Competing Bandits: The Perils of Exploration under Competition

Abstract

We empirically study the interplay between *exploration* and *competition*. Systems that learn from interactions with users often engage in *exploration*: making potentially suboptimal decisions in order to acquire new information for future decisions. However, exploration may hurt system's reputation in the near term, with adverse competitive effects. In particular, a system may enter a "death spiral" when decreasing market share leaves the system with less users to learn from, which degrades system's performance relative to competition and further decreases the market share.

We ask whether better exploration algorithms are incentivized under competition. We run extensive numerical experiments in a stylized duopoly model in which two firms deploy multi-armed bandit algorithms and compete for myopic users. We find that duopoly and monopoly tend to favor a primitive "greedy algorithm" that does not explore, whereas a temporary monopoly (a duopoly with an early entrant) may incentivize better bandit algorithms. Our findings shed light on the "first-mover advantage" in the digital economy.

1 Introduction

Many modern online platforms simultaneously compete for users as well as learn from the users they manage to attract. This creates a tradeoff between *exploration* and *competition*: firms experiment with potentially sub-optimal options for the sake of gaining information to make better decisions tomorrow, while they need to incentivize consumers to select them over their competitors today. For instance, Google Search and Bing compete for users in the search engine market yet at the same time need to experiment with their search and ranking algorithms to learn what works best.

Platforms routinely deploy A/B tests, and are increasingly adopting more sophisticated exploration methodologies based on *multi-armed bandits*, a well-known framework for exploration and making decisions under uncertainty. While deploying "better" learning algorithms for exploration would improve performance, this is not necessarily beneficial under competition, even putting aside the deployment/maintenance costs. In particular, excessive experimentation may hurt platform's reputation and decrease market share in the near term. This would leave the learning algorithms with less users to learn from, which may further degrade platform's performance relative to competitors who keep learning and improving from *their* users, and so forth.

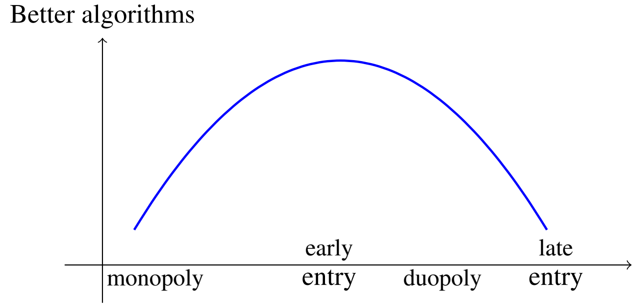

Figure 1: A stylized "inverted-U relationship" between strength of competition and "level of innovation".

We ask whether competition incentivizes adoption of "better" algorithms for exploration. We investigate this issue via extensive numerical experiments in a stylized duopoly model. In our model, two firms compete for users, and simultaneously learn from them. Each firm commits to a multi-armed bandit algorithm, and *explores* according to this algorithm. Users select between the two firms based on the current reputation score: rewards from the firm's algorithm, averaged over a recent time window. Each firm's objective is to maximize its market share (the fraction of users choosing this firm).

We consider a *permanent duopoly* in which both firms start at the same time, as well as *temporary monopoly*: a duopoly with an early entrant. Accordingly, the intensity of competition in the model varies from "permanent monopoly" (just one firm) to "early entry" (in the temporary monopoly) to permanent duopoly to "late entry" (temporary monopoly from the incumbent's perspective). We find that a "greedy algorithm" that does not explicitly explore is most beneficial under duopoly, and even more so under "late entry". This algorithm also prevails under monopoly, simply because it tends to be easier to deploy. Whereas a temporary monopoly incentivizes more advanced exploration algorithms that perform better in the long run. The disincentives to explore under duopoly arise entirely because of "reputational costs", rather than R&D costs (which are absent from our model).

Interpreting the adoption of better algorithms as "innovation", our findings can be framed in terms of an "inverted-U relationship" between competition and innovation (see Figure 1). This relationship – too little or too much competition is bad for innovation, but intermediate levels of competition tend to be better – is a familiar theme in the economics literature, dating back to (Schumpeter 1942).

Our findings on temporary monopoly shed light on the "first-mover advantage" phenomenon in the digital economy. Being first in the market gives firms free data to learn from (a "data advantage") as well as a more definite, and possibly better reputation compared to an entrant (a "reputation advantage"). We investigate which of the two is a stronger barrier to entry. We find that both are strong barriers on their own: removing either one still gives the incumbent a substantially larger share of the market compared to the case where both are removed. However, the data advantage tends to be a stronger barrier when the incumbent commits to a more advanced bandit algorithm. One take-away is that the data advantage is not just about data quantity but also data quality.

We also investigate how algorithms' performance "in isolation" (without competition) is predictive of the outcomes under competition. In particular, we find that mean reputation – arguably, the most natural performance measure "in isolation" – is sometimes not a good predictor. We suggest a more refined performance measure, and use it to explain (and frame) some of the competition outcomes.

Related work. Our work is related to a longstanding economics literature on competition vs. innovation, *e.g.,* (Schumpeter 1942; Barro and Sala-i Martin 2004; Aghion et al. 2005). While this literature focuses on R&D costs of innovation, "reputational costs" seem new and specific to exploration.

Multi-armed bandits (MAB) is a tractable abstraction for the tradeoff between exploration and *exploitation* (making good near-term decisions based on available information). MAB problems have been studied for many decades, see (Bubeck and Cesa-Bianchi 2012) for background. We consider i.i.d. rewards, a well-studied and well-understood MAB model (Auer, Cesa-Bianchi, and Fischer 2002). We focus on a well-known distinction between "greedy" (exploitation-only) algorithms, "naive" algorithms that separate exploration and exploitation, and "smart" algorithms that combine them. Switching from "greedy" to "naive" to "smart" algorithms involves substantial adoption costs in infrastructure and personnel training (Agarwal et al. 2016; 2017).

The study of competition vs. exploration has been initiated in (Mansour, Slivkins, and Wu 2018). Their model differs from ours in two key respects. First, users do not see any signal about firms' past performance, and instead choose between firms according to the Bayesian-expected reward. Second, they vary the strength of competition using assumptions about (ir)rational consumer behavior, whereas we use early entry. Their results are purely theoretical; their model is amenable to proofs but not to numerical experiments. Their high-level conclusion is an inverted-U relationship between competition and innovation that is similar to ours.

The interplay between exploration, exploitation and incentives has been studied in other scenarios: incentivizing exploration in a recommendation system, *e.g.,* (Kremer, Mansour, and Perry 2014; Frazier et al. 2014; Che and Hörner 2018; Mansour, Slivkins, and Syrgkanis 2015; Bimpikis, Papanastasiou, and Savva 2018), dynamic auctions (see (Bergemann and Said 2011) for background), online ad auctions, *e.g.,* (Babaioff, Sharma, and Slivkins 2014; Devanur and Kakade 2009; Nazerzadeh, Saberi, and Vohra 2008; Babaioff, Kleinberg, and Slivkins 2015; Amin, Rostamizadeh, and Syed 2013), and human computation (Ho, Slivkins, and Vaughan 2016; Ghosh and Hummel 2013; Singla and Krause 2013). Our setting is also closely related to the "dueling algorithms" framework (Immorlica et al. 2011), but this framework considers offline / full feedback scenarios whereas we focus on online machine learning problems.

2 Model and Preliminaries

We consider a game involving two firms and T customers (henceforth, *agents*). The game lasts for T rounds. In each round, a new agent arrives, chooses among the two firms, interacts with the chosen firm, and leaves forever.

Each interaction between a firm and an agent proceeds as follows. There is a set A of K actions, henceforth *arms*, same for both firms and all rounds. The firm chooses an arm, and the agent experiences a numerical reward observed by the firm. Each arm corresponds to a different version of the experience that a firm can provide for an agent, and the reward corresponds to the agent's satisfaction level. The other firm does not observe anything about this interaction, not even the fact that this interaction has happened.

From each firm's perspective, the interactions with agents follow the protocol of the multi-armed bandit problem (MAB). We focus on i.i.d. Bernoulli rewards: the reward of each arm a is drawn from $\{0,1\}$ independently with expectation $\mu(a)$. The mean rewards $\mu(a)$ are the same for all rounds and both firms, but but initially unknown.

Before the game starts, each firm commits to an MAB algorithm, and uses this algorithm to choose its actions. Each algorithm receives a "warm start": additional T_0 agents that arrive before the game starts, and interact with the firm as described above. Each firm's objective is to maximize its market share: the fraction of users who chose this firm.

In some of our experiments, one firm is the "incumbent" who enters the market before the other ("entrant"), and therefore enjoys a *temporary monopoly*. 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.

Agents. Agents are myopic and non-strategic: they choose among the firms so as to maximize their expected reward, without attempting to influence the firms' learning algorithms or rewards of the future users. Agents are not well-informed: they only receive a rough signal about each firm's performance before they choose a firm, and no other information.

Concretely, each firm has a *reputation score*, and each agent's choice is driven by these two numbers. We posit a version of rational behavior: each agent chooses a firm with a maximal reputation score (breaking ties uniformly).

The reputation score is simply a sliding window average: an average reward of the last M agents that chose this firm.

MAB algorithms. We consider three classes of algorithms, ranging from more primitive to more sophisticated:

1. *Greedy algorithms* that strive to take actions with maximal mean reward, based on the current information.
2. *Exploration-separating algorithms* that separate exploration and exploitation. The "exploitation" choices strives to maximize mean reward in the next round, and the "exploration" choices do not use the rewards observed so far.
3. *Adaptive exploration*: algorithms that combine exploration and exploitation, and sway the exploration choices towards more promising alternatives.

For concreteness, we fix one algorithm from each class. Our pilot experiments indicate that the results do not change substantially if other algorithms are chosen. For technical reasons, we consider Bayesian versions initialized with a "fake" prior (*i.e.,* not based on actual knowledge). We consider:

1. a greedy algorithm that chooses an arm with largest posterior mean reward. We call it "Dynamic Greedy" (because the chosen arm may change over time), `DG` in short.
2. an exploration-separated algorithm that in each round, *explores* with probability ε: chooses an arm independently and uniformly at random, and with the remaining probability *exploits* according to `DG`. We call it "dynamic epsilon-greedy", `DEG` in short.[1]
3. an adaptive-exploration algorithm called "Thompson Sampling" (`TS`). In each round, this algorithm updates the posterior distribution for the mean reward of each arm a, draws an independent sample s_a from this distribution, and chooses an arm with the largest s_a.

For ease of comparison, all three algorithms are parameterized with the same fake 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.

Both `DEG` and `TS` are classic and well-understood MAB algorithms, see (Bubeck and Cesa-Bianchi 2012; Russo et al. 2018) for background. It is well-known that `TS` is near-optimal in terms of the cumulative rewards, and `DEG` is very suboptimal, but still much better than `DG`.[2] In a stylized formula: `TS` $\gg$ `DEG` $\gg$ `DG` as stand-alone MAB algorithms.

MAB instances. We consider instances with $K = 10$ arms. Since we focus on 0-1 rewards, an instance of the MAB problem is specified by the *mean reward vector* $(\mu(a) : a \in A)$. Initially this vector is 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 $[0.25, 0.75]$.
3. *Heavy-tail instance*: the mean reward of each arm is drawn independently from $\texttt{Beta}(.6, .6)$ distribution.[3]

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, 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.

Terminology. Following a standard game-theoretic terminology, algorithm Alg1 *(weakly) dominates* algorithm Alg2 for a given firm if Alg1 provides a larger (or equal) market share than Alg2 at the end of the game. An algorithm is a (weakly) dominant strategy for the firm if it (weakly) dominates all other algorithms. This is for a particular MAB instance and a particular selection of the game parameters.

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 not due to noise in the realizations but 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 $W(T_{\max} + t, a)$.

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 $T = 2000$.

The simulations are computationally intensive. An experiment on a particular MAB instance comprised multiple runs of the competition game: N mean reward vectors times 9 pairs of algorithms times three values for the warm start. We used a parallel implementation over a cluster of 12 2.2 GHz CPU cores, with 8 GB RAM per core. With this implementation, each experiment took about 10 hours.

While we experiments with various MAB instances and parameter settings, in the main paper we only report plots for selected, representative experiments. Additional plots can

[1] For our experiments, we fix $\varepsilon = 0.05$. Our pilot experiments showed that different ε did not qualitatively change the results.

[2] Formally, `TS` achieves regret $\tilde{O}(\sqrt{TK})$ and $O(\frac{1}{\Delta} \log T)$, where Δ is the gap in mean rewards between the best and second-best arms. `DEG` has regret $\tilde{\Theta}(T^{2/3} K^{1/3})$ in the worst case. And `DG` can have regret as high as $\Omega(T)$. Deeper discussion of these distinctions is not very relevant to this paper.

[3] This distribution has substantial "tail probabilities".

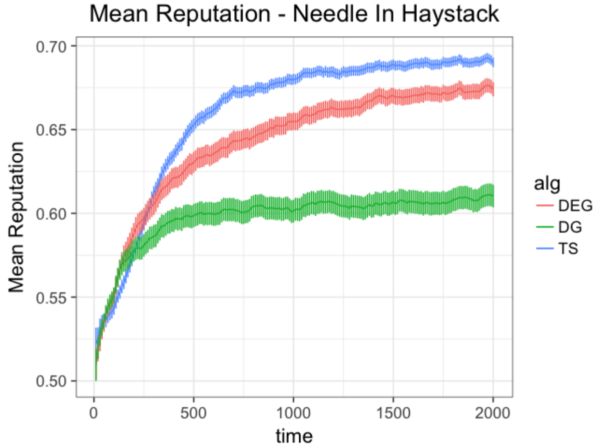

Figure 2: Mean reputation trajectories for Needle-in-Haystack. The shaded area displays 95% confidence intervals.

be found in the supplement. Unless noted otherwise, our findings are based on and consistent with all these plots.

3 Algorithms' Performance in Isolation

We start with a pilot experiment in which we investigate each algorithm's performance "in isolation": in a standalone MAB problem without competition. We focus on reputation scores generated by each algorithm. We confirm that algorithms' performance is ordered as we'd expect: TS $>$ DEG $>$ DG 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. Figure 2 shows such a plot 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 are arranged as predicted by prior work:* TS $>$ DEG $>$ DG *for a sufficiently long time horizon.*

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 3 (top) shows the relative reputation trajectory for TS vs DG for the Uniform instance. The relative reputation is less than $\frac{1}{2}$ in the early rounds, meaning that DG 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 TS leads to worse decisions initially, but allows for better decisions later. The time period when relative reputation vs. DG 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 MAB instance. However, it does not exist for the Needle-in-Haystack instance, see Figure 3 (bottom).[4]

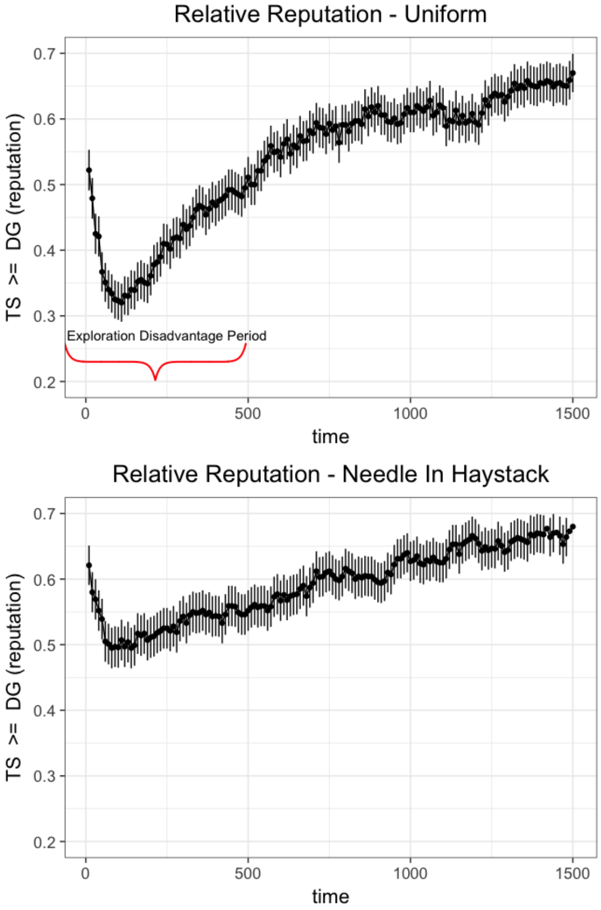

Figure 3: Relative reputation trajectory for TS vs DG, on Uniform instance (top) and Needle-in-Haystack instance (bottom). Shaded area display 95% confidence intervals.

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

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

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

	Heavy-Tail			Needle-in-Haystack		
	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$
TS vs DG	**0.29** ±0.03 EoG 55 (0)	**0.72** ±0.02 EoG 570 (0)	**0.76** ±0.02 EoG 620 (99)	**0.64** ±0.03 EoG 200 (27)	**0.6** ±0.03 EoG 370 (0)	**0.64** ±0.03 EoG 580 (122)
TS vs DEG	**0.3** ±0.03 EoG 37 (0)	**0.88** ±0.01 EoG 480 (0)	**0.9** ±0.01 EoG 570 (114)	**0.57** ±0.03 EoG 150 (14)	**0.52** ±0.03 EoG 460 (79)	**0.56** ±0.02 EoG 740 (628)
DG vs DEG	**0.62** ±0.03 EoG 410 (7)	**0.6** ±0.02 EoG 790 (762)	**0.57** ±0.03 EoG 730 (608)	**0.46** ±0.03 EoG 340 (129)	**0.42** ±0.02 EoG 650 (408)	**0.42** ±0.02 EoG 690 (467)

Table 1: **Permanent duopoly**, for Heavy-Tail and Needle-in-Haystack instances. 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 time horizon is $T = 2000$.

4 Competition and Better MAB Algorithms

Our main experiments are with the duopoly game defined in Section 2. As the "intensity of competition" varies from permanent monopoly to "early entry" to permanent duopoly to "late entry", we find a stylized inverted-U relationship as in Figure 1. More formally, we look for equilibria in the duopoly game, where each firm's choices are limited to DG, DEG and TS. We do this for each "intensity level" and each MAB instance, and look for findings that are consistent across MAB instances. For cleaner results, we break ties towards less advanced algorithms (as they tend to have lower adoption costs (Agarwal et al. 2016; 2017)). Note that DG is trivially the dominant strategy under permanent monopoly.

Permanent duopoly. 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 permanent duopoly:*

(a) (DG,DG) *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, TS *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 Heavy-Tail and Needle-in-Haystack instances in Table 1 (see the first line in each cell), for a range of values of the warm start T_0. Table 2 reports similarly on the Uniform instance. We find that DG is a weakly dominant strategy for the Heavy-tail and Uniform instances, as long as T_0 is sufficiently small. However, TS is a weakly dominant strategy for the Needle-in-Haystack instance. We find that for a sufficiently small T_0, DG yields more than half the market against TS, but achieves similar market share vs. DG and DEG. By our tie-breaking rule, (DG,DG) is the only pure-strategy equilibrium.

We attribute the prevalence of DG on exploration-disadvantaged MAB instances to its prevalence on the initial "exploration disadvantage period", as described in Section 3. Increasing the warm start length T_0 makes this period shorter: indeed, considering relative reputation trajectory in Figure 3 (top), increasing T_0 effectively shifts the starting time point to the right. This is why it helps DG if T_0 is small.

	$T_0 = 20$	$T_0 = 250$	$T_0 = 500$
TS vs DG	**0.46** ±0.03	**0.52** ±0.02	**0.6** ±0.02
TS vs DEG	**0.41** ±0.03	**0.51** ±0.02	**0.55** ±0.02
DG vs DEG	**0.51** ±0.03	**0.48** ±0.02	**0.45** ±0.02

Table 2: **Permanent duopoly**, for the Uniform MAB instance. Semantics are the same as in Table 1.

Temporary Monopoly. We turn our attention to the temporary monopoly 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 TS, in a strong sense:

Finding 4. *Under temporary monopoly,* TS *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 3, for a particular $X = 200$. We see that TS 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	DEG	DG
TS	**0.003**±0.003	**0.083**±0.02	**0.17**±0.02
DEG	**0.045**±0.01	**0.25**±0.02	**0.23**±0.02
DG	**0.12**±0.02	**0.36**±0.03	**0.3**±0.02

Table 3: **Temporary monopoly**, with $X = 200$ (and $T_0 = 20$), for the 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 (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.

[4] We see two explanations for this: TS identifies the best arm faster for the Needle In Haystack instance, and there are no "very bad" arms which make exploration very expensive in the short term.

DG is a weakly dominant strategy for the entrant, for Heavy-Tail instance in Table 3 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 temporary monopoly,* DG *is a weakly dominant strategy for the entrant for exploration-disadvantaged MAB instances.*

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 DG wins out for the trivial reason of tie-breaking. The highest levels are permanent duopoly and "late entry" (temporary monopoly from the entrant's perspective). We see that DG is incentivized for exploration-disadvantaged MAB instances. In fact, incentives for DG get stronger when the model transitions from permanent duopoly to "late entry".[5] Finally, the middle level of competition, "early entry" (temporary monopoly for the entrant) creates strong incentives for TS. In stylized form, this relationship is captured in Figure 1.

Our intuition for why "early entry" creates more incentives for exploration is as follows. During the temporary monopoly period, reputation costs 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 tradeoff, and is incentivized to choose algorithms that are best at optimizing this tradeoff.

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 permanent duopoly scenario, we specify the EoG values in Table 1 (the second line of each cell). We find that the EoG values are indeed small:

Finding 6. *Under permanent duopoly,* EoG *values tend to be much smaller than the time horizon 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 better learning curve). Indeed, we know that the "effective end of game" in this scenario typically occurs when a better algorithm loses, and helping it delays the loss.

5 Data and Reputation as Barriers to Entry

Under temporary monopoly, 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 over the entrant. While both advantages create a strong barrier to entry, we investigate whether either one is a strong barrier alone, and which one is stronger. Our findings provide a quantitative insight into 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 permanent duopoly. The results for the Heavy-Tail instance are presented in Table 4, in the same structure as Table 3. For the other two instances, the results are qualitatively similar.

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 permanent duopoly, for the same pair of algorithms and the same problem instance. In this language, our findings are as follows.

Finding 7. *(a) Data advantage and reputation advantage alone are substantially large, across all algorithms and all MAB instances. (b) The data advantage is larger than the reputation advantage when the incumbent chooses* TS*. (c) The two advantages are similar in magnitude when the incumbent chooses* DEG *or* DG*.*

Our intuition for Finding 7(b) is as follows. Suppose the incumbent switches from DG to TS. 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. One take-away is that the data advantage is not just about data quantity but also data quality.

6 Performance in Isolation, Revisited

We saw in Section 4 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:

[5] For the Heavy-Tail instance, DG goes from a weakly dominant startegy to a strictly dominant one. For the Uniform instance, DG goes from a Nash equilibrium strategy to a weakly dominant one.

	Reputation advantage			Data advantage		
	TS	DEG	DG	TS	DEG	DG
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
DEG	**0.26**±0.03	**0.3**±0.02	**0.26**±0.02	**0.073**±0.01	**0.29**±0.02	**0.25**±0.02
DG	**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.

Conjecture 1. If one algorithm's mean reputation trajectory lies above another, perhaps after some initial time interval (*e.g.,* as in Figure 2), 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 DG attains a larger market share than DEG 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:

Finding 8. *Conjecture 1 is false: mean reputation trajectories do not suffice to explain the outcomes under competition.*

To see what could go wrong with Conjecture 1, 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.[6] For concreteness, consider the needle-in-haystack instance at time $t = 500$, plotted in Figure 4. (The other MAB instances lead to a similar intuition.)

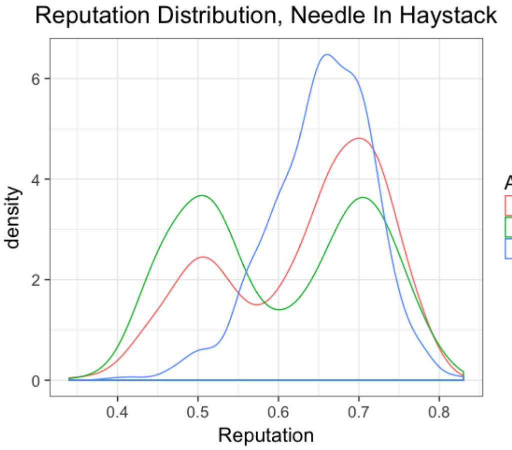

Figure 4: Distribution of reputation scores for Needle-in-Haystack at $t = 500$ (smoothed using a kernel density estimate)

We see that the "naive" algorithms DG and DEG have a bi-modal reputation distribution, whereas TS does not. The reason is that for this MAB instance, DG either finds the best arm and sticks to it, or gets stuck on the bad arms. In the former case DG does slightly better than TS, and in the latter case it does substantially worse. However, the mean reputation trajectory may fail to capture this complexity since it simply takes average over different mean reward vectors. This may be inadequate for explaining the outcome of the duopoly game, given that the latter is determined by a simple comparison between the firm's reputation scores.

To further this intuition, consider the difference in reputation scores (*reputation difference*) between TS and DG 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 5 shows such plots for several time points. We observe that the distribution is skewed to the right, precisely due to the fact that DG either does slightly better than TS or does substantially worse. Therefore, the mean is not a good measure of the central tendency, or typical value, of this distribution.

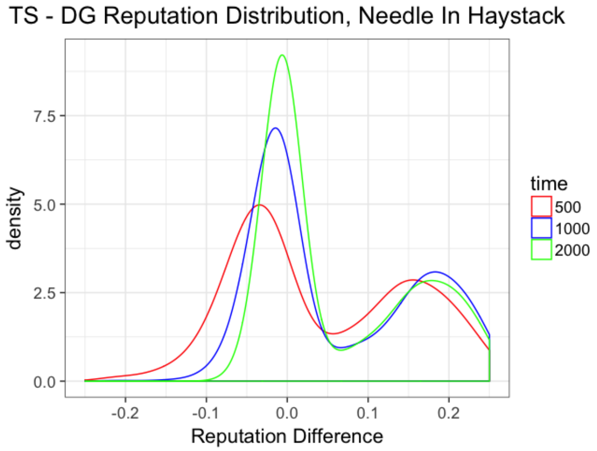

Figure 5: Distribution of reputation difference TS − DG for Needle-in-Haystack (smoothed via a kernel density estimate)

References

Agarwal, A.; Bird, S.; Cozowicz, M.; Dudik, M.; Langford, J.; Li, L.; Hoang, L.; Melamed, D.; Sen, S.; Schapire, R.; and Slivkins, A. 2016. Multi-world testing: A system for experimentation, learning, and decision-making. A white paper, available at

[6] Recall that each mean reward vector in our experimental setup comes with one specific realization table.

https://github.com/Microsoft/mwt-ds/raw/master/images/MWT-WhitePaper.pdf.

Agarwal, A.; Bird, S.; Cozowicz, M.; Hoang, L.; Langford, J.; Lee, S.; Li, J.; Melamed, D.; Oshri, G.; Ribas, O.; Sen, S.; and Slivkins, A. 2017. Making contextual decisions with low technical debt. Techical report at arxiv.org/abs/1606.03966.

Aghion, P.; Bloom, N.; Blundell, R.; Griffith, R.; and Howitt, P. 2005. Competition and innovation: An inverted u relationship. *Quaterly J. of Economics* 120(2):701–728.

Amin, K.; Rostamizadeh, A.; and Syed, U. 2013. Learning prices for repeated auctions with strategic buyers. In *26th NIPS*, 1169–1177.

Auer, P.; Cesa-Bianchi, N.; and Fischer, P. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine Learning* 47(2-3):235–256.

Babaioff, M.; Kleinberg, R.; and Slivkins, A. 2015. Truthful mechanisms with implicit payment computation. *Journal of the ACM* 62(2):10. Subsumes the conference papers in *ACM EC 2010* and *ACM EC 2013*.

Babaioff, M.; Sharma, Y.; and Slivkins, A. 2014. Characterizing truthful multi-armed bandit mechanisms. *SIAM J. on Computing* 43(1):194–230. Preliminary version in *10th ACM EC*, 2009.

Barro, R. J., and Sala-i Martin, X. 2004. Economic growth: Mit press. *Cambridge, Massachusettes*.

Bergemann, D., and Said, M. 2011. Dynamic auctions: A survey. In *Wiley Encyclopedia of Operations Research and Management Science, Vol. 2*. Wiley: New York. 1511–1522.

Bimpikis, K.; Papanastasiou, Y.; and Savva, N. 2018. Crowdsourcing exploration. *Management Science*. Forthcoming. Published online as *Articles in Advance* in April 2017.

Bubeck, S., and Cesa-Bianchi, N. 2012. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. *Foundations and Trends in Machine Learning* 5(1).

Che, Y.-K., and Hörner, J. 2018. Optimal design for social learning. *Quarterly Journal of Economics*. Forthcoming. First published draft: 2013.

Devanur, N., and Kakade, S. M. 2009. The price of truthfulness for pay-per-click auctions. In *10th ACM EC*, 99–106.

Frazier, P.; Kempe, D.; Kleinberg, J. M.; and Kleinberg, R. 2014. Incentivizing exploration. In *ACM EC*, 5–22.

Ghosh, A., and Hummel, P. 2013. Learning and incentives in user-generated content: multi-armed bandits with endogenous arms. In *ITCS*, 233–246.

Ho, C.-J.; Slivkins, A.; and Vaughan, J. W. 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*.

Immorlica, N.; Kalai, A. T.; Lucier, B.; Moitra, A.; Postlewaite, A.; and Tennenholtz, M. 2011. Dueling algorithms. In *43rd ACM STOC*, 215–224.

Kerin, R. A.; Varadarajan, P. R.; and Peterson, R. A. 1992. First-mover advantage: A synthesis, conceptual framework, and research propositions. *The Journal of Marketing* 33–52.

Kremer, I.; Mansour, Y.; and Perry, M. 2014. Implementing the wisdom of the crowd. *J. of Political Economy* 122:988–1012. Preliminary version in *ACM EC 2014*.

Mansour, Y.; Slivkins, A.; and Syrgkanis, V. 2015. Bayesian incentive-compatible bandit exploration. In *15th ACM EC*.

Mansour, Y.; Slivkins, A.; and Wu, S. 2018. Competing bandits: Learning under competition. In *9th ITCS*.

Nazerzadeh, H.; Saberi, A.; and Vohra, R. 2008. Dynamic cost-per-action mechanisms and applications to online advertising. In *17th WWW*.

Russo, D.; Roy, B. V.; Kazerouni, A.; Osband, I.; and Wen, Z. 2018. A tutorial on thompson sampling. *Foundations and Trends in Machine Learning* 11(1):1–96.

Schumpeter, J. 1942. *Capitalism, Socialism and Democracy*. Harper & Brothers.

Singla, A., and Krause, A. 2013. Truthful incentives in crowdsourcing tasks using regret minimization mechanisms. In *22nd WWW*, 1167–1178.