haystack.*

We investigate the firms' market shares when they choose different algorithms (otherwise, by symmetry both firms get half of the agents). We report the market shares for each instance in Table 1. We find that `BG` is a weakly dominant strategy for the Heavy-Tail and Uniform instances, as long as T_0 is sufficiently small. However, `ThompsonSampling` is a weakly dominant strategy for the Needle-in-Haystack instance. We find that for a sufficiently small T_0, `BayesGreedy` yields more than

	Heavy-Tail			Needle-in-Haystack			Uniform		
	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$
TS vs BG	**0.31** ±0.03	**0.72** ±0.02	**0.75** ±0.02	**0.68** ±0.03	**0.62** ±0.03	**0.65** ±0.03	**0.44** ±0.03	**0.52** ±0.02	**0.58** ±0.02
TS vs BEG	**0.3** ±0.03	**0.89** ±0.01	**0.9** ±0.01	**0.6** ±0.03	**0.52** ±0.03	**0.55** ±0.02	**0.41** ±0.03	**0.47** ±0.02	**0.55** ±0.02
BG vs BEG	**0.63** ±0.03	**0.6** ±0.02	**0.56** ±0.03	**0.42** ±0.03	**0.41** ±0.03	**0.39** ±0.02	**0.5** ±0.03	**0.46** ±0.02	**0.45** ±0.02

Table 1: **Simultaneous Entry, Market Share**. Each cell describes a game between two algorithms, call them Alg1 vs. Alg2, for a particular value of the warm start T_0. Each cell contains the market share of Alg 1: the average (in bold) and the 95% confidence band. The time horizon is $T = 2000$.

half the market against ThompsonSampling, but achieves similar market share vs. BayesGreedy and BayesEpsilonGreedy. By our tie-breaking rule, (BayesGreedy,BayesGreedy) is the only pure-strategy equilibrium.

We attribute the prevalence of BayesGreedy on exploration-disadvantaged MAB instances to its prevalence on the initial "exploration disadvantage period", as described in Section 5.2. Increasing the warm start length T_0 makes this period shorter: indeed, considering the relative reputation trajectory in Figure 5 (left), increasing T_0 effectively shifts the starting time point to the right. This is why it helps BayesGreedy if T_0 is small.

First-Mover. We turn our attention to the first-mover scenario. Recall that the incumbent firm enters the market and serves as a monopolist until the entrant firm enters at round X. We make X large enough, but still much smaller than the time horizon T. We find that the incumbent is incentivized to choose ThompsonSampling, in a strong sense:

Finding 4. *Under first-mover,* ThompsonSampling *is the dominant strategy for the incumbent. This holds across all MAB instances, if X is large enough.*

The simulation results for the Heavy-Tail MAB instance are reported in Table 2, for a particular $X = 200$. We see that ThompsonSampling is a dominant strategy for the incumbent. Similar tables for the other MAB instances and other values of X are reported in the supplement, with the same conclusion.

	TS	BEG	BG
TS	**0.003**±0.003	**0.083**±0.02	**0.17**±0.02
BEG	**0.045**±0.01	**0.25**±0.02	**0.23**±0.02
BG	**0.12**±0.02	**0.36**±0.03	**0.3**±0.02

Table 2: Market share of row player (entrant), 200 round head-start, Heavy-Tail Instance

BayesGreedy is a weakly dominant strategy for the entrant, for Heavy-Tail instance in Table 2 and the Uniform instance, but not for the Needle-in-Haystack instance. We attribute this finding to exploration-disadvantaged property of these two MAB instance, for the same reasons as discussed above.

Finding 5. *Under first-mover,* BayesGreedy *is a weakly dominant strategy for the entrant for exploration-disadvantaged MAB instances.*

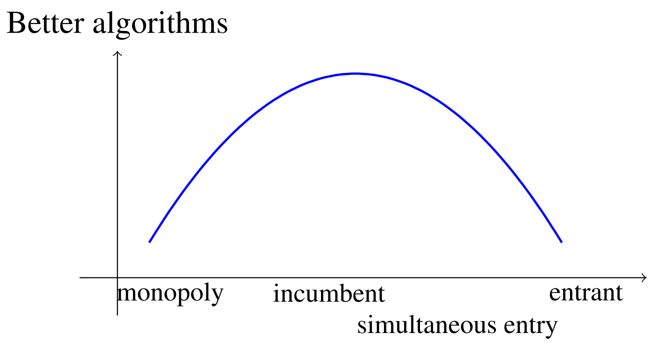

Figure 6: A stylized "inverted-U relationship" between strength of competition and "level of innovation".

Inverted-U relationship. We interpret our findings through the lens of the inverted-U relationship between the "intensity of competition" and the "quality of technology". The lowest level of competition is monopoly, when `BayesGreedy` wins out for the trivial reason of tie-breaking. The highest levels are simultaneous entry and "late entrant". We see that `BayesGreedy` is incentivized for exploration-disadvantaged MAB instances. In fact, incentives for `BayesGreedy` get stronger when the model transitions from simultaneous entry to "late entrant".[15] Finally, the middle level of competition, "incumbent" in the first-mover regime creates strong incentives for `ThompsonSampling`. In stylized form, this relationship is captured in Figure 6.[16]

Our intuition for why incumbency creates more incentives for exploration is as follows. During the period in which the incumbent is the only firm in the market, reputation consequences of exploration vanish. Instead, the firm wants to improve its performance as much as possible by the time competition starts. Essentially, the firm only faces a classical explore-exploit trade-off, and chooses algorithms that are best at optimizing this trade-off.

Death spiral effect. Further, we investigate the "death spiral" effect mentioned in the Introduction. Restated in terms of our model, the effect is that one firm attracts new customers at a lower rate than the other, and falls behind in terms of performance because the other firm has more customers to learn from, and this gets worse over time until (almost) all new customers go to the other firm. With this intuition in mind, we define *effective end of game* (`EoG`) for a particular mean reward vector and realization table, as the last round t such that the agents at this and previous round choose different firms. Indeed, the game, effectively, ends after this round. We interpret low `EoG` as a strong evidence of the "death spiral" effect. Focusing on the simultaneous entry scenario, we specify the `EoG` values in Table 3. We find that the `EoG` values are indeed small:

Finding 6. *Under simultaneous entry,* `EoG` *values tend to be much smaller than T.*

We also see that the `EoG` values tend to increase as the warm start T_0 increases. We conjecture this is because larger T_0 tends to be more beneficial for a better algorithm (as it tends to follow a

[15]For the Heavy-Tail instance, `BayesGreedy` goes from a weakly dominant strategy to a strictly dominant. For the Uniform instance, `BayesGreedy` goes from a Nash equilibrium strategy to a weakly dominant.

[16]We consider the monopoly scenario for comparison only. We just assume that a monopolist chooses the greedy algorithm, because it is easier to deploy in practice. Implicitly, users have no "outside option": the service provided is an improvement over not having it (and therefore the monopolist is not incentivized to deploy better learning algorithms). This is plausible with free ad-supported platforms such as Yelp or Google.

better learning curve). Indeed, we know that the `EoG` in this scenario typically occurs when a better algorithm loses, and helping it merely delays the loss.

	Heavy-Tail			Needle-in-Haystack			Uniform		
	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$
`TS vs BG`	68 (0)	560 (8.5)	610 (86.5)	180 (30)	380 (0)	550 (6.5)	260 (0)	780 (676.5)	880 (897.5)
`TS vs BEG`	37 (0)	430 (0)	540 (105)	150 (10)	460 (25)	780 (705)	230 (0)	830 (772)	980 (1038)
`BG vs BEG`	340 (110)	640 (393)	670 (425)	410 (8.5)	760 (666)	740 (646)	530 (101)	990 (1058)	1000 (1059)

Table 3: **Simultaneous Entry,** `EoG`. Each cell describes a game between two algorithms, call them Alg1 vs. Alg2, for a particular value of the warm start T_0. Each cell specifies the "effective end of game" (`EoG`): the average and the median (in brackets). The time horizon is $T = 2000$.

Welfare implications. We study the effects of competition on consumer welfare: the total reward collected by the users over time. Rather than welfare directly, we find it more lucid to consider *market regret*: $T \max_a \mu(a) - \sum_{t \in [T]} \mu(a_t)$, where a_t is the arm chosen by agent t. This is a standard performance measure in the literature on multi-armed bandits. Note that smaller regret means higher welfare.

We assume that both firms play their respective equilibrium strategies. As discussed previously, it is `BayesGreedy` in the monopoly scenario, and `BayesGreedy` for both firms for simultaneous entry (Finding 3). For the first-mover scenario, it is `ThompsonSampling` for the incumbent (Finding 4) and `BayesGreedy` for the entrant (Finding 5).

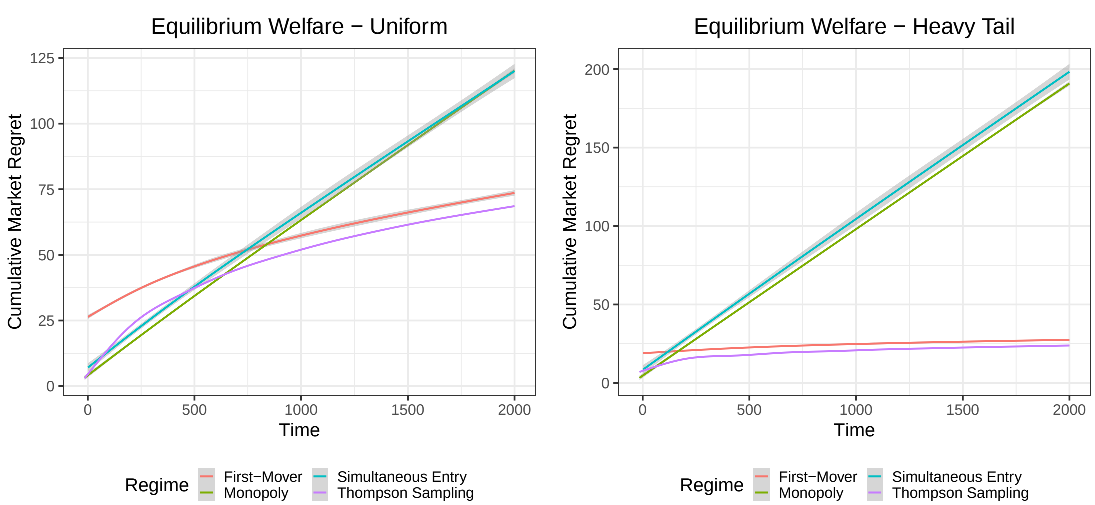

Figure 7: Smoothed welfare plots resulting from equilibrium strategies in the different market structures. Note that welfare at $t = 0$ incorporates the regret incurred during the incumbent and warm start periods. The Thompson Sampling trajectory displays the regret incurred by running Thompson Sampling in isolation on the given instances.

Figure 7 displays the market regret (averaged over multiple runs) under different levels of competition. Consumers are *better off* in the first-mover case than in the simultaneous entry case. Recall that under first-mover, the incumbent is incentivized to play `ThompsonSampling`. Moreover, we find that the welfare is close to that of having a single firm for all agents and running `ThompsonSampling`. We also observe that monopoly and simultaneous entry achieve similar welfare.

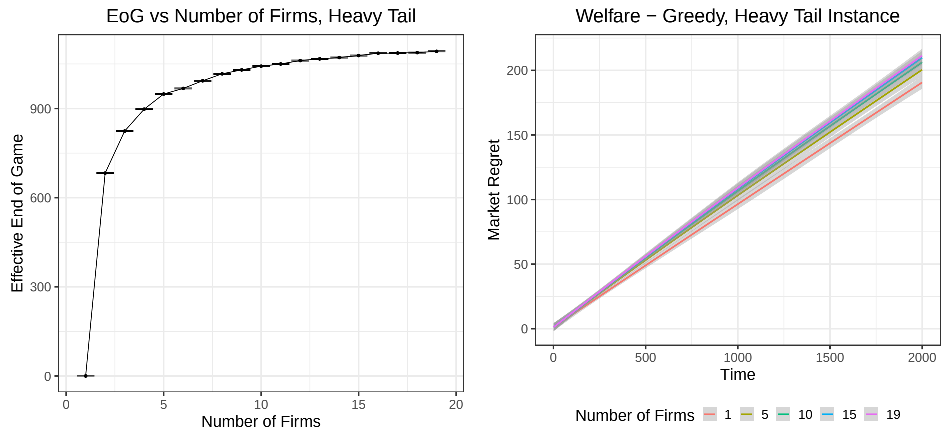

Figure 8: Average welfare and EoG as we increase the number of firms playing BayesGreedy

Finding 7. *In equilibrium, consumer welfare is (a) highest under first-mover, (b) similar for monopoly and simultaneous entry.*

Finding 7(b) is interesting because, in equilibrium, both firms play BayesGreedy in both settings, and one might conjecture that the welfare should increase with the number of firms playing BayesGreedy. Indeed, one run of BayesGreedy may get stuck on a bad arm. However, two firms independently playing BayesGreedy are less likely to get stuck simultaneously. If one firm gets stuck and the other does not, then the latter should attract most agents, leading to improved welfare.

To study this phenomenon further, we go beyond the duopoly setting to more than two firms playing BayesGreedy (and starting at the same time). Figure 8 reports the average welfare across these simulations. Welfare not only does not get better, *but is weakly worse* as we increase the number of firms.

Finding 8. *When all firms deploy* BayesGreedy*, and start at the same time, welfare is weakly decreasing as the number of firms increases.*

We track the average EoG in each of the simulations and notice that it *increases* with the number of firms. This observation also runs counter of the intuition that with more firms running BayesGreedy, one of them is more likely to "get lucky" and take over the market (which would cause EoG to *decrease* with the number of firms).

5.4 Data as a Barrier to Entry

In the first-mover regime, the incumbent can explore without incurring immediate reputational costs, and build up a high reputation before the entrant appears. Thus, the early entry gives the incumbent both a *data* advantage and a *reputational* advantage. We explore which of the two factors is more significant. Our findings provide a quantitative insight into the role of the classic "first mover advantage" phenomenon in the digital economy.

For a more succinct terminology, recall that the incumbent enjoys an extended warm start of $X + T_0$ rounds. Call the first X of these rounds the *monopoly period* (and the rest is the proper "warm start"). The rounds when both firms are competing for customers are called *competition period.*

We run two additional experiments to isolate the effects of the two advantages mentioned above. The *data-advantage experiment* focuses on the data advantage by, essentially, erasing the reputation advantage. Namely, the data from the monopoly period is not used in the computation of the incumbent's reputation score. Likewise, the *reputation-advantage experiment* erases the data advantage and focuses on the reputation advantage: namely, the incumbent's algorithm 'forgets' the data gathered during the monopoly period.

We find that either data or reputational advantage alone gives a substantial boost to the incumbent, compared to simultaneous entry duopoly. The results for the Heavy-Tail instance are presented in Table 4, in the same structure as Table 2. For the other two instances, the results are qualitatively similar.

	Reputation advantage (only)			Data advantage (only)		
	`TS`	`BEG`	`BG`	`TS`	`BEG`	`BG`
`TS`	**0.021**±0.009	**0.16**±0.02	**0.21** ±0.02	**0.0096**±0.006	**0.11**±0.02	**0.18**±0.02
`BEG`	**0.26**±0.03	**0.3**±0.02	**0.26**±0.02	**0.073**±0.01	**0.29**±0.02	**0.25**±0.02
`BG`	**0.34**±0.03	**0.4**±0.03	**0.33**±0.02	**0.15**±0.02	**0.39**±0.03	**0.33**±0.02

Table 4: Data advantage vs. reputation advantage experiment, on Heavy-Tail MAB instance. Each cell describes the duopoly game between the entrant's algorithm (the **row**) and the incumbent's algorithm (the **column**). The cell specifies the entrant's market share for the rounds in which hit was present: the average (in bold) and the 95% confidence interval. NB: smaller average is better for the incumbent.

We can quantitatively define the data (resp., reputation) advantage as the incumbent's market share in the competition period in the data-advantage (resp., reputation advantage) experiment, minus the said share under simultaneous entry duopoly, for the same pair of algorithms and the same problem instance. In this language, our findings are as follows.

Finding 9. *(a) Data advantage and reputation advantage alone are large, across all algorithms and MAB instances. (b) The data advantage is larger than the reputation advantage when the incumbent chooses* `ThompsonSampling`*. (c) The two advantages are similar in magnitude when the incumbent chooses* `BayesEpsilonGreedy` *or* `BayesGreedy`.

Our intuition for Finding 9(b) is as follows. Suppose the incumbent switches from `BayesGreedy` to `ThompsonSampling`. This switch allows the incumbent to explore actions more efficiently – collect better data in the same number of rounds – and therefore should benefit the data advantage. However, the same switch increases the reputation cost of exploration in the short run, which could weaken the reputation advantage.

5.5 Non-deterministic choice model (`HardMax&Random`)

Let us consider an extension in which the agents' response function (1) is no longer deterministic. We focus on `HardMax&Random` model, where each agent selects between the firms uniformly with probability $\epsilon \in (0, 1)$, and takes the firm with the higher reputation score with the remaining probability.

	Heavy-Tail (HMR with $\epsilon = .1$)			Heavy-Tail (HM)		
	TS vs BG	TS vs BEG	BG vs BEG	TS vs BG	TS vs BEG	BG vs BEG
$T = 2000$	**0.43** $\pm$ 0.02 Var: 0.15	**0.44** $\pm$ 0.02 Var: 0.15	**0.6** $\pm$ 0.02 Var: 0.1	**0.29** $\pm$ 0.03 Var: 0.2	**0.28** $\pm$ 0.03 Var: 0.19	**0.63** $\pm$ 0.03 Var: 0.18
$T = 5000$	**0.66** $\pm$ 0.01 Var: 0.056	**0.59** $\pm$ 0.02 Var: 0.092	**0.56** $\pm$ 0.02 Var: 0.098	**0.29** $\pm$ 0.03 Var: 0.2	**0.29** $\pm$ 0.03 Var: 0.2	**0.62** $\pm$ 0.03 Var: 0.19
$T = 10000$	**0.76** $\pm$ 0.01 Var: 0.026	**0.67** $\pm$ 0.02 Var: 0.067	**0.52** $\pm$ 0.02 Var: 0.11	**0.3** $\pm$ 0.03 Var: 0.21	**0.3** $\pm$ 0.03 Var: 0.2	**0.6** $\pm$ 0.03 Var: 0.2

Table 5: HardMax (HM) and HardMax&Random (HMR) choice models on the Heavy-Tail MAB instance. Each cell describes the market shares in a game between two algorithms, call them Alg1 vs. Alg2, at a particular value of t. Line 1 in the cell is the market share of Alg 1: the average (in bold) and the 95% confidence band. Line 2 specifies the variance of the market shares across the simulations. The results reported here are with $T_0 = 20$.

	Uniform (HMR with $\epsilon = .1$)			Needle-In-Haystack (HMR with $\epsilon = .1$)		
	TS vs BG	TS vs BEG	BG vs BEG	TS vs BG	TS vs BEG	BG vs BEG
$T = 2000$	**0.42** $\pm$ 0.02 Var: 0.13	**0.45** $\pm$ 0.02 Var: 0.13	**0.49** $\pm$ 0.02 Var: 0.093	**0.55** $\pm$ 0.02 Var: 0.15	**0.61** $\pm$ 0.02 Var: 0.13	**0.46** $\pm$ 0.02 Var: 0.12
$T = 5000$	**0.48** $\pm$ 0.02 Var: 0.089	**0.53** $\pm$ 0.02 Var: 0.098	**0.46** $\pm$ 0.02 Var: 0.072	**0.56** $\pm$ 0.02 Var: 0.13	**0.63** $\pm$ 0.02 Var: 0.12	**0.43** $\pm$ 0.02 Var: 0.11
$T = 10000$	**0.54** $\pm$ 0.01 Var: 0.055	**0.6** $\pm$ 0.02 Var: 0.073	**0.44** $\pm$ 0.02 Var: 0.064	**0.58** $\pm$ 0.02 Var: 0.083	**0.65** $\pm$ 0.02 Var: 0.096	**0.4** $\pm$ 0.02 Var: 0.1

Table 6: HardMax&Random (HMR) choice model for Uniform and Needle-In-Haystack MAB instances.

One can view HardMax&Random as a version of "warm start", where a firm receives some customers without competition, but these customers are dispersed throughout the game. The expected duration of this "dispersed warm start" is ϵT. If this quantity is large enough, we expect better algorithms to reach their long-term performance and prevail in competition. We confirm this intuition; we also find that this effect is negligible for smaller (but relevant) values of ϵ or T.

Finding 10. ThompsonSampling *is weakly dominant under* HardMax&Random*, if and only if* ϵT *is sufficiently large. Moreover,* HardMax&Random *leads to lower variance in market share, compared to* HardMax.

Table 5 shows the average market shares under HardMax vs HardMax&Random. In contrast to what happens under HardMax, TS becomes weakly dominant under HardMax&Random, as T gets sufficiently large. These findings hold across all problem instances, see Table 6 (with the same semantics as in Table 5).

However, it takes a significant amount of randomness and a relatively large time horizon for this effect to take place. Even with $T = 10000$ and $\epsilon = 0.1$ we see that BEG still outperforms BG on the Heavy-Tail MAB instance as well as that TS only starts to become weakly dominant at $T = 10000$ for the Uniform MAB instance.

5.6 Performance in Isolation, Revisited

We saw in Section 5.3 that mean reputation trajectories do not suffice to explain the outcomes under competition. Let us provide more evidence and intuition for this.

Mean reputation trajectories are so natural that one is tempted to conjecture that they determine the outcomes under competition. More specifically:

Conjecture 5.2. If one algorithm's mean reputation trajectory lies above another, perhaps after some initial time interval (*e.g.,* as in Figure 4), then the first algorithm prevails under competition, for a sufficiently large warm start T_0.

However, we find a more nuanced picture. For example, in Figure 1 we see that `BayesGreedy` attains a larger market share than `BayesEpsilonGreedy` even for large warm starts. We find that this also holds for $K = 3$ arms and longer time horizons, see the supplement for more details. We conclude that Conjecture 5.2 is false:

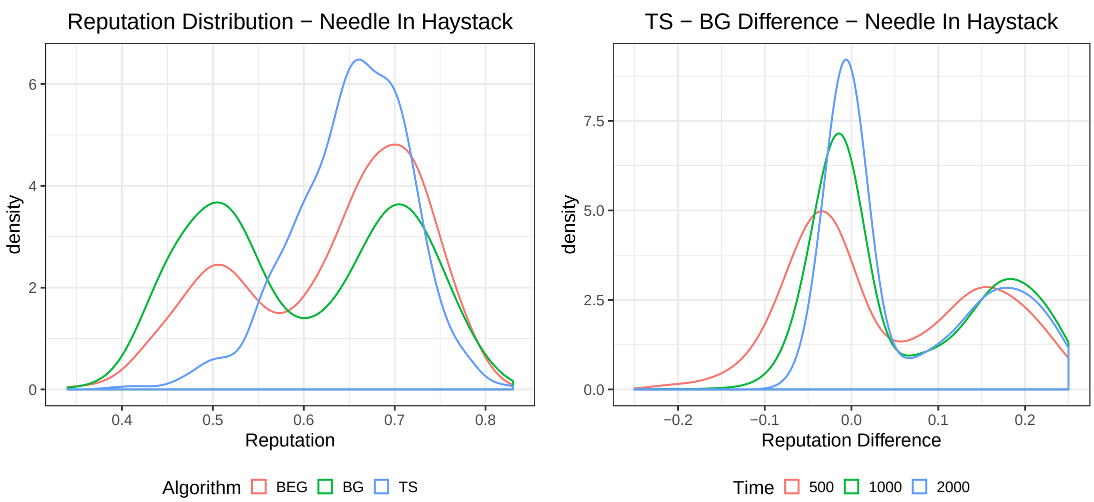

Figure 9: Needle-in-Haystack: reputation scores at $t = 500$ (left), reputation difference `ThompsonSampling` − `BayesGreedy` (right). Both are smoothed using a kernel density estimate.

Finding 11. *Mean reputation trajectories do not explain the outcomes under competition.*

To see what could go wrong with Conjecture 5.2, consider how an algorithm's reputation score is distributed at a particular time. That is, consider the empirical distribution of this score over different mean reward vectors.[17] For concreteness, consider the Needle-in-Haystack instance at time $t = 500$, plotted in Figure 9 (left). (The other MAB instances lead to a similar intuition.)

We see that the "naive" algorithms `BayesGreedy` and `BayesEpsilonGreedy` have a bi-modal reputation distribution, whereas `ThompsonSampling` does not. The reason is that for this MAB instance, `BayesGreedy` either finds the best arm and sticks to it, or gets stuck on the bad arms. In the former case `BayesGreedy` does slightly better than `ThompsonSampling`, and in the latter case it does substantially worse. However, the mean reputation trajectory fails to capture this complexity since it takes average over different mean reward vectors. This is inadequate for explaining the

[17] Recall that each mean reward vector in our experimental setup comes with a separate realization table.

outcome of the duopoly game, given that the latter is determined by a comparison between the firm's reputation scores.

To further this intuition, consider the difference in reputation scores (*reputation difference*) between `ThompsonSampling` and `BayesGreedy` on a particular mean reward vector. Let's plot the empirical distribution of the reputation difference (over the mean reward vectors) at a particular time point. Figure 9 (right) shows such plots for several time points. We observe that the distribution is skewed to the right, precisely due to the fact that `BayesGreedy` either does slightly better than `ThompsonSampling` or does substantially worse. So, the mean is not a good measure of the central tendency of this distribution.

6 Conclusions

We study the tension between exploration and competition. We consider a stylized duopoly model in which two firms face the same multi-armed bandit problem and compete for a stream of users. A firm makes progress on its learning problem only if it attracts users. We find that firms are incentivized to adopt a "greedy algorithm" which does no purposeful exploration and leads to welfare losses for users. We then consider two relaxations of competition: we soften users' decision rule and give one firm a first-mover advantage. Both relaxations induce firms to adopt "better" bandit algorithms, which benefits user welfare.

Our results have two economic interpretations. The first is that they can be framed in terms of the classic inverted-U relationship between innovation and competition, where *innovation* refers to the adoption of better bandit algorithms. Unlike other models in the literature, what prevents innovation is not its direct costs, but the short-term reputation consequences of exploration. The second interpretation concerns the role of data in the digital economy. We find that even a small initial disparity in data or reputation gets amplified under competition to a very substantial difference in the eventual market share. Thus, we endogenously obtain "network effects" without explicitly baking them into the model, and elucidate the role of data as a barrier to entry.

With this paper as a departure point, there are several exciting directions to explore. First, when the firms can set prices, they may be able to compensate early users for exploration, and potentially prevent the "death spiral" effects. (Our paper zeroes in on competition between free, ad-supported platforms that primarily compete on quality.) Second, horizontally differentiated user preferences may help explain how competition may encourage specialization, *i.e.,* how the firms may *learn to specialize* under competition. Third, while we focus on a stationary world, another well-motivated regime is "continuous learning", when exploration continuously counteracts change. The economic story would be about competition between relatively mature firms.[18]

References

Agarwal, A., S. Bird, M. Cozowicz, L. Hoang, J. Langford, S. Lee, J. Li, D. Melamed, G. Oshri, O. Ribas, S. Sen, and A. Slivkins (2017): "Making Contextual Decisions with Low Technical Debt," Techical report at `arxiv.org/abs/1606.03966`.

[18]One difficulty is that the "bandit model" becomes considerably more complicated: there are many reasonable ways to deal with a continuously changing world, starting from Slivkins and Upfal (2008), and the distinctions between better and worse algorithms are not as clear and well-established.

AGHION, P., N. BLOOM, R. BLUNDELL, R. GRIFFITH, AND P. HOWITT (2005): "Competition and innovation: An inverted-U relationship," *The Quarterly Journal of Economics*, 120, 701–728.

AGRAWAL, S. AND N. GOYAL (2012): "Analysis of Thompson Sampling for the multi-armed bandit problem," in *25nd Conf. on Learning Theory (COLT)*.

——— (2013): "Further Optimal Regret Bounds for Thompson Sampling," in *16th Intl. Conf. on Artificial Intelligence and Statistics (AISTATS)*, 99–107.

AKCIGIT, U. AND Q. LIU (2016): "The Role Of Information In Innovation And Competition," *Journal of the European Economic Association*, 14, 828–870.

ARIDOR, G., K. LIU, A. SLIVKINS, AND S. WU (2019): "The Perils of Exploration under Competition: A Computational Modeling Approach," in *20th ACM Conf. on Economics and Computation (ACM-EC)*.

AUER, P., N. CESA-BIANCHI, AND P. FISCHER (2002a): "Finite-time Analysis of the Multiarmed Bandit Problem." *Machine Learning*, 47, 235–256.

AUER, P., N. CESA-BIANCHI, Y. FREUND, AND R. E. SCHAPIRE (2002b): "The Nonstochastic Multiarmed Bandit Problem." *SIAM J. Comput.*, 32, 48–77, preliminary version in *36th IEEE FOCS*, 1995.

BABAIOFF, M., Y. SHARMA, AND A. SLIVKINS (2014): "Characterizing Truthful Multi-armed Bandit Mechanisms," *SIAM J. on Computing (SICOMP)*, 43, 194–230, preliminary version in *10th ACM EC*, 2009.

BAGWELL, K. (1990): "Informational product differentiation as a barrier to entry," *International Journal of Industrial Organization*, 8, 207–223.

BAJARI, P., V. CHERNOZHUKOV, A. HORTAÇSU, AND J. SUZUKI (2018): "The impact of big data on firm performance: An empirical investigation," Tech. rep., National Bureau of Economic Research.

BASTANI, H., M. BAYATI, AND K. KHOSRAVI (2021): "Mostly Exploration-Free Algorithms for Contextual Bandits," *Management Science*, 67, 1329–1349, working paper available on `arxiv.org` since 2017.

BEN-PORAT, O. AND M. TENNENHOLTZ (2019): "Regression equilibrium," in *Proceedings of the 2019 ACM Conference on Economics and Computation*, 173–191.

BERGEMANN, D. AND M. SAID (2011): "Dynamic Auctions: A Survey," in *Wiley Encyclopedia of Operations Research and Management Science, Vol. 2*, Wiley: New York, 1511–1522.

BERGEMANN, D. AND J. VÄLIMÄKI (1997): "Market diffusion with two-sided learning," *The RAND Journal of Economics*, 773–795.

——— (2000): "Experimentation in markets," *The Review of Economic Studies*, 67, 213–234.

BERGEMANN, D. AND J. VÄLIMÄKI (2006): "Bandit Problems," in *The New Palgrave Dictionary of Economics, 2nd ed.*, ed. by S. Durlauf and L. Blume, Macmillan Press.

BIETTI, A., A. AGARWAL, AND J. LANGFORD (2018): "A Contextual Bandit Bake-off," *CoRR arXiv:1802.04064*.

BOLTON, P. AND C. HARRIS (1999): "Strategic Experimentation," *Econometrica*, 67, 349–374.

BUBECK, S. AND N. CESA-BIANCHI (2012): "Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems," *Foundations and Trends in Machine Learning*, 5, 1–122, published with *Now Publishers* (Boston, MA, USA). Also available at `https://arxiv.org/abs/1204.5721`.

CABRAL, L. M. AND M. H. RIORDAN (1994): "The learning curve, market dominance, and predatory pricing," *Econometrica: Journal of the Econometric Society*, 1115–1140.

CESA-BIANCHI, N. AND G. LUGOSI (2006): *Prediction, learning, and games*, Cambridge, UK: Cambridge University Press.

CHE, Y.-K. AND J. HÖRNER (2018): "Recommender Systems as Mechanisms for Social Learning," *Quarterly Journal of Economics*, 133, 871–925, working paper since 2013, titled 'Optimal design for social learning'.

DASGUPTA, P. AND J. STIGLITZ (1988): "Learning-by-doing, market structure and industrial and trade policies," *Oxford Economic Papers*, 40, 246–268.

DE CORNIERE, A. AND G. TAYLOR (2020): "Data and Competition: a General Framework with Applications to Mergers, Market Structure, and Privacy Policy," .

DEVANUR, N. AND S. M. KAKADE (2009): "The Price of Truthfulness for Pay-Per-Click Auctions," in *10th ACM Conf. on Electronic Commerce (ACM-EC)*, 99–106.

EVEN-DAR, E., S. MANNOR, AND Y. MANSOUR (2006): "Action Elimination and Stopping Conditions for the Multi-Armed Bandit and Reinforcement Learning Problems," *J. of Machine Learning Research (JMLR)*, 7, 1079–1105.

FUDENBERG, D. AND J. TIROLE (1983): "Learning-by-doing and market performance," *The Bell Journal of Economics*, 522–530.

FURMAN, J., D. COYLE, A. FLETCHER, D. MCAULEY, AND P. MARSDEN (2019): "Unlocking digital competition," *Report of the digital competition expert panel*.

GHOSH, A. AND P. HUMMEL (2013): "Learning and incentives in user-generated content: multi-armed bandits with endogenous arms," in *Innovations in Theoretical Computer Science Conf. (ITCS)*, 233–246.

GITTINS, J., K. GLAZEBROOK, AND R. WEBER (2011): *Multi-Armed Bandit Allocation Indices*, Hoboken, NJ, USA: John Wiley & Sons, 2nd ed.

HAGIU, A. AND J. WRIGHT (2020): "Data-enabled learning, network effects and competitive advantage," Tech. rep., Working Paper.

HO, C.-J., A. SLIVKINS, AND J. W. VAUGHAN (2016): "Adaptive Contract Design for Crowdsourcing Markets: Bandit Algorithms for Repeated Principal-Agent Problems," *J. of Artificial Intelligence Research*, 55, 317–359, preliminary version appeared in *ACM EC 2014*.

HÖRNER, J. AND A. SKRZYPACZ (2017): "Learning, Experimentation, and Information Design," in *Advances in Economics and Econometrics: 11th World Congress*, ed. by B. Honoré, A. Pakes, M. Piazzesi, and L. Samuelson, Cambridge University Press, vol. 1, 63–98.

HOTELLING, H. (1929): "Stability in Competition," *The Economic Journal*, 39, 41–57.

IMMORLICA, N., A. T. KALAI, B. LUCIER, A. MOITRA, A. POSTLEWAITE, AND M. TENNENHOLTZ (2011): "Dueling algorithms," in *43rd ACM Symp. on Theory of Computing (STOC)*, 215–224.

JULLIEN, B. AND W. SAND-ZANTMAN (2019): "The economics of platforms: A theory guide for competition policy," *TSE Digital Center Policy Papers series*.

KANNAN, S., J. MORGENSTERN, A. ROTH, B. WAGGONER, AND Z. S. WU (2018): "A Smoothed Analysis of the Greedy Algorithm for the Linear Contextual Bandit Problem," in *Advances in Neural Information Processing Systems (NIPS)*.

KAUFMANN, E., N. KORDA, AND R. MUNOS (2012): "Thompson Sampling: An Asymptotically Optimal Finite-Time Analysis," in *23rd Intl. Conf. on Algorithmic Learning Theory (ALT)*, 199–213.

KELLER, G. AND S. RADY (2003): "Price dispersion and learning in a dynamic differentiated-goods duopoly," *RAND Journal of Economics*, 138–165.

KELLER, G., S. RADY, AND M. CRIPPS (2005): "Strategic Experimentation with Exponential Bandits," *Econometrica*, 73, 39–68.

KERIN, R. A., P. R. VARADARAJAN, AND R. A. PETERSON (1992): "First-mover advantage: A synthesis, conceptual framework, and research propositions," *The Journal of Marketing*, 33–52.

KLEINBERG, R. D., B. WAGGONER, AND E. G. WEYL (2016): "Descending Price Optimally Coordinates Search," in *17th ACM Conf. on Economics and Computation (ACM-EC)*, 23–24.

KREMER, I., Y. MANSOUR, AND M. PERRY (2014): "Implementing the "Wisdom of the Crowd"," *J. of Political Economy*, 122, 988–1012, preliminary version in *ACM EC 2013*.

LAI, L., H. JIANG, AND H. V. POOR (2008): "Medium access in cognitive radio networks: A competitive multi-armed bandit framework," in *42nd Asilomar Conference on Signals, Systems and Computers*.

LAI, T. L. AND H. ROBBINS (1985): "Asymptotically efficient Adaptive Allocation Rules," *Advances in Applied Mathematics*, 6, 4–22.

LAMBRECHT, A. AND C. E. TUCKER (2015): "Can Big Data protect a firm from competition?" .

LATTIMORE, T. AND C. SZEPESVÁRI (2020): *Bandit Algorithms*, Cambridge, UK: Cambridge University Press, versions available at `https://banditalgs.com/` since 2018.

LIU, K. AND Q. ZHAO (2010): "Distributed learning in multi-armed bandit with multiple players," *IEEE Trans. Signal Processing*, 58, 5667–5681.

LUCE, D. (1959): *Individual Choice Behavior: A Theoretical Analysis*, John Wiley.

MANSOUR, Y., A. SLIVKINS, V. SYRGKANIS, AND S. WU (2022): "Bayesian Exploration: Incentivizing Exploration in Bayesian Games," *Operations Research*, 70, preliminary version in *ACM EC 2016*.

MANSOUR, Y., A. SLIVKINS, AND S. WU (2018): "Competing Bandits: Learning under Competition," in *9th Innovations in Theoretical Computer Science Conf. (ITCS)*.

MATĚJKA, F. AND A. MCKAY (2015): "Rational inattention to discrete choices: A new foundation for the multinomial logit model," *American Economic Review*, 105, 272–98.

MILGROM, P. AND N. STOKEY (1982): "Information, trade and common knowledge," *J. of Economic Theory*, 26, 17–27.

MITZENMACHER, M. AND E. UPFAL (2005): *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*, Cambridge, UK: Cambridge University Press.

MOSTELLER, F. AND P. NOGEE (1951): "An experimental measurement of utility," *Journal of Political Economy*, 59, 371–404.

PERLOFF, J. M. AND S. C. SALOP (1985): "Equilibrium with Product Differentiation," *Review of Economic Studies*, LII, 107–120.

PRUFER, J. AND C. SCHOTTMÜLLER (2017): "Competing with big data," .

RAGHAVAN, M., A. SLIVKINS, J. W. VAUGHAN, AND Z. S. WU (2018): "The Externalities of Exploration and How Data Diversity Helps Exploitation," in *Conf. on Learning Theory (COLT)*, 1724–1738.

RUSSO, D., B. VAN ROY, A. KAZEROUNI, I. OSBAND, AND Z. WEN (2018): "A Tutorial on Thompson Sampling," *Foundations and Trends in Machine Learning*, 11, 1–96, published with *Now Publishers* (Boston, MA, USA). Also available at `https://arxiv.org/abs/1707.02038`.

RYSMAN, M. (2009): "The Economics of Two-Sided Markets," *J. of Economic Perspectives*, 23, 125–144.

SCHMALENSEE, R. (1982): "Product differentiation advantages of pioneering brands," *The American Economic Review*, 72, 349–365.

SCHUMPETER, J. (1942): *Capitalism, Socialism and Democracy*, Harper & Brothers.

SCOTT MORTON, F., P. BOUVIER, A. EZRACHI, B. JULLIEN, R. KATZ, G. KIMMELMAN, A. D. MELAMED, AND J. MORGENSTERN (2019): "Committee for the study of digital platforms: Market structure and antitrust subcommittee report," *Chicago: Stigler Center for the Study of the Economy and the State, University of Chicago Booth School of Business*.

SELLKE, M. AND A. SLIVKINS (2021): "The Price of Incentivizing Exploration: A Characterization via Thompson Sampling and Sample Complexity," in *22th ACM Conf. on Economics and Computation (ACM-EC)*, to appear in *Operations Research*, 2022.

SLIVKINS, A. (2019): "Introduction to Multi-Armed Bandits," *Foundations and Trends® in Machine Learning*, 12, 1–286, published with *Now Publishers* (Boston, MA, USA). Also available at `https://arxiv.org/abs/1904.07272`. Latest online revision: Jan 2022.

SLIVKINS, A. AND E. UPFAL (2008): "Adapting to a Changing Environment: the Brownian Restless Bandits," in *21st Conf. on Learning Theory (COLT)*, 343–354.

THOMPSON, W. R. (1933): "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples." *Biometrika*, 25, 285–294.

TIROLE, J. (1988): *The theory of industrial organization*, MIT press.

VARIAN, H. (2018): "Artificial intelligence, economics, and industrial organization," in *The Economics of Artificial Intelligence: An Agenda*, University of Chicago Press.

VIVES, X. (2008): "Innovation and Competitive Pressure," *J. of Industrial Economics*, 56.

WEYL, G. AND A. WHITE (2014): "Let the Right 'One' Win: Policy Lessons from the New Economics of Platforms," *Competition Policy International*, 12, 29–51.

A Background for non-specialists: multi-armed bandits

We present self-contained background on multi-armed bandits (MAB), to make the paper accessible to researchers who are not experts on MAB. More details can be found in books (Bubeck and Cesa-Bianchi, 2012; Slivkins, 2019; Lattimore and Szepesvári, 2020).

We focus on three algorithm classes, as in Section 5:

- *Greedy algorithms* that strive to maximize the reward for the next round given the available information. Thus, they always "exploit" and never explicitly "explore".
- *Exploration-separating algorithms* that separate exploration and exploitation: essentially, each round is dedicated to one and completely ignores the other.
- *Adaptive-exploration* algorithms that combine exploration and exploitation, and gradually sway the exploration choices towards more promising alternatives.

Below we discuss which algorithms are better than others, and what does it *mean* for one bandit algorithm to be better than another. This is a rather subtle issue, because some algorithms may be better for some problem instances and/or time intervals, and worse for some others. In particular, "better" algorithms are better in the long run, but could be worse initially.

While we list precise upper and lower bounds on the regret rates, the main goal is to illustrate how the three algorithm classes are separated from one another; the exact results are not essential for this paper. For ease of presentation, we use standard asymptotic notation from computer science: $O(f(t))$ and $\Omega(f(t))$ means at most (resp., at least) $f(n)$, up to constant factors, starting from large enough t. Likewise, $\tilde{O}(f(t))$ notation suppresses the $\mathrm{polylog}(t)$ factors.

Fundamentals. We are concerned with the following problem. There are T rounds and K *arms* to choose from. In each round $t \in [T]$, the algorithm chooses an arm and receives a reward $r_t \in [0, 1]$ for this arm, drawn from a fixed but unknown distribution.[19] The algorithm's goal is to maximize the total reward.

A standard performance measure is *regret*, defined as the difference in the total expected reward between the algorithm and the best arm. In a formula, regret is $T \cdot \max_{\text{arms } a} \mu_a - \mathbb{E}\left[\sum_{t \in [T]} r_t \right]$, where μ_a is the mean reward of arm a. Normalized by the best arm, regret allows to compare algorithms across different problem instances. The primary concern is the asymptotic growth rate of regret as a function of T.

The three classes of algorithms perform very differently in terms of regret: adaptive-exploration algorithms are by far the best, greedy algorithms are by far the worst, and exploration-separating ones are in the middle. Adaptive-exploration algorithms achieve optimal regret rates: $\tilde{O}(\sqrt{KT})$ for all problem instances, and simultaneously a vastly improved regret rate of $O(\frac{K}{\Delta} \log T)$ for all problem instances with gap $\geq \Delta$ ("easy" instances), without knowing the Δ in advance (Lai and Robbins, 1985; Auer et al., 2002a,b).[20] Exploration-separating algorithms can only achieve regret $\tilde{O}(T^{2/3})$ across all problem instances. They can achieve the "gap-dependent" regret rate stated

[19] All "negative" results (*i.e.,* lower bounds on regret) assume reward distributions with constant variance.

[20] The *gap* is the difference in mean reward between the best arm and the second-best arm.

above, but *only* if they know the Δ in advance, and with terrible regret $\Omega(\Delta T)$ for some other problem instances (Babaioff et al., 2014). Finally, the greedy algorithm is terrible on a wide variety of problem instances, in the sense that with constant probability it fails to try the best arm even once, and therefore suffers regret $\Omega(T)$ (see Chapter 11.2 in Slivkins, 2019).

The optimal regret rates are achieved by several adaptive-exploration algorithms, of which the most known are Thompson Sampling (Thompson, 1933; Russo et al., 2018),[21] UCB1 (Auer et al., 2002a), and Successive Elimination (Even-Dar et al., 2006).[22] These algorithms are very simple to describe. Focus on one round and consider the posterior distribution and/or the confidence interval on each arm's mean reward. Thompson Sampling draws a sample ("score") from each arm's posterior distribution, and picks an arm with the largest score. UCB1 picks an arm with the largest upper confidence bound. Successive Elimination eliminates an arm once it is worse than some other arm with high confidence, and chooses uniformly among the remaining arms.

Exploration-separating algorithms completely separate exploration and exploitation. Ahead of time, each round is either selected for exploration, in which case the distribution over arms does not depend on the observed data, or it is assigned to exploitation, in which case the data from this round is discarded. The simplest approach, called *Explore-First*, explores uniformly for a predetermined number of rounds, then chooses one arm for "exploitation" and uses it from then on. A more refined approach, called *Epsilon-Greedy*, explores uniformly in each round with a predetermined probability, and "exploits" with the remaining probability. Both algorithms, and the associated $\tilde{O}(T^{2/3})$ regret bounds, have been "folklore knowledge" for decades. The general definition and lower bounds trace back to Babaioff et al. (2014).[23]

Advanced aspects. Switching from "greedy" to "exploration-separating" to "adaptive-exploration" algorithms involves substantial adoption costs in infrastructure and personnel training (Agarwal et al., 2017). Inserting exploration into a complex decision-making pipeline necessitates a substantial awareness of the technology and a certain change in mindset, as well as an infrastructure to collect and analyze the data. Adaptive exploration requires the said infrastructure to propagate the data analysis back to the "front-end" where the decisions are made, and do it on a sufficiently fast and regular cadence. Framing the problem (*e.g.,* choosing modeling assumptions and action features) and debugging the machine learning algorithms tend to be quite subtle, too.

The lower bounds mentioned above are fairly typical: while they are usually (and most cleanly) presented as worst-case, the actually hold for a wide variety of problem instances. The $\Omega(\sqrt{T})$ lower bound from Auer et al. (2002b) can be extended to hold for most problem instances, in the following sense: for each instance $\mathcal{I}$ there exists a "decoy instance" $\mathcal{I}'$ such that any algorithm incurs regret $\Omega(\sqrt{T})$ on at least one of them. The "gap-dependent" lower bound of $\Omega(\frac{K}{\Delta}\log T)$ in fact holds for all problem instances and all algorithms that are not *terrible* on the large-gap instances (Lai and Robbins, 1985). The $\Omega(T^{2/3})$ lower bound for exploration-separating algorithms in fact applies to all problem instances, as long as the algorithm achieves $\tilde{O}(T^{2/3})$ regret rate in the worst case (Babaioff et al., 2014).[24]

[21]While Thompson Sampling dates back to 1933 and is probably the best-known bandit algorithm, its regret has not been understood until recently (Agrawal and Goyal, 2012; Kaufmann et al., 2012; Agrawal and Goyal, 2013).

[22]A substantial follow-up work on more "refined" regret rates is not as relevant to this paper.

[23]Babaioff et al. (2014) consider a closely related, but technically different setting, which can be easily "translated" into ours (either as a corollary or as another application of the same proof technique).

[24]Moreover, there is a tradeoff between the worst-case upper bound on the regret rate and a lower bound that applies for all problem instances (Theorem 4.3 in Babaioff et al., 2014).

Some MAB algorithms, *e.g.,* Thompson Sampling, are Bayesian: they input a prior on mean rewards, and attain strong Bayesian guarantees (in expectation over the prior) when the prior is correct. Such algorithms can also be initialized with some simple 'fake' priors; in fact, this is how Thompson Sampling can be made to satisfy the optimal regret bounds.

The intuition on (the separation between) the three algorithm classes applies more generally, far beyond the basic MAB model discussed above. In particular, all algorithms that we explicitly mentioned are in fact general algorithmic techniques that are known to extend to a variety of more general MAB scenarios, typically with a similarly stark separation in regret bounds.

The greedy algorithm can perform well *sometimes* in a more general model of *contextual bandits*, where auxiliary payoff-relevant signals, a.k.a. contexts, are observed before each round. This phenomenon has been observed in practice (Bietti et al., 2018), and in theory (Kannan et al., 2018; Bastani et al., 2021; Raghavan et al., 2018) under (very) substantial assumptions. The prevalent intuition is that the diversity of contexts can — under some conditions and to a limited extent — substitute for explicit exploration.

Instantaneous regret. Cumulative performance measures such as regret are not quite appropriate for our setting, as we need to characterize interactions in particular rounds. Instead, our theoretical results focus on *Bayesian instantaneous regret* (BIR), as defined in Section 4.1. Recall that we posit a Bayesian prior on the mean reward vectors. In the notation of this appendix, the BIR is simply:

$$\texttt{BIR}(t) := \mathop{\mathbb{E}}_{\text{prior}} \left[\max_{\text{arms } a} \mu_a - r_t \right].$$

Note that Bayesian regret (*i.e.,* regret in expectation over the prior) is precisely

$$\texttt{BReg}(T) := \mathop{\mathbb{E}}_{\text{prior}} \left[T \cdot \max_{\text{arms } a} \mu_a - \sum_{t=1}^{T} r_t \right] = \sum_{t=1}^{T} \texttt{BIR}(t). \tag{17}$$

We are primarily interested in how fast BIR decreases with t, treating K as a constant.

The three classes are well-separated in terms of BIR, much like they are in terms of regret.

- BayesGreedy has at least a constant BIR for many reasonable priors (where the constant can depend on K and the prior, but not on t). The reason / proof is the same as for regret.

- Exploration-separating algorithms can achieve $\texttt{BIR}(t) = \tilde{O}\left(t^{-1/3}\right)$ for all priors, *e.g.,* by using Epsilon-Greedy algorithm with exploration probability $\epsilon_t = t^{-1/3}$ in each round t. In the typical scenario when $\texttt{BReg}(t) \geq \Omega(t^{2/3})$, the BIR rate of $t^{-1/3}$ cannot be improved by (17), in the following sense: if $\texttt{BIR}(t) = \tilde{O}\left(t^{-\gamma}\right)$ for all t, then $\gamma \geq 1/3$.

- Adaptive-exploration algorithms *can* have an even better regret rate: $\texttt{BIR}(t) = \tilde{O}\left(t^{-1/2}\right)$. This holds for Successive Elimination (Even-Dar et al., 2006) and for Thompson Sampling (Sellke and Slivkins, 2021).[25] Any optimal MAB algorithm enjoys this regret rate "on average" by (17), since $\texttt{BReg}(T) \leq \tilde{O}(\sqrt{T})$. In particular, if such algorithm satisfies $\texttt{BIR}(t) = \tilde{O}\left(t^{-\gamma}\right)$ for all rounds t and some constant γ, then $\gamma \leq 1/2$.

This theoretical intuition is supported by our numerical simulations: see Figure 5 and Appendix E.1.

[25] However, such result is not known for UCB1 algorithm, to the best of our knowledge.

B Monotone MAB algorithms

This appendix proves Bayesian-monotonicity of `BayesGreedy` and `BayesEpsilonGreedy`. (The former is needed in Section 4, the latter merely adds motivation for our theoretical story.) Recall that an algorithm is called Bayesian-monotone if its Bayesian-expected reward is non-decreasing in time. Note that `ThompsonSampling` is known to be Bayesian-monotone if the prior $\mathcal{P}_{\texttt{mean}}$ is independent across arms (Sellke and Slivkins, 2021).

We consider Bayesian MAB with Bernoulli rewards. There are T rounds and K arms. In each round $t \in [T]$, the algorithm chooses an arm $a_t \in A$ and receives a reward $r_t \in \{0, 1\}$ for this arm, drawn from a fixed but unknown distribution. The set of all arms is A; mean reward of arm a is denoted μ_a. The mean reward vector $\mu = (\mu_a : \; a \in A)$ is drawn from a common Bayesian prior $\mathcal{P}_{\texttt{mean}}$. We let $\texttt{rew}(t) = \mu_{a_t}$ denote the instantaneous mean reward of the algorithm.

Monotonicity for the greedy algorithm. We state the monotonicity-in-information result for the "Bayesian-greedy step": informally, exploitation can only get better with more data. We invoke this result directly in Section 4, and use it to derive monotonicity of `BayesGreedy` and `BayesEpsilonGreedy`.

A formal statement needs some scaffolding. The n-step history is the random sequence $H_n = (\,(a_t, r_t) : \; t \in [n]\,)$. Realizations of H_n are called *realized histories*. Let $\mathcal{H}_n$ be the set of all possible values of H_n. The *Bayesian-greedy step* given an n-step history $h \in \mathcal{H}_n$ is defined as

$$\texttt{BG}(h) := \operatorname*{argmax}_{a \in A} \mathbb{E}\left[\, \mu_a \mid H_n = h \,\right], \quad \text{ties broken arbitrarily}.$$

(However, recall that such ties are ruled out by Assumption 3.) Now, the result is as follows:

Lemma B.1 (Mansour et al. (2022))**.** *Let h, h' be two realized histories such that h is a prefix of h'. Then*

$$\mathbb{E}\left[\, \mu_{\texttt{BG}(h)} \,\right] \leq \mathbb{E}\left[\, \mu_{\texttt{BG}(h')} \,\right].$$

Corollary B.2. `BayesGreedy` *is Bayesian-monotone. Moreover,* $\mathbb{E}[\texttt{rew}(n)]$ *strictly increases in each time step n with* $\Pr[a_n \neq a_{n+1}] > 0$.

Proof. Bayesian-monotonicity follows directly. The "strictly increases" statement holds because the arm chosen in a given round has a strictly largest Bayesian-expected reward for that round. □

Monotonicity for Epsilon-Greedy. Lemma B.1 immediately implies monotonicity of `BayesEpsilonGreedy`, for a generic choice of exploration probabilities. Recall that in each round t, `BayesEpsilonGreedy` algorithm explores uniformly with a predetermined probability ϵ_t, and "exploits" with the remaining probability using the Bayesian-greedy step: $a_t = \texttt{BG}(\text{current data})$.

Corollary B.3. `BayesEpsilonGreedy` *is Bayesian-monotone whenever probabilities ϵ_t are non-increasing.*

C Non-degeneracy via a random perturbation

We provide two examples when Assumption (3) holds almost surely under a small random perturbation of the prior. We posit Bernoulli rewards, and assume that the prior $\mathcal{P}_{\texttt{mean}}$ is independent across arms.

Beta priors. Suppose the mean reward μ_a for each arm a is drawn from some Beta distribution $\text{Beta}(\alpha_a, \beta_a)$. Given any history H that contains h_a number of heads and t_a number of tails from arm a, the posterior mean reward is $\mathbb{E}[\mu_a \mid H] = \frac{\alpha_a+h_a}{\alpha_a+h_a+\beta_a+t_a}$. Therefore, perturbing the parameters α_a and β_a independently with any continuous noise will induce a prior with property (3) with probability 1.

A prior with a finite support. Consider the probability vector in the prior for arm a:

$$\vec{p}_a = (\ \Pr[\mu_a = \nu] :\ \nu \in \texttt{support}(\mu_a)\ ).$$

We apply a small random perturbation independently to each such vector:

$$\vec{p}_a \leftarrow \vec{p}_a + \vec{q}_a, \quad \text{where} \quad \vec{q}_a \sim \mathcal{N}_a. \tag{18}$$

Here $\mathcal{N}_a$ is the noise distribution for arm a: a distribution over real-valued, zero-sum vectors of dimension $d_a = |\texttt{support}(\mu_a)|$. We need the noise distribution to satisfy the following property:

$$\forall x \in [-1,1]^{d_a} \setminus \{0\} \qquad \Pr_{q\sim\mathcal{N}_a}[x \cdot (\vec{p}_a + q) \neq 0] = 1. \tag{19}$$

Theorem C.1. *Consider an instance of MAB with 0-1 rewards. Assume that the prior $\mathcal{P}_{\texttt{mean}}$ is independent across arms, and each mean reward μ_a has a finite support that does not include 0 or 1. Assume that noise distributions $\mathcal{N}_a$ satisfy property (19). If random perturbation (18) is applied independently to each arm a, then Eq. (3) holds almost surely for each history h.*

Remark C.2. As a generic example of a noise distribution which satisfies Property (19), consider the uniform distribution $\mathcal{N}$ over the bounded convex set $Q = \left\{ q \in \mathbb{R}^{d_a} \mid q \cdot \vec{1} = 0 \text{ and } \|q\|_2 \leq \epsilon \right\}$, where $\vec{1}$ denotes the all-1 vector. If $x = a\vec{1}$ for some non-zero value of a, then (19) holds because $x \cdot (p+q) = x \cdot p = a \neq 0$. Otherwise, denote $p = \vec{p}_a$ and observe that $x \cdot (p+q) = 0$ only if $x \cdot q = c \triangleq x \cdot (-p)$. Since $x \neq \vec{1}$, the intersection $Q \cap \{x \cdot q = c\}$ either is empty or has measure 0 in Q, which implies $\Pr_q[x \cdot (p+q) \neq 0] = 1$.

To prove Theorem C.1, it suffices to focus on two arms, and perturb one. Since realized rewards have finite support, there are only finitely many possible histories. So, it suffices to focus on a fixed history h.

Lemma C.3. *Consider an instance of MAB with Bernoulli rewards. Assume that the prior $\mathcal{P}_{\texttt{mean}}$ is independent across arms, and that $\texttt{support}(\mu_1)$ is finite and does not include 0 or 1. Suppose random perturbation (18) is applied to arm 1, with noise distribution $\mathcal{N}_1$ that satisfies (19). Then $\mathbb{E}[\mu_1 \mid h] \neq \mathbb{E}[\mu_2 \mid h]$ almost surely for any fixed history h.*

Proof. Note that $\mathbb{E}[\mu_a \mid h]$ does not depend on the algorithm which produced this history. Therefore, for the sake of the analysis, we can assume w.l.o.g. that this history has been generated by a particular algorithm, as long as this algorithm can can produce this history with non-zero probability. Let us consider the algorithm that deterministically chooses same actions as h. Let $S = \texttt{support}(\mu_1)$. Then:

$$\begin{aligned}\mathbb{E}[\mu_1 \mid h] &= \textstyle\sum_{\nu\in S}\ \nu \cdot \Pr[\mu_1 = \nu \mid h] \\ &= \textstyle\sum_{\nu\in S}\ \nu \cdot \Pr[h \mid \mu_1 = \nu] \cdot \Pr[\mu_1 = \nu]\ /\ \Pr[h], \\ \Pr[h] &= \textstyle\sum_{\nu\in S}\ \Pr[h \mid \mu_1 = \nu] \cdot \Pr[\mu_1 = \nu].\end{aligned}$$

Therefore, $\mathbb{E}[\mu_1 \mid h] = \mathbb{E}[\mu_2 \mid h]$ if and only if

$$\textstyle\sum_{\nu\in S}(\nu - C) \cdot \Pr[h \mid \mu_1 = \nu] \cdot \Pr[\mu_1 = \nu] = 0, \quad \text{where} \quad C = \mathbb{E}[\mu_2 \mid h].$$

Since $\mathbb{E}[\mu_2 \mid h]$ and $\Pr[h \mid \mu_1 = \nu]$ do not depend on the probability vector $\vec{p}_1$, we conclude that

$$\mathbb{E}[\mu_1 \mid h] = \mathbb{E}[\mu_2 \mid h] \quad \Leftrightarrow \quad x \cdot \vec{p}_1 = 0,$$

where vector

$$x := (\ (\nu - C) \cdot \Pr[h \mid \mu_1 = \nu] :\ \nu \in S\ ) \in [-1,1]^{d_1}$$

does not depend on $\vec{p}_1$.

Thus, it suffices to prove that $x \cdot \vec{p}_1 \neq 0$ almost surely under the perturbation. In a formula:

$$\Pr_{q\sim\mathcal{N}_1}[x \cdot (\vec{p}_1 + q) \neq 0] = 1 \tag{20}$$

Note that $\Pr[h \mid \mu_1 = \nu] > 0$ for all $\nu \in S$, because $0, 1 \notin S$. It follows that at most one coordinate of x can be zero. So (20) follows from property (19). □

D Full proofs for Section 4

Some notation. Without loss of generality, we label actions as $A = [K]$ and sort them according to their prior mean rewards, so that $\mathbb{E}[\mu_1] > \mathbb{E}[\mu_2] > \ldots > \mathbb{E}[\mu_K]$.

Fix principal $i \in \{1, 2\}$ and (local) step n. The arm chosen by algorithm $\texttt{alg}_i$ at this step is denoted $a_{i,n}$, and the corresponding BIR is denoted $\texttt{BIR}_i(n)$. History of $\texttt{alg}_i$ up to this step is denoted $H_{i,n}$.

Fix agent t. Recall that $n_i(t)$ denotes the number of global rounds before t in which principal i is chosen. Let $\mathcal{N}_{i,t}$ denote the distribution of $n_i(t)$.

Write $\texttt{PMR}(a \mid E) = \mathbb{E}[\mu_a \mid E]$ for posterior mean reward of action a given event E.

Chernoff Bounds. We use an elementary concentration inequality known as *Chernoff Bounds*, in a formulation from Mitzenmacher and Upfal (2005).

Theorem D.1 (Chernoff Bounds)**.** *Consider n i.i.d. random variables $X_1 \ldots X_n$ with values in $[0, 1]$. Let $X = \frac{1}{n}\sum_{i=1}^{n} X_i$ be their average, and let $\nu = \mathbb{E}[X]$. Then:*

$$\min\left(\ \Pr[X - \nu > \delta\nu], \quad \Pr[\nu - X > \delta\nu]\ \right) < e^{-\nu n \delta^2/3} \quad \textit{for any } \delta \in (0, 1).$$

D.1 Main result on HardMax: Proof of Theorem 4.2

Proof of Lemma 4.4. Since the two algorithms coincide on the first $n_0 - 1$ steps, it follows by symmetry that histories H_{1,n_0} and H_{2,n_0} have the same distribution. We use a *coupling argument*: w.l.o.g., we assume the two histories coincide, $H_{1,n_0} = H_{2,n_0} = H$.

At local step n_0, BayesGreedy chooses an action $a_{1,n_0} = a_{1,n_0}(H)$ which maximizes the posterior mean reward given history H: for any realized history $h \in \texttt{support}(H)$ and any action $a \in A$

$$\texttt{PMR}(a_{1,n_0} \mid H = h) \geq \texttt{PMR}(a \mid H = h). \tag{21}$$

By assumption (3), it follows that

$$\texttt{PMR}(a_{1,n_0} \mid H = h) > \texttt{PMR}(a \mid H = h) \quad \text{for any } h \in \texttt{support}(H) \text{ and } a \neq a_{1,n_0}(h). \tag{22}$$

Since the two algorithms deviate at step n_0, there is a set $S \subset \texttt{support}(H)$ of step-n_0 histories such that $\Pr[S] > 0$ and any history $h \in S$ satisfies $\Pr[a_{2,n_0} \neq a_{1,n_0} \mid H = h] > 0$. Combining this with (22),

$$\texttt{PMR}(a_{1,n_0} \mid H = h) > \mathbb{E}\left[\mu_{a_{2,n_0}} \mid H = h\right] \quad \text{for each history } h \in S. \tag{23}$$

Using (21) and (23) and integrating over realized histories h, we obtain $\mathbb{E}[\texttt{rew}_1(n_0)] > \mathbb{E}[\texttt{rew}_2(n_0)]$. □

Proof of Lemma 4.5. Let us use induction on round $t \geq t_0$, with the base case $t = t_0$. Let $\mathcal{N} = \mathcal{N}_{1,t_0}$ be the agents' posterior distribution for n_{1,t_0}, the number of global rounds before t_0 in which principal 1 is chosen. By induction, all agents from t_0 to $t-1$ chose principal 1, so $\texttt{PMR}_2(t_0) = \texttt{PMR}_2(t)$. Therefore,

$$\texttt{PMR}_1(t) = \mathop{\mathbb{E}}_{n\sim\mathcal{N}}[\texttt{rew}_1(n+1+t-t_0)] \geq \mathop{\mathbb{E}}_{n\sim\mathcal{N}}[\texttt{rew}_1(n+1)] = \texttt{PMR}_1(t_0) > \texttt{PMR}_2(t_0) = \texttt{PMR}_2(t),$$

where the first inequality holds because $\texttt{alg}_1$ is Bayesian-monotone, and the second one is the base case. □

Proof of Theorem 4.2. Since the two algorithms coincide on the first $n_0 - 1$ steps, it follows by symmetry that $\mathbb{E}[\texttt{rew}_1(n)] = \mathbb{E}[\texttt{rew}_2(n)]$ for any $n < n_0$. By Lemma 4.4, it holds that $\mathbb{E}[\texttt{rew}_1(n_0)] > \mathbb{E}[\texttt{rew}_2(n_0)]$.

Recall that $n_i(t)$ is the number of global rounds $s < t$ in which principal i is chosen, and $\mathcal{N}_{i,t}$ is the agents' posterior distribution for this quantity. By symmetry, each agent $t < n_0$ chooses a principal uniformly at random. It follows that $\mathcal{N}_{1,n_0} = \mathcal{N}_{2,n_0}$ (denote both distributions by $\mathcal{N}$ for brevity), and $\mathcal{N}(n_0 - 1) > 0$. Therefore:

$$\begin{aligned}
\texttt{PMR}_1(n_0) &= \mathop{\mathbb{E}}_{n\sim\mathcal{N}}[\texttt{rew}_1(n+1)] = \sum_{n=0}^{n_0-1} \mathcal{N}(n) \cdot \mathbb{E}[\texttt{rew}_1(n+1)] \\
&> \mathcal{N}(n_0-1) \cdot \mathbb{E}[\texttt{rew}_2(n_0)] + \sum_{n=0}^{n_0-2} \mathcal{N}(n) \cdot \mathbb{E}[\texttt{rew}_2(n+1)] \\
&= \mathop{\mathbb{E}}_{n\sim\mathcal{N}}[\texttt{rew}_2(n+1)] = \texttt{PMR}_2(n_0)
\end{aligned} \tag{24}$$

So, agent n_0 chooses principal 1. By Lemma 4.5 (noting that BayesGreedy is Bayesian-monotone), all subsequent agents choose principal 1, too. □

D.2 HardMax with biased tie-breaking: Proof of Theorem 4.7

The proof re-uses Lemmas 4.4 and 4.5, which do not rely on fair tie-breaking.

Recall that i_t is the principal chosen in a given global round t. Because of the biased tie-breaking,

$$\text{if } \texttt{PMR}_1(t) \geq \texttt{PMR}_2(t) \text{ then } \Pr[i_t = 1] > \tfrac{1}{2}. \tag{25}$$

Let m_0 be the first step when $\texttt{alg}_2$ deviates from BayesGreedy, or BayesGreedy deviates from StaticGreedy, whichever comes sooner. Then $\texttt{alg}_2$, BayesGreedy and StaticGreedy coincide on the first $m_0 - 1$ steps. Moreover, $m_0 \leq n_0$ (since BayesGreedy deviates from StaticGreedy at step n_0), so $\texttt{alg}_1$ coincides with BayesGreedy on the first m_0 steps.

So, $\mathbb{E}[\texttt{rew}_1(n)] = \mathbb{E}[\texttt{rew}_2(n)]$ for each step $n < m_0$, because $\texttt{alg}_1$ and $\texttt{alg}_2$ coincide on the first $m_0 - 1$ steps. Moreover, if $\texttt{alg}_2$ deviates from BayesGreedy at step m_0 then $\mathbb{E}[\texttt{rew}_1(m_0)] > \mathbb{E}[\texttt{rew}_2(m_0)]$ by Lemma 4.4; else, we trivially have $\texttt{rew}_1(m_0) = \texttt{rew}_2(m_0)$. To summarize:

$$\mathbb{E}[\texttt{rew}_1(n)] \geq \mathbb{E}[\texttt{rew}_2(n)] \quad \text{for all steps } n \leq m_0. \tag{26}$$

We claim that $\Pr[i_t = 1] > \frac{1}{2}$ for all global rounds $t \leq m_0$. We prove this claim using induction on t. The base case $t = 1$ holds by (25) and the fact that in step 1, BayesGreedy chooses the arm with the highest prior mean reward. For the induction step, we assume that $\Pr[i_t = 1] > \frac{1}{2}$ for all global rounds $t < t_0$, for some $t_0 \leq m_0$. It follows that distribution $\mathcal{N}_{1,t_0}$ stochastically dominates distribution $\mathcal{N}_{2,t_0}$.[26] Observe that

$$\texttt{PMR}_1(t_0) = \mathop{\mathbb{E}}_{n \sim \mathcal{N}_{1,t_0}} [\texttt{rew}_1(n+1)] \geq \mathop{\mathbb{E}}_{n \sim \mathcal{N}_{2,t_0}} [\texttt{rew}_2(n+1)] = \texttt{PMR}_2(t_0). \tag{27}$$

So the induction step follows by (25). Claim proved.

Now let us focus on global round m_0, and denote $\mathcal{N}_i = \mathcal{N}_{i,m_0}$. By the above claim,

$$\mathcal{N}_1 \text{ stochastically dominates } \mathcal{N}_2, \text{ and moreover } \mathcal{N}_i(m_0 - 1) > \mathcal{N}_i(m_0 - 1). \tag{28}$$

By definition of m_0, either (i) $\texttt{alg}_2$ deviates from BayesGreedy starting from local step m_0, which implies $\texttt{rew}_1(m_0) > \texttt{rew}_2(m_0)$ by Lemma 4.4, or (ii) BayesGreedy deviates from StaticGreedy starting from local step m_0, which implies $\mathbb{E}[\texttt{rew}_1(m_0)] > \mathbb{E}[\texttt{rew}_1(m_0 - 1)]$ by Lemma B.2. In both cases, using (26) and (28), it follows that the inequality in (27) is strict for $t_0 = m_0$.

Therefore, agent m_0 chooses principal 1, and by Lemma 4.5 so do all subsequent agents.

D.3 The main result for HardMax&Random: Proof of Theorem 4.8

Without loss of generality, assume $m_0 = n_0$. Consider global round $t \geq n_0$. Recall that each agent chooses principal 1 with probability at least $f_{\texttt{resp}}(-1) > 0$.

Then $\mathbb{E}[n_1(t+1)] \geq 2\epsilon_0\, t$. By Chernoff Bounds (Theorem D.1), we have that $n_1(t+1) \geq \epsilon_0 t$ holds with probability at least $1 - q$, where $q = \exp(-\epsilon_0 t/12)$.

[26] For random variables X, Y on $\mathbb{R}$, we say that X *stochastically dominates* Y if $\Pr[X \geq x] \geq \Pr[Y \geq x]$ for any $x \in \mathbb{R}$.

We need to prove that $\texttt{PMR}_1(t) - \texttt{PMR}_2(t) > 0$. For any m_1 and m_2, consider the quantity

$$\Delta(m_1, m_2) := \texttt{BIR}_2(m_2 + 1) - \texttt{BIR}_1(m_1 + 1).$$

Whenever $m_1 \geq \epsilon_0 t/2 - 1$ and $m_2 < t$, it holds that

$$\Delta(m_1, m_2) \geq \Delta(\epsilon_0 t/2, t) \geq \texttt{BIR}_2(t)/2.$$

The above inequalities follow, resp., from algorithms' Bayesian-monotonicity and (6). Now,

$$\begin{aligned}
\texttt{PMR}_1(t) - \texttt{PMR}_2(t) &= \mathop{\mathbb{E}}_{m_1 \sim \mathcal{N}_{1,t},\, m_2 \sim \mathcal{N}_{2,t}} [\Delta(m_1, m_2)] \\
&\geq -q + \mathop{\mathbb{E}}_{m_1 \sim \mathcal{N}_{1,t},\, m_2 \sim \mathcal{N}_{2,t}} [\Delta(m_1, m_2) \mid m_1 \geq \epsilon_0 t/2 - 1] \\
&\geq \texttt{BIR}_2(t)/2 - q \\
&> \texttt{BIR}_2(t)/4 > 0 \qquad \textit{(by Eq. (7))}.
\end{aligned}$$

D.4 A little greedy goes a long way (Proof of Theorem 4.10)

Let $\texttt{rew}_{\texttt{gr}}(n)$ denote the Bayesian-expected reward of the "greedy choice" after after $n-1$ steps of $\texttt{alg}_1$. Note that $\texttt{rew}_1(\cdot)$ and $\texttt{rew}_{\texttt{gr}}(\cdot)$ are non-decreasing: the former because $\texttt{alg}_1$ is Bayesian-monotone and the latter because the "greedy choice" is only improved with an increasing set of observations, see Lemma B.1. Using (9), we conclude that the greedy modification $\texttt{alg}_2$ is Bayesian-monotone.

By definition of the "greedy choice," $\texttt{rew}_1(n) \leq \texttt{rew}_{\texttt{gr}}(n)$ for all steps n. Moreover, by Lemma 4.4, $\texttt{alg}_1$ has a strictly smaller $\texttt{rew}(n_0)$ compared to $\texttt{BayesGreedy}$; so, $\texttt{rew}_1(n_0) < \texttt{rew}_2(n_0)$.

Let $\texttt{alg}$ denote a copy of $\texttt{alg}_1$ that is running "inside" $\texttt{alg}_2$. Let $m_2(t)$ be the number of global rounds before t in which the agent chooses principal 2 *and* $\texttt{alg}$ is invoked; *i.e.,* it is the number of agents seen by $\texttt{alg}$ before global round t. Let $\mathcal{M}_{2,t}$ be the agents' posterior distribution for $m_2(t)$.

We claim that in each global round $t \geq n_0$, distribution $\mathcal{M}_{2,t}$ stochastically dominates distribution $\mathcal{N}_{1,t}$, and $\texttt{PMR}_1(t) < \texttt{PMR}_2(t)$. We use induction on t. The base case $t = n_0$ holds because $\mathcal{M}_{2,t} = \mathcal{N}_{1,t}$ (because the two algorithms coincide on the first $n_0 - 1$ steps), and $\texttt{PMR}_1(n_0) < \texttt{PMR}_2(n_0)$ is proved as in (24), using the fact that $\texttt{rew}_1(n_0) < \texttt{rew}_2(n_0)$.

The induction step is proved as follows. The induction hypothesis for global round $t-1$ implies that agent $t-1$ is seen by $\texttt{alg}$ with probability $(1-\epsilon_0)(1-p)$, which is strictly larger than ϵ_0, the probability with which this agent is seen by $\texttt{alg}_2$. Therefore, $\mathcal{M}_{2,t}$ stochastically dominates $\mathcal{N}_{1,t}$.

$$\begin{aligned}
\texttt{PMR}_1(t) &= \mathop{\mathbb{E}}_{n \sim \mathcal{N}_{1,t}} [\texttt{rew}_1(n+1)] \\
&\leq \mathop{\mathbb{E}}_{m \sim \mathcal{M}_{2,t}} [\texttt{rew}_1(m+1)] && (29) \\
&< \mathop{\mathbb{E}}_{m \sim \mathcal{M}_{2,t}} [(1-p) \cdot \texttt{rew}_1(m+1) + p \cdot \texttt{rew}_{\texttt{gr}}(m+1)] && (30) \\
&= \texttt{PMR}_2(t).
\end{aligned}$$

Here (29) holds because $\mathbb{E}[\texttt{rew}_1(\cdot)]$ is Bayesian-monotone and $\mathcal{M}_{2,t}$ stochastically dominates $\mathcal{N}_{1,t}$, and inequality (30) holds because $\mathbb{E}[\texttt{rew}_1(n_0)] < \mathbb{E}[\texttt{rew}_2(n_0)]$ and $\mathcal{M}_{2,t}(n_0) > 0$.[27]

[27] If $\mathbb{E}[\texttt{rew}_1(\cdot)]$ is strictly increasing, then (29) is strict, too; this is because $\mathcal{M}_{2,t}(t-1) > \mathcal{N}_{1,t}(t-1)$.

D.5 SoftMax: proof of Theorem 4.19

Let $\beta_1 = \min\{c_0'\delta_0, \beta_0/20\}$ with δ_0 defined in (10). Recall each agent chooses $\texttt{alg}_1$ with probability at least $f_{\texttt{resp}}(-1) = \epsilon_0$. By By condition (13) and the fact that $\texttt{BIR}_1(n) \to 0$, there exists some sufficiently large T_1 such that for any $t \geq T_1$, $\texttt{BIR}_1(\epsilon_0 T_1/2) \leq \beta_1/c_0'$ and $\texttt{BIR}_2(t) > e^{-\epsilon_0 t/12}$. Moreover, for any $t \geq T_1$, we know $\mathbb{E}[n_1(t+1)] \geq \epsilon_0\, t$, and by the Chernoff Bounds (Theorem D.1), we have $n_1(t+1) \geq \epsilon_0 t/2$ holds with probability at least $1 - q_1(t)$ with $q_1(t) = \exp(-\epsilon_0 t/12) < \texttt{BIR}_2(t)$. It follows that for any $t \geq T_1$,

$$\begin{aligned}
\texttt{PMR}_2(t) - \texttt{PMR}_1(t) &= \mathop{\mathbb{E}}_{m_1 \sim \mathcal{N}_{1,t},\, m_2 \sim \mathcal{N}_{2,t}} \left[\texttt{BIR}_1(m_1+1) - \texttt{BIR}_2(m_2+1)\right] \\
&\leq q_1(t) + \mathop{\mathbb{E}}_{m_1 \sim \mathcal{N}_{1,t}} \left[\texttt{BIR}_1(m_1+1) \mid m_1 \geq \epsilon_0 t/2 - 1\right] - \texttt{BIR}_2(t) \\
&\leq \texttt{BIR}_1(\epsilon_0 T_1/2) \leq \beta_1/c_0'
\end{aligned}$$

Since the response function $f_{\texttt{resp}}$ is c_0'-Lipschitz in the neighborhood of $[-\delta_0, \delta_0]$, each agent after round T_1 will choose $\texttt{alg}_1$ with probability at least

$$p_t \geq \tfrac{1}{2} - c_0'\left(\texttt{PMR}_2(t) - \texttt{PMR}_1(t)\right) \geq \tfrac{1}{2} - \beta_1.$$

Next, we will show that there exists a sufficiently large T_2 such that for any $t \geq T_1 + T_2$, with high probability $n_1(t) > \max\{n_0, (1-\beta_0)n_2(t)\}$, where n_0 is defined in (12). Fix any $t \geq T_1 + T_2$. Since each agent chooses $\texttt{alg}_1$ with probability at least $1/2 - \beta_1$, by Chernoff Bounds (Theorem D.1) we have with probability at least $1 - q_2(t)$ that the number of agents that choose $\texttt{alg}_1$ is at least $\beta_0({}^1\!/\!{}_2 - \beta_1)t/5$, where

$$q_2(x) = \exp\left(-{}^1\!/\!{}_3\ ({}^1\!/\!{}_2 - \beta_1)(1-\beta_0/5)^2 x\right).$$

The number of agents received by $\texttt{alg}_2$ is at most $T_1 + ({}^1\!/\!{}_2 + \beta_1)t + ({}^1\!/\!{}_2 - \beta_1)(1-\beta_0/5)t$.

Then as long as $T_2 \geq \frac{5T_1}{\beta_0}$, we can guarantee that $n_1(t) > n_2(t)(1-\beta_0)$ and $n_1(t) > n_0$ with probability at least $1 - q_2(t)$ for any $t \geq T_1 + T_2$. Note that the weak BIR-dominance condition in (12) implies that for any $t \geq T_1 + T_2$ with probability at least $1 - q_2(t)$, we have $\texttt{BIR}_1(n_1(t)) < (1-\alpha_0)\,\texttt{BIR}_2(n_2(t))$.

It follows that for any $t \geq T_1 + T_2$,

$$\begin{aligned}
\texttt{PMR}_1(t) - \texttt{PMR}_2(t) &= \mathop{\mathbb{E}}_{m_1 \sim \mathcal{N}_{1,t},\, m_2 \sim \mathcal{N}_{2,t}} \left[\texttt{BIR}_2(m_2+1) - \texttt{BIR}_1(m_1+1)\right] \\
&\geq (1 - q_2(t))\,\alpha_0\,\texttt{BIR}_2(t) - q_2(t) \geq \alpha_0\,\texttt{BIR}_2(t)/4,
\end{aligned}$$

where the last inequality holds as long as $q_2(t) \leq \alpha_0 \texttt{BIR}_2(t)/4$, and is implied by the condition in (13) as long as T_2 is sufficiently large. Hence, by the definition of our SoftMax response function and assumption in (10), we have $\Pr[i_t = 1] \geq {}^1\!/\!{}_2 + {}^1\!/\!{}_4\ c_0\,\alpha_0\,\texttt{BIR}_2(t)$.

E Full experimental results

In this appendix we provide full results for the experiments described in Section 5.

E.1 "Performance In Isolation" (Section 5.2)

We present the full plots for Section 5.2: mean reputation trajectories and instantaneous reward trajectories for all three MAB instances. For "instantaneous reward" at a given time t, we report the average (over all mean reward vectors) of the mean rewards at this time, instead of the average of the *realized* rewards, so as to decrease the noise. In all plots, the shaded area represents 95% confidence interval.

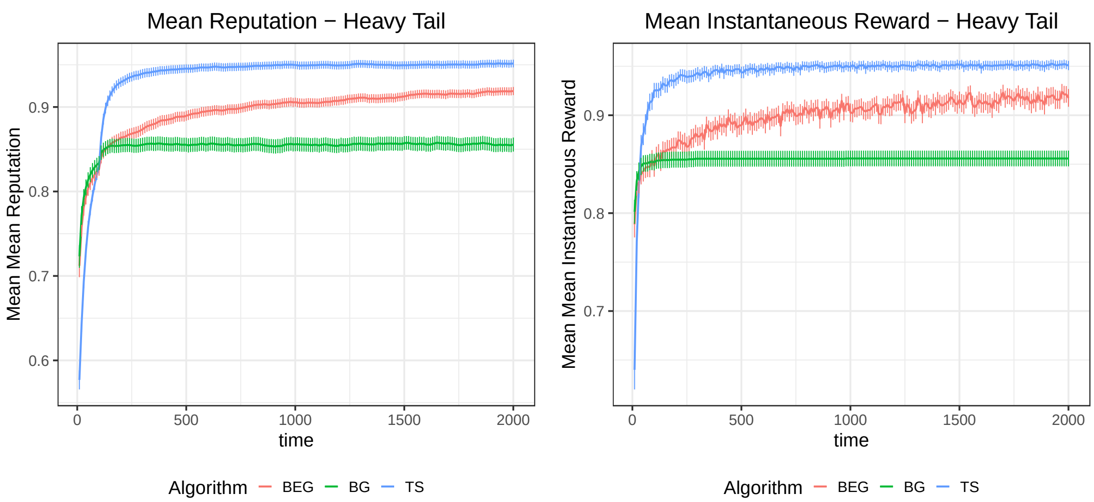

Figure 10: Mean Reputation (left) and Mean Instantaneous Reward (right) for Heavy Tail Instance

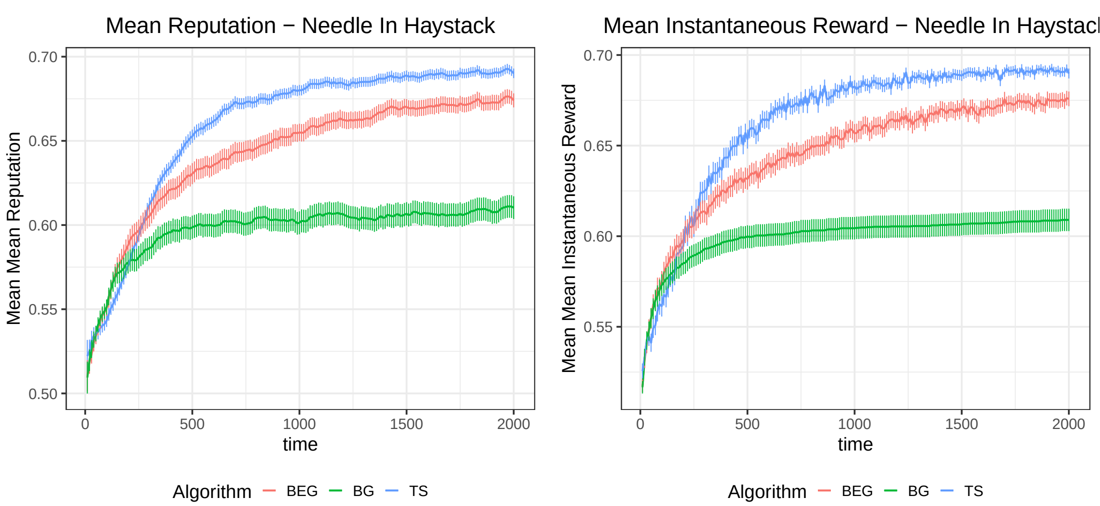

Figure 11: Mean Reputation (left) and Mean Instantaneous Reward (right) for Needle In Haystack Instance

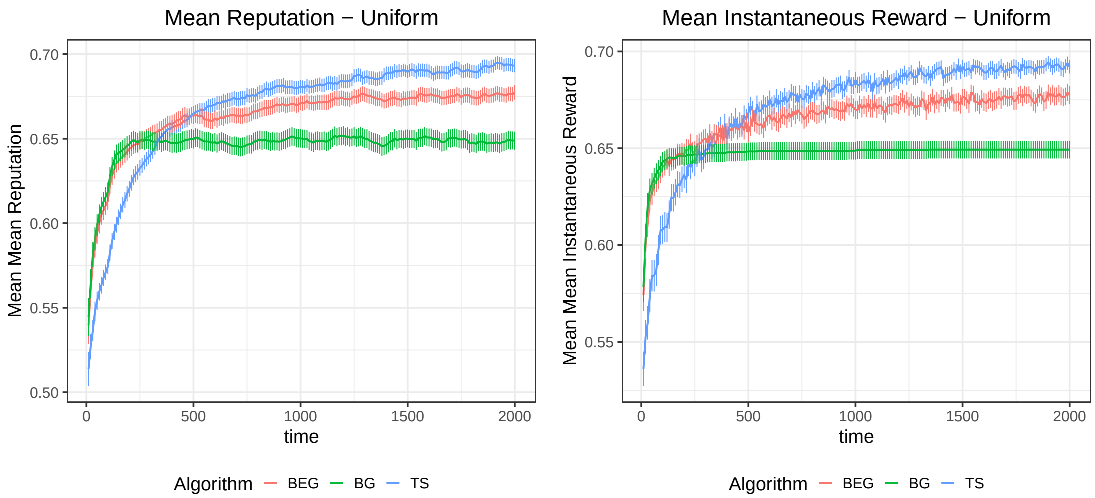

Figure 12: Mean Reputation (left) and Mean Instantaneous Reward (right) for Uniform Instance

E.2 First-mover regime (Section 5.3)

We present additional experiments on the first-mover regime from Section 5.3, across various MAB instances and various values of the incumbent advantage parameter X.

Each experiment is presented as a table with the same semantics as in the main text. Namely, each cell in the table describes the duopoly game between the entrant's algorithm (the row) and the incumbent's algorithm (the column). The cell specifies the entrant's market share (fraction of rounds in which it was chosen) for the rounds in which he was present. We give the average (in bold) and the 95% confidence interval. NB: smaller average is better for the incumbent.

	$X = 50$			$X = 200$		
	TS	BEG	BG	TS	BEG	BG
TS	**0.054** ±0.01	**0.16** ±0.02	**0.18** ±0.02	**0.003** ±0.003	**0.083** ±0.02	**0.17** ±0.02
BEG	**0.33** ±0.03	**0.31** ±0.02	**0.26** ±0.02	**0.045** ±0.01	**0.25** ±0.02	**0.23** ±0.02
BG	**0.39** ±0.03	**0.41** ±0.03	**0.33** ±0.02	**0.12** ±0.02	**0.36** ±0.03	**0.3** ±0.02

Table 7: Heavy-Tail MAB Instance

	$X = 300$			$X = 500$		
	TS	BEG	BG	TS	BEG	BG
TS	**0.0017** ±0.002	**0.059** ±0.01	**0.16** ±0.02	**0.002** ±0.003	**0.043** ±0.01	**0.16** ±0.02
BEG	**0.029** ±0.007	**0.23** ±0.02	**0.23** ±0.02	**0.03** ±0.007	**0.21** ±0.02	**0.24** ±0.02
BG	**0.097** ±0.02	**0.34** ±0.03	**0.29** ±0.02	**0.091** ±0.01	**0.32** ±0.03	**0.3** ±0.02

Table 8: Heavy-Tail MAB Instance

	$X = 50$			$X = 200$		
	TS	BEG	BG	TS	BEG	BG
TS	**0.34** ±0.03	**0.4** ±0.03	**0.48** ±0.03	**0.17** ±0.02	**0.31** ±0.03	**0.41** ±0.03
BEG	**0.22** ±0.02	**0.34** ±0.03	**0.42** ±0.03	**0.13** ±0.02	**0.26** ±0.02	**0.36** ±0.03
BG	**0.18** ±0.02	**0.28** ±0.02	**0.37** ±0.03	**0.093** ±0.02	**0.23** ±0.02	**0.33** ±0.03

Table 9: Needle In Haystack MAB Instance

	$X = 300$			$X = 500$		
	TS	BEG	BG	TS	BEG	BG
TS	**0.1** ±0.02	**0.28** ±0.03	**0.39** ±0.03	**0.053** ±0.01	**0.23** ±0.02	**0.37** ±0.03
BEG	**0.089** ±0.02	**0.23** ±0.02	**0.36** ±0.03	**0.051** ±0.01	**0.2** ±0.02	**0.33** ±0.03
BG	**0.05** ±0.01	**0.21** ±0.02	**0.33** ±0.03	**0.031** ±0.009	**0.18** ±0.02	**0.31** ±0.02

Table 10: Needle In Haystack MAB Instance

	$X = 50$			$X = 200$		
	TS	BEG	BG	TS	BEG	BG
TS	**0.27** ±0.03	**0.21** ±0.02	**0.26** ±0.02	**0.12** ±0.02	**0.16** ±0.02	**0.2** ±0.02
BEG	**0.39** ±0.03	**0.3** ±0.03	**0.34** ±0.03	**0.25** ±0.02	**0.24** ±0.02	**0.29** ±0.02
BG	**0.39** ±0.03	**0.31** ±0.02	**0.33** ±0.02	**0.23** ±0.02	**0.24** ±0.02	**0.29** ±0.02

Table 11: Uniform MAB Instance

	$X = 300$			$X = 500$		
	TS	BEG	BG	TS	BEG	BG
TS	**0.094** ±0.02	**0.15** ±0.02	**0.2** ±0.02	**0.061** ±0.01	**0.12** ±0.02	**0.2** ±0.02
BEG	**0.2** ±0.02	**0.23** ±0.02	**0.29** ±0.02	**0.17** ±0.02	**0.21** ±0.02	**0.29** ±0.02
BG	**0.21** ±0.02	**0.23** ±0.02	**0.29** ±0.02	**0.18** ±0.02	**0.22** ±0.02	**0.29** ±0.02

Table 12: Uniform MAB Instance

E.3 Reputation Advantage vs. Data Advantage (Section 5.4)

This section presents full experimental results on reputation advantage vs. data advantage.

Each experiment is presented as a table with the same semantics as in the main text. Namely, each cell in the table describes the duopoly game between the entrant's algorithm (the **row**) and the incumbent's algorithm (the **column**). The cell specifies the entrant's market share for the rounds in which hit was present: the average (in bold) and the 95% confidence interval. NB: smaller average is better for the incumbent.

	Data Advantage			Reputation Advantage		
	TS	BEG	BG	TS	BEG	BG
TS	**0.0096** ± 0.006	**0.11** ± 0.02	**0.18** ± 0.02	**0.021** ± 0.009	**0.16** ± 0.02	**0.21** ± 0.02
BEG	**0.073** ± 0.01	**0.29** ± 0.02	**0.25** ± 0.02	**0.26** ± 0.03	**0.3** ± 0.02	**0.26** ± 0.02
BG	**0.15** ± 0.02	**0.39** ± 0.03	**0.33** ± 0.02	**0.34** ± 0.03	**0.4** ± 0.03	**0.33** ± 0.02

Table 13: Heavy Tail MAB Instance, $X = 200$

	Data Advantage			Reputation Advantage		
	TS	BEG	BG	TS	BEG	BG
TS	**0.0017** ±0.002	**0.06** ±0.01	**0.18** ±0.02	**0.022** ±0.009	**0.13** ±0.02	**0.21** ±0.02
BEG	**0.04** ±0.009	**0.24** ±0.02	**0.25** ±0.02	**0.26** ±0.03	**0.29** ±0.02	**0.28** ±0.02
BG	**0.12** ±0.02	**0.35** ±0.03	**0.33** ±0.02	**0.33** ±0.03	**0.39** ±0.03	**0.34** ±0.02

Table 14: Heavy Tail MAB Instance, $X = 500$

	Data Advantage			Reputation Advantage		
	TS	BEG	BG	TS	BEG	BG
TS	**0.25** $\pm$ 0.03	**0.36** $\pm$ 0.03	**0.45** $\pm$ 0.03	**0.35** $\pm$ 0.03	**0.43** $\pm$ 0.03	**0.52** $\pm$ 0.03
BEG	**0.21** $\pm$ 0.02	**0.32** $\pm$ 0.03	**0.41** $\pm$ 0.03	**0.26** $\pm$ 0.03	**0.36** $\pm$ 0.03	**0.43** $\pm$ 0.03
BG	**0.18** $\pm$ 0.02	**0.29** $\pm$ 0.03	**0.4** $\pm$ 0.03	**0.19** $\pm$ 0.02	**0.3** $\pm$ 0.02	**0.36** $\pm$ 0.02

Table 15: Needle-in-Haystack MAB Instance, $X = 200$

	Data Advantage			Reputation Advantage		
	TS	BEG	BG	TS	BEG	BG
TS	**0.098** $\pm$0.02	**0.27** $\pm$0.03	**0.41** $\pm$0.03	**0.29** $\pm$0.03	**0.44** $\pm$0.03	**0.52** $\pm$0.03
BEG	**0.093** $\pm$0.02	**0.24** $\pm$0.02	**0.38** $\pm$0.03	**0.19** $\pm$0.02	**0.35** $\pm$0.03	**0.42** $\pm$0.03
BG	**0.064** $\pm$0.01	**0.22** $\pm$0.02	**0.37** $\pm$0.03	**0.15** $\pm$0.02	**0.27** $\pm$0.02	**0.35** $\pm$0.02

Table 16: Needle-in-Haystack MAB Instance, $X = 500$

	Data Advantage			Reputation Advantage		
	TS	BEG	BG	TS	BEG	BG
TS	**0.2** $\pm$ 0.02	**0.22** $\pm$ 0.02	**0.27** $\pm$ 0.03	**0.27** $\pm$ 0.03	**0.23** $\pm$ 0.02	**0.27** $\pm$ 0.02
BEG	**0.33** $\pm$ 0.03	**0.32** $\pm$ 0.03	**0.35** $\pm$ 0.03	**0.4** $\pm$ 0.03	**0.3** $\pm$ 0.02	**0.32** $\pm$ 0.02
BG	**0.32** $\pm$ 0.03	**0.31** $\pm$ 0.03	**0.35** $\pm$ 0.03	**0.36** $\pm$ 0.03	**0.29** $\pm$ 0.02	**0.3** $\pm$ 0.02

Table 17: Uniform MAB Instance, $X = 200$

	Data Advantage			Reputation Advantage		
	TS	BEG	BG	TS	BEG	BG
TS	**0.14** $\pm$0.02	**0.18** $\pm$0.02	**0.26** $\pm$0.03	**0.24** $\pm$0.02	**0.2** $\pm$0.02	**0.26** $\pm$0.02
BEG	**0.26** $\pm$0.02	**0.26** $\pm$0.02	**0.34** $\pm$0.03	**0.37** $\pm$0.03	**0.29** $\pm$0.02	**0.31** $\pm$0.02
BG	**0.25** $\pm$0.02	**0.27** $\pm$0.02	**0.34** $\pm$0.03	**0.35** $\pm$0.03	**0.27** $\pm$0.02	**0.3** $\pm$0.02

Table 18: Uniform MAB Instance, $X = 500$

E.4 Mean Reputation vs. Relative Reputation

We present the experiments omitted from Section 5.6. Namely, experiments on the Heavy-Tail MAB instance with $K = 3$ arms, both for "performance in isolation" and the permanent duopoly game. We find that `BayesEpsilonGreedy` $>$ `BayesGreedy` according to the mean reputation trajectory but that `BayesGreedy` $>$ `BayesEpsilonGreedy` according to the relative reputation trajectory *and* in the competition game. As discussed in Section 5.6, the same results also hold for $K = 10$ for the warm starts that we consider.

The result of the permanent duopoly experiment for this instance is shown in Table 19.

	Heavy-Tail		
	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$
`TS vs. BG`	**0.4** ±0.02 `EoG` 770 (0)	**0.59** ±0.01 `EoG` 2700 (2979.5)	**0.6** ±0.01 `EoG` 2700 (3018)
`TS vs. BEG`	**0.46** ±0.02 `EoG` 830 (0)	**0.73** ±0.01 `EoG` 2500 (2576.5)	**0.72** ±0.01 `EoG` 2700 (2862)
`BG vs. BEG`	**0.61** ±0.01 `EoG` 1400 (556)	**0.61** ±0.01 `EoG` 2400 (2538.5)	**0.6** ±0.01 `EoG` 2400 (2587.5)

Table 19: Duopoly Experiment: Heavy-Tail, $K = 3$, $T = 5000$.
Each cell describes a game between two algorithms, call them Alg1 vs. Alg2, for a particular value of the warm start T_0. Line 1 in the cell is the market share of Alg 1: the average (in bold) and the 95% confidence band. Line 2 specifies the "effective end of game" (`EoG`): the average and the median (in brackets).

The mean reputation trajectories for algorithms' performance in isolation and the relative reputation trajectory of `BayesEpsilonGreedy` vs. `BayesGreedy`:

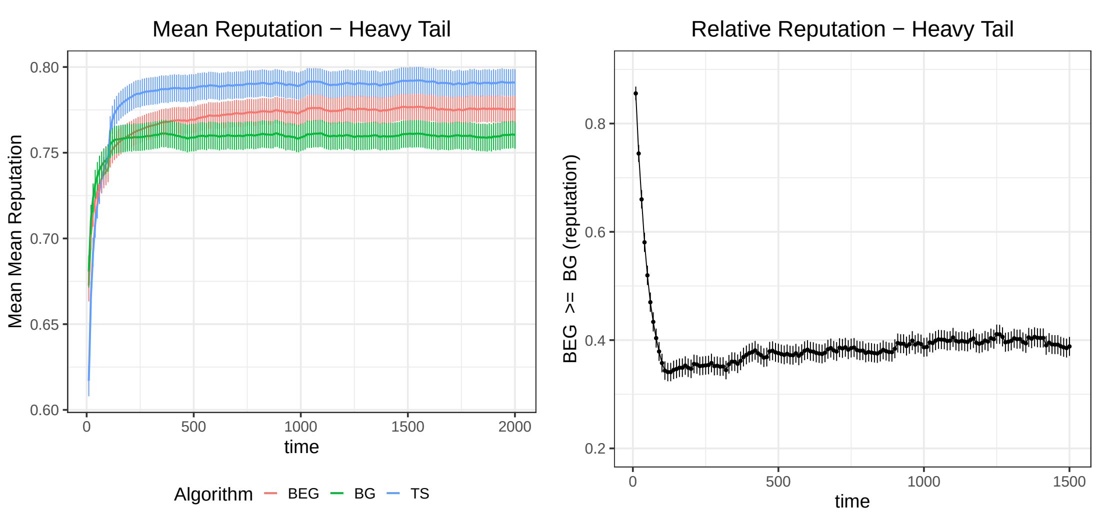

Figure 13: Mean reputation (left) and relative reputation trajectory (right) for Heavy-Tail, $K = 3$