Received: 3 June 2022 | Accepted: 1 November 2022

DOI: 10.1111/poms.13919

ORIGINAL ARTICLE

Data-driven collusion and competition in a pricing duopoly with multinomial logit demand

Thomas Loots[1] | **Arnoud V. den Boer**[1,2]

[1] Korteweg-de Vries Institute for Mathematics, University of Amsterdam, Amsterdam, The Netherlands

[2] Amsterdam Business School, University of Amsterdam, Amsterdam, The Netherlands

Correspondence
Thomas Loots, Korteweg-de Vries Institute for Mathematics, University of Amsterdam, 1098 XG Amsterdam, The Netherlands.
Email: t.loots@uva.nl

Handling Editor: Stefanus Jasin

Funding information
Nederlandse Organisatie voor Wetenschappelijk Onderzoek, Grant/Award Number: OCENW.KLEIN.016

Abstract

We consider dynamic pricing and demand learning in a duopoly with multinomial logit demand, both from the perspective where firms *compete* against each other and from the perspective where firms aim to *collude* to increase revenues. We show that joint-revenue maximization is not always beneficial to both firms compared to the Nash equilibrium, and show that several other axiomatic notions of collusion can be constructed that are always beneficial to both firms and a threat to consumer welfare. Next, we construct a price algorithm and prove that it learns to charge supra-competitive prices if deployed by both firms, and learns to respond optimally against a class of competitive algorithms. Our algorithm includes a mechanism to infer demand observations from the competitor's price path, so that our algorithm can operate in a setting where prices are public but demand is private information. Our work contributes to the understanding of well-performing price policies in a competitive multi-agent setting, and shows that collusion by algorithms is possible and deserves the attention of lawmakers and competition policy regulators.

KEYWORDS
algorithmic collusion, competition, demand learning, dynamic pricing

1 | INTRODUCTION

1.1 | Background and motivation

The mathematical analysis of dynamic price algorithms in the presence of demand learning is at the forefront of contemporary research, undoubtedly inspired by the significant potential for implementation in practice. A vast and rapidly increasing stream of literature analyzes such price algorithms (see the references discussed in Subsection 1.3). Most of these settings are monopolistic, and do not take the intricacies of pricing and learning in the presence of competitors into account. This motivates the current work, in which we study pricing and learning in a duopoly.

An important aspect of duopoly pricing is the absence of an unequivocal notion of what it means to price optimally. In most monopolistic models, there exists a unique optimal price if model parameters remain fixed throughout the time horizon. For duopolies, however, there exists a spectrum of equilibria, with Nash equilibria on one side and supra-competitive equilibria on the other side. In this paper, we consider price policies for both sides of the spectrum.

The potential convergence of price algorithms to supra-competitive equilibria has recently drawn the attention of competition regulators worldwide (Hoffman, 2018; OECD, 2017b; Spiridonova & Juchnevicius, 2020). In particular, the question has emerged whether price algorithms are capable of learning supra-competitive prices from data (specifically in the scenario where multiple firms in a market use the same price algorithm), within the boundaries of existing competition law. Among other things, this means that the algorithms cannot make use of illicit forms of signaling or communication. Some argue that algorithmic collusion poses a fundamental threat to consumer welfare (Ezrachi & Stucke, 2016, 2017a, 2017b, 2017c, 2020), whereas others believe

Accepted by Stefanus Jasin, after one revision.

that algorithmic collusion belongs to the realm of "science fiction" (Kühn & Tadelis, 2017; Schrepel, 2017; Schwalbe, 2018; Veljanovski, 2020). If algorithmic collusion is possible, then the existing legal framework of competition law might need substantial revision or re-interpretation (Ezrachi & Stucke, 2016, 2017a, 2017b, 2017c, 2020; Gal, 2019; Harrington, 2018; OECD, 2017b).

A challenge in achieving algorithmic collusion is the fact that, without communication, there might be ambiguity in the firms' understanding of what it means to collude. But as shown by Meylahn and den Boer (2022), a definition of collusion can be built into the algorithm; an "algorithmic agreement," as it were (cf. Gal, 2019, p. 78). This is a powerful tool for creating stable cartels under mutual use of the algorithm, as it prevents bargaining and deviation issues that often result in the collapse of collusion (Bos & Harrington, 2010, pp. 107–108).

A second challenge is the fact that collusive prices usually depend on parameters that are *unknown* and thus have to be *learned* from sales data. This task is further complicated by the fact that prices are often publicly observable, but sales transactions are not: firms typically do not have access to the sales data of their competitor (see Meylahn and den Boer (2022) for exceptions). This hinders estimation of the notion of collusion implemented in the algorithm. It is therefore necessary to study if and how algorithms can overcome the challenge of private demand observations.

A third challenge that arises when hard-coding a notion of collusion into a price algorithm is defining an appropriate collusive solution. Collusion is often associated with joint-revenue maximization (Gal, 2019; OECD, 2017b; Schwalbe, 2018), but this notion may not perform well for individual firms in asymmetric settings without side payments (Fischer & Normann, 2019). In such cases, algorithmic collusion requires a notion of collusion that is always beneficial to all firms.

The purpose of this paper is to address these three challenges in the context of a pricing duopoly. We show that several axiomatic notions of collusion can be constructed that are beneficial to both firms and detrimental to consumer welfare. Next, we construct an algorithm that learns the collusive price when deployed by both players in the market and learns to respond optimally against a class of competitive algorithms. We furthermore propose a mechanism to overcome the challenge of private demand observations by exploiting the information that is conveyed by the algorithm's price decisions. This algorithm can generate supra-competitive prices without breaking existing competition law, and therefore shows that the concerns of competition regulators are justified.

1.2 | Legal environment

We consider a duopoly where both firms independently adopt the same price algorithm. In the absence of communication, the firms should not be in violation of the law even if mutual adoption of the algorithm leads to supra-competitive outcomes. This is in accordance with findings of the OECD (2017a, p. 10): "Absent concerted action, independent adoption of the same or similar pricing algorithms is unlikely to lead to antitrust liability even if it makes interdependent pricing more likely." Ezrachi and Stucke (2020, p. 220) derive similar conclusions. Along these lines, Harrington (2018, pp. 339–340) makes a very clear distinction between collusion as an *outcome* and the *means* by which the outcome is achieved: "Supracompetitive prices are legal. Collusion is legal. It is the process by which firms achieve collusion that is illegal, rather than the state of collusion itself. […] At a minimum, there must be evidence of communication to allow a court to conclude that firms have not acted independently. Though all forms of collusion are harmful, some forms are legal because of the inability to effectively distinguish collusive conduct from competitive conduct".

If competing firms purchase the algorithm from the same software supplier who boasts the collusive capabilities of the algorithm, then the supplier runs the risk of being seen as a "middle man" who facilitates collusion and could potentially be held liable. The OECD (2019) has collected many real-life cases of this type of collusion, so-called hub-and-spoke collusion, where coordination does not occur through direct exchanges between the horizontal competitors (the "spokes"), but through exchanges via a vertically related supplier or retailer (the "hub"). Of particular interest is the case of Partneo (see, e.g., Mandrescu, 2018): between 2008 and 2013 five carmakers colluded by using the same car part pricing software, Partneo, from the company Accenture. They boosted their revenues by more than 1 billion dollars thanks to using Partneo, and yet "no formal findings […] have been found against the carmakers or Accenture" (Ezrachi & Stucke, 2020, p. 220). The OECD (2019, p. 2) states that "[t]he main challenge for enforcement agencies is to identify when inherently legitimate exchanges between suppliers and retailers turn into a prohibited horizontal co-ordination, without direct proof of collusion. Enforcement and jurisprudence in particular in the United States, but also in Europe, […] require proof of a horizontal connection between the spokes, a rim, and an awareness of all actors involved". In the scenario that we consider in this paper, there are no exchanges between the competitors via the software supplier. This means that a "rim" (a connection between competitors) can only be established by the workings of the algorithm itself. In Section EC.3 of the Supporting Information, we explain that it is inevitable that prices generated by an algorithm contain and transfer information. This information transfer is an "accidental" and unavoidable by-product of the generated price decisions that have legitimate alternative justifications. This further suggests that users (and possibly distributors) of the algorithm do not violate competition laws.

Although it is not settled whether a software supplier can sell the same algorithm to competing firms without violating laws, this problem does not arise in other scenarios. Consider the situation of multiple third-party sellers who sell their products on an online platform. Suppose, in addition, that the computer code of a collusive algorithm has been made available on an online software repository. The collusive capabilities of the algorithm are not mentioned; all that gradually becomes known through word-of-mouth rumors is that the algorithm appears to increase profits substantially. It is conceivable that in this situation multiple sellers start using the algorithm, without engaging in illicit communication with each other or with a "middle man." This scenario becomes increasingly likely due to the rising popularity of platforms that connect third-party sellers to consumers, and of open-source software repositories. There are many software suppliers who offer repricing algorithms to third-party sellers on various platforms, and generally charge a fee that increases with the number of listings. There could thus be much interest in free-to-download pricing software.

We emphasize that we do not and cannot claim with certainty that the algorithm proposed in this paper (or the distribution thereof) is legal, because that can only be decided in court. For the same reason it is also not clear that the algorithm is *illegal*. Regulators are well-aware that algorithms can facilitate tacit collusion by reducing or eliminating the need for human communication (OECD, 2017b, p. 7), and that signaling between algorithms can occur through prices: "For instance, firms may program snapshot price changes during the middle of the night, which won't have any impact on sales but may be identified as a signal by rivals' algorithms. Or, alternatively, companies may use algorithms to publicly disclose a lot of detailed data that is used as a code to propose and negotiate price increases, as it was observed in the US case on Airline Tariff Publishing Company" (OECD, 2017b, p. 30). An important insight of this paper is that information can be transferred between algorithms in much *subtler* ways. Indeed, we concretely show that prices generated by algorithms reveal information that can be used to achieve supra-competitive outcomes, not because extraneous information is explicitly *encoded* in the prices, but because legitimate price decisions inevitably allow *decoding* of information by a copy of the same algorithm. This creates a challenge for lawmakers: information transfer through prices is inevitable, but it is far from trivial to determine where to draw the line.

1.3 | Related literature

Recently, a large number of studies have appeared on dynamic pricing with demand learning (Besbes & Zeevi, 2009, 2015; Broder & Rusmevichientong, 2012; Chen et al., 2019, 2021; Cheung et al., 2017; den Boer & Zwart, 2014, 2015; Hansen et al., 2021; Harrison et al., 2012; Keskin & Zeevi, 2014; Wang et al., 2014, 2021). We refer to den Boer (2015) for a recent survey. Although these papers consider an impressive number of models and settings, the formal analysis of price algorithms with demand learning in the presence of competitors has received relatively little attention in this stream of literature. Exceptions include Bertsimas and Perakis (2006), who formulate an optimal pricing problem with incomplete information in a dynamic programming framework and propose tractable approximations; Cooper et al. (2015), who study the limit behavior of prices in a duopoly where both players use a mis-specified passive learning policy; and Yang et al. (2020), who study convergence to equilibrium prices under the assumption that all players in the market use the same algorithm. Our paper contributes to this stream of literature by proposing price policies and formally proving regret bounds in a duopoly, both in a competitive and in a collusive setting.

Our work contributes to the nascent field of algorithmic collusion. A key question in this field is up to what extent algorithms are capable of learning supra-competitive prices under self-play while simultaneously learning to price competitively against reasonable alternative algorithms. Recent contributions include Calvano et al. (2020), Cooper et al. (2015), and Klein (2021), where we remark that the focus of the second paper is not algorithmic collusion per se but rather the interplay of learning and model mis-specification. These studies show by means of simulations that particular price policies, when playing against copies of itself, can lead to supra-competitive limit prices and revenues with positive probability. However, as illustrated by figures 6 and 7 of Cooper et al. (2015), sub-competitive prices and revenues are also possible outcomes. In addition, these papers do not provide performance guarantees in case these policies do *not* play against a copy of itself but against a "reasonable" alternative strategy. These two observations diminish the likelihood of such algorithms being implemented in practice (den Boer et al., 2022).

Two examples where the notion of collusion is explicitly programmed in the algorithm are Meylahn and den Boer (2022) and Aouad and den Boer (2021). Meylahn and den Boer (2022) explain that pre-programmed collusion does not make an algorithm illegal. The authors consider a pricing duopoly, define collusion as maximizing the joint-revenue function, and construct a policy that learns this collusive price provided this is mutually beneficial to the firms and both firms use the algorithm. In addition, it is shown that prices converge to a best response in case the opponent prices according to a reaction function. Aouad and den Boer (2021) consider similar questions in an assortment-optimization duopoly with multinomial logit demand.

A limitation of Meylahn and den Boer (2022) is that collusion is only guaranteed if the joint-revenue-maximizing prices are mutually beneficial compared to pricing under the Nash equilibrium. In practice this is often not the case if the firms are not symmetric (Bos & Harrington, 2010; Fischer & Normann, 2019). A second limitation is that demand observations are assumed to be public information. These limitations might give lawmakers the wrong impression that algorithmic collusion is only possible in markets where these two limitations do not apply. In contrast, in this work we show and explicitly construct notions of collusion that firms can adopt that are *always* mutually beneficial. Hard-coding these cartel prices in algorithms enables collusion in many more markets than only (near-)symmetric ones. In addition, we show how the challenge of private demand information can be overcome by inferring demand data from public price data.

This paper also contributes to the axiomatic bargaining literature by elaborating how firms can collude on price in a pricing duopoly with multinomial logit demand. To the best of our knowledge, there appears to be no existing work that conducts this analysis. We take an axiomatic approach in the spirit of Roth (1979). Fischer and Normann (2019) study similar concepts for Cournot games.

1.4 | Contributions and insights

The technical contributions of this paper are threefold. First, we analyze in detail three notions of collusion in a pricing duopoly with multinomial logit demand functions, based on different sets of axioms such as Pareto optimality and equal relative/absolute gains. We prove existence and uniqueness under different combinations of axioms, and show how these collusive prices can be computed up to arbitrary precision. We show that these collusive prices reduce consumer welfare and increase prices and revenues of both firms compared to the Nash equilibrium. Simulation results with parameter values based on empirical data show that these effects can be substantial.

Second, we construct a price algorithm and prove that prices converge to the implemented collusive solution if the algorithm is used by both players. Importantly, the players do not have to start the algorithm at the same moment, demand observations are allowed to be private information, and the players do not have to know that the opponent is using the same algorithm. If the algorithm plays against a competitive price rule, we prove that prices converge to a best response and provide asymptotically optimal regret bounds.

Third, we show how a competitor's private demand realizations can be inferred from public prices. This facilitates consistent estimation of the unknown model parameters and has the additional benefit that the players, under mutual use of our proposed algorithm, always base their decisions on the same sales data set. This enables players to immediately detect when the other firm deviates from the algorithm's price suggestions, strengthening the stability of the cartel.

Our technical results have several implications for competition regulation professionals. First, we show that two major challenges of firms to achieve algorithmic collusion can be overcome when they use the same algorithm. Hard-coding a notion of collusion in an algorithm removes the need to communicate and agree on what it means to collude, which, according to Bos and Harrington (2010), is a critically difficult step in the formation of cartels. Furthermore, using the same algorithm enables the firms to "understand" each other's price decisions, so that, for example, price experiments aimed at guaranteeing learning the model parameters will not mistakenly be interpreted as "cheating" the cartel. In fact, deviation from the prices prescribed by the algorithm will immediately be detected by the competing firm, which strengthens the stability of the cartel.

Second, our examples of collusive solutions demonstrate that firms can be creative in what they mean by collusion, and can adopt definitions that are not necessarily on the radar of a regulator. Competition regulators should therefore not blindly focus on one particular notion of collusion (e.g., joint-revenue maximization) but be on the lookout for any sustainable price pair that reduces consumer welfare and is beneficial for the firms.

Third, our mechanism to infer sales data from price paths reveals an important point related to communication: *price paths contain and transfer information*. Illicit communication between firms to establish collusion is forbidden by law (Harrington, 2018), but the fact that prices may signal information about, for example, a seller's parameter estimates or beliefs cannot always be avoided, and therefore does not constitute illegal communication per se. In our algorithm, information transfer is an "accidental" by-product of the generated price decisions, which is used to overcome the challenge of private sales data and to learn the collusive price. Competition regulators may wish to reflect on whether this information transfer is a desirable property of prices generated by algorithms.

1.5 | Organization of the paper

The rest of this paper is organized as follows. Subsection 2.1 describes the static pricing game under consideration. In Subsection 2.2, we consider several axiomatic notions of collusion, and prove results on existence and uniqueness. Subsection 2.3 provides comparisons with Nash prices, revenues, and consumer welfare. Subsection 3.1 extends the static game to a dynamic setting in which the demand functions are unknown upfront, and defines the concept of a price policy. Subsection 3.2 describes the algorithm that we propose. In Section 4, we analyze its performance against different types of competitors. Subsection 4.1 establishes that prices converge to the built-in collusive solution when the algorithm is used by both firms in the duopoly, and provides a bound for the corresponding regret. In Subsection 4.2, we study the performance of our algorithm when pricing against a "true" competitor, and prove an asymptotically optimal regret bound. Subsection 4.3 provides numerical illustrations. A discussion and directions for future research are contained in Section 5. All mathematical proofs are collected in the Supporting Information, where we also discuss legal aspects of the proposed algorithm (Section EC.3 in the Supporting Information), consider the performance of alternative price rules against our policy (Section EC.4 in the Supporting Information), and explain that cheating the algorithm is difficult (Section EC.5 in the Supporting Information).

2 | COLLUSION IN A STATIC PRICING DUOPOLY

In this section, we describe different definitions of collusion that firms could devise and implement in a price algorithm. Subsection 2.1 describes the static pricing game under consideration. Subsection 2.2 establishes that joint-revenue maximization, a standard notion of collusion, is often not beneficial to both firms and therefore not always a suitable definition of collusion. We discuss alternative notions of collusion based on different axioms and prove that these are always beneficial to both firms and always diminish consumer welfare. Subsection 2.3 provides numerical results.

2.1 | Static model

We consider a price-setting duopoly where each firm sells exactly one product; firm one sells product one, firm two sells product two. The price charged for product $j \in \{1, 2\}$ is denoted by $p_j \geq 0$. We assume that the customer's purchase behavior follows the well-known multinomial logit model, which means that the customer assigns a (random) utility

$$U_j(p_j; \theta) := a_j - b_j p_j + \epsilon_j \tag{1}$$

to each product j, and a (random) utility $U_0 := \epsilon_0$ to the no-purchase option, where $\theta = (a_1, b_1, a_2, b_2)$ is an underlying parameter from the parameter space $\Theta := \mathbb{R} \times (0, \infty) \times \mathbb{R} \times (0, \infty)$, and where ϵ_0, ϵ_1, and ϵ_2 are independent, mean zero Gumbel random variables with scale parameter one. Among the three purchase options, the customer chooses the one with the highest utility, with ties broken uniformly at random (note that ties occur with probability zero).

The exponential $v_j(p_j; \theta) := \exp(a_j - b_j p_j)$ of the expected utility of product j is called its *attraction value*. The attraction value of the no-purchase option equals 1. As derived by, for example, Gallego and Topaloglu (2019, p. 113), product j is purchased with probability

$$\lambda_j(\mathbf{p}; \theta) := \frac{v_j(p_j; \theta)}{1 + v_1(p_1; \theta) + v_2(p_2; \theta)}, \tag{2}$$

and a no-purchase takes place with probability $\lambda_0(\mathbf{p}; \theta) := 1 - \lambda_1(\mathbf{p}; \theta) - \lambda_2(\mathbf{p}; \theta)$, where $\mathbf{p} = (p_1, p_2)$ denotes the pair of both prices. We set marginal costs to zero, so the (expected) revenue for firm j as a function of the charged prices is given by $r_j(\mathbf{p}; \theta) := \lambda_j(\mathbf{p}; \theta) p_j$.

For future reference, we now define the optimal monopoly prices (i.e., when there is no competitor), the Nash equilibrium prices, and the maximizer of the joint-revenue function. Regarding the monopoly prices, Li and Huh (2011, Theorem 2 and Corollary 1) show that if firm j is a monopolist—which is equivalent to setting the competitor's price to infinity—there exists a unique revenue-maximizing price

$$p_j^{\text{MONO}}(\theta) := \frac{W_0(\exp(a_j - 1)) + 1}{b_j}, \tag{3}$$

where W_0 denotes the 0th branch of the Lambert W function (see Section EC.1 in the Supporting Information for details). The optimal monopoly revenue for firm j is given by $r_j^{\text{MONO}}(\theta) = W_0(\exp(a_j - 1))/b_j$.

Li and Huh (2011, Theorem 4) also show that the static pricing game has a unique Nash equilibrium $\mathbf{p}^{\text{NE}}(\theta) = (p_1^{\text{NE}}(\theta), p_2^{\text{NE}}(\theta))$ given by

$$p_j^{\text{NE}}(\theta) := \frac{1}{b_j} \cdot \frac{1}{1 - \lambda_j^{\text{NE}}(\theta)}, \tag{4}$$

where $\lambda_j^{\text{NE}}(\theta) := V(\lambda_0^{\text{NE}}(\theta) \cdot \exp(a_j - 1))$ denotes the purchase probability for product j under the Nash equilibrium, and where the corresponding no-purchase probability $\lambda_0^{\text{NE}}(\theta)$ is the unique solution $z \in (0, 1)$ to the equation $z + V(z \cdot \exp(a_1 - 1)) + V(z \cdot \exp(a_2 - 1)) = 1$; here, for $x > 0$, $V(x)$ is the unique solution $z \in (0, 1)$ to the equation $z \cdot \exp(z/(1 - z)) = x$. The expected revenue for firm j under the Nash equilibrium price pair is denoted by $r_j^{\text{NE}}(\theta)$.

Finally, Li and Huh (2011, Theorem 2) show that the joint-revenue function $\mathbf{p} \mapsto r_1(\mathbf{p}; \theta) + r_2(\mathbf{p}; \theta)$ has a unique maximizer $\mathbf{p}^{\text{JRM}}(\theta) = (p_1^{\text{JRM}}(\theta), p_2^{\text{JRM}}(\theta))$ given by

$$p_j^{\text{JRM}}(\theta) := u + \frac{1}{b_j}, \tag{5}$$

where u is the unique solution to the equation $u = \exp(a_1 - 1 - b_1 u)/b_1 + \exp(a_2 - 1 - b_2 u)/b_2$. The expected revenue for firm j under the joint-revenue-maximizing price pair is denoted by $r_j^{\text{JRM}}(\theta)$.

We conclude this section with the following remarks regarding notation. For each $j \in \{1, 2\}$, the notation $\neg j$ denotes the only element in the set $\{1, 2\} \setminus \{j\}$, that is, represents the competitor of firm j. Also, for all $x, y \in \mathbb{R}$, we use the notation

$$[x, y]_j := \begin{cases} (x, y) & \text{if } j = 1, \\ (y, x) & \text{if } j = 2. \end{cases} \tag{6}$$

In some proofs, we omit the argument θ for the underlying parameter if θ is fixed within that context, that is, we write $r_j(\mathbf{p})$ instead of $r_j(\mathbf{p}; \theta)$, p_j^{NE} instead of $p_j^{\text{NE}}(\theta)$, and so forth. Finally, price pairs are understood to be elements of $[0, \infty)^2$ unless stated otherwise.

2.2 | Notions of collusion

2.2.1 | Joint-revenue maximization

Collusion is frequently associated with—or even defined as—joint-revenue maximization (Gal, 2019, p. 71, OECD, 2017b, p. 19, Schwalbe, 2018, p. 569). However, the concepts of collusion and joint-revenue maximization are fundamentally different: from the perspective of an individual firm, the objective of collusion is to raise its *own* revenue to a supra-competitive level, not to maximize joint revenue per se.

Although maximizing joint revenue can sometimes increase revenues for all involved firms, this need not always be the case.

To illustrate this point, we generate 1 million instances of $\theta = (a_1, b_1, a_2, b_2)$ and verify whether $r_j^{\mathrm{JRM}}(\theta) < r_j^{\mathrm{NE}}(\theta)$ for some $j \in \{1, 2\}$, where a_1 and a_2 are drawn uniformly at random from the interval $[-1, 5]$, and where b_1 and b_2 are drawn uniformly at random from the interval [0.001, 0.019]. These sampling intervals are in line with the empirical work of Li et al. (2019), as well as the simulation framework of van de Geer and den Boer (2022)—which in turn is "in line with the empirical work of Delahaye et al. (2017) […] and the simulation framework of Rayfield et al. (2015)". It turns out that in all except 19% of the parameter instances, the joint-revenue-maximizing price pair causes revenue losses compared to the Nash equilibrium for one of the two firms (see the rightmost column of Table 1). These losses are especially large in asymmetric markets—for example, when a_1 and b_2 are high, and a_2 and b_1 are low.

This shows that in many cases (asymmetric) firms that are in principle willing to collude are unlikely to act on it if collusion is defined by charging joint-revenue-maximizing prices. This does not mean that such firms cannot collude: in what follows we show that there are alternative ways to define cartel prices that are always profitable for both firms.

2.2.2 | Axiomatic notions of collusion

We refer to any function $\psi : \Theta \to [0, \infty)^2$ as a *notion of collusion*, which can be based on a variety of favorable axioms from the perspective of the firms. We consider the following axioms.

AXIOM 1 (Mutually beneficial). *For all $\theta \in \Theta$: $r_j(\psi(\theta); \theta) > r_j^{\mathrm{NE}}(\theta)$ for each $j \in \{1, 2\}$.*

AXIOM 2 (Pareto optimality). *For all $\theta \in \Theta$, there exists no price pair $\mathbf{x}$ such that $r_j(\mathbf{x}; \theta) \geq r_j(\psi(\theta); \theta)$ for each $j \in \{1, 2\}$, with strict inequality for at least one j.*

AXIOM 3 (Equal relative gains). *For all $\theta \in \Theta$: $r_1(\psi(\theta); \theta)/r_1^{\mathrm{NE}}(\theta) = r_2(\psi(\theta); \theta)/r_2^{\mathrm{NE}}(\theta)$.*

AXIOM 4 (Equal absolute gains). *For all $\theta \in \Theta$: $r_1(\psi(\theta); \theta) - r_1^{\mathrm{NE}}(\theta) = r_2(\psi(\theta); \theta) - r_2^{\mathrm{NE}}(\theta)$.*

Mutually beneficial notions of collusion provide strictly higher revenue than the Nash equilibrium, for each firm, regardless of the model parameters (in contrast to, for example, joint-revenue maximization). Pareto optimality ensures that no additional revenue can be gained without harming the other firm. The axioms of equal relative/absolute gains prevent imbalances in the collusive surplus compared to the Nash equilibrium, thereby making the notion of collusion more mutually attractive.

In this section, we consider three notions of collusion that firms could construct. Two of these definitions are induced by taking different combinations of the axioms given above: the first notion of collusion is defined by Axioms 2 and 3, the second notion by Axioms 2 and 4. The axiomatic derivation of the third notion of collusion, the Nash bargaining solution, is given in Section EC.2 in the Supporting Information because the underlying axioms are based on a broader framework. All three notions of collusion under consideration are mutually beneficial (Axiom 1) and Pareto optimal (Axiom 2).

Before we analyze these notions of collusion, we prove a technical result about Pareto optimal *price pairs*; a price pair $\mathbf{p}$ is called Pareto optimal for a parameter $\theta \in \Theta$ if there exists no price pair $\mathbf{x}$ such that $r_j(\mathbf{x}; \theta) \geq r_j(\mathbf{p}; \theta)$ for each $j \in \{1, 2\}$, with strict inequality for at least one j. The following lemma characterizes the set of all Pareto optimal price pairs as the graph of a decreasing function on an open interval, and establishes that there do not exist two different Pareto optimal price pairs that give a firm the same amount of revenue. The proof mostly revolves around showing that Pareto optimal price pairs that provide a certain amount of revenue for one of the two firms are unique maximizers of certain functions.

Lemma 1. *Let $\theta = (a_1, b_1, a_2, b_2) \in \Theta$ and $j \in \{1, 2\}$. A price pair $\mathbf{p}$ is Pareto optimal for θ if and only if both $p_j > p_j^{\mathrm{MONO}}(\theta)$ and $p_{\neg j} = f_{\neg j}^{\mathrm{PO}}(p_j; \theta)$, where, for $x \in (p_j^{\mathrm{MONO}}(\theta), \infty)$,*

$$x \mapsto f_{\neg j}^{\mathrm{PO}}(x; \theta) := \frac{1}{b_{\neg j}} \left(W_0 \left(\frac{b_j x - 1}{b_j x - 1 - \exp(a_j - b_j x)} \cdot \exp(a_{\neg j} - 1) \right) + 1 \right) \tag{7}$$

is a function that strictly decreases from ∞ to $p_{\neg j}^{\mathrm{MONO}}(\theta)$ on its domain. Furthermore, for any $\tilde{r} \in (0, r_j^{\mathrm{MONO}}(\theta))$ there exists a unique price pair $\mathbf{p}$ with $r_j(\mathbf{p}; \theta) = \tilde{r}$ that is Pareto optimal for θ.

We now consider the three aforementioned axiomatic notions of collusion in detail. Our first theorem shows that the axioms of Pareto optimality and equal relative gains uniquely define a notion of collusion, denoted by ψ^{RPO} (R for relative, PO for Pareto optimal), and establishes that ψ^{RPO} is smooth and mutually beneficial. Because each function value $\psi^{\mathrm{RPO}}(\theta)$ is the unique zero of a strictly decreasing function, it can be computed to arbitrary precision with a bisection algorithm.

Theorem 1. *There exists a unique notion of collusion ψ^{RPO} that satisfies both Axiom 2 (Pareto optimality) and Axiom 3 (equal relative gains). For all $\theta \in \Theta$ and each $j \in \{1, 2\}$, it holds that $\psi_j^{\mathrm{RPO}}(\theta)$ is the unique zero of the continuous* *function*

$$x \mapsto f_j^{\mathrm{RPO}}(x;\theta) := \frac{r_j\Big([x, f_{\neg j}^{\mathrm{PO}}(x;\theta)]_j;\theta\Big)}{r_j^{\mathrm{NE}}(\theta)} - \frac{r_{\neg j}\Big([x, f_{\neg j}^{\mathrm{PO}}(x;\theta)]_j;\theta\Big)}{r_{\neg j}^{\mathrm{NE}}(\theta)}, \tag{8}$$

which strictly decreases from a positive value to a negative value on its domain $(p_j^{\mathrm{MONO}}(\theta), \infty)$*. Moreover,* ψ^{RPO} *is infinitely differentiable and mutually beneficial* (*Axiom* 1).

To prove the theorem, we deduce from Lemma 1 that the revenue of firm j strictly decreases from the optimal monopoly value to zero along the graph of the function defined in (7), whereas the opponent's revenue strictly increases in the reversed direction. Consequently, the function defined in (8) strictly decreases from a positive value to a negative value. This guarantees the existence of a unique zero, which in turn ensures existence and uniqueness of a Pareto optimal notion of collusion that provides equal relative gains. Because the Nash equilibrium is not Pareto optimal it follows that ψ^{RPO} is mutually beneficial. Smoothness follows from two subsequent applications of the implicit function theorem; first to show infinite differentiability of the function $\theta \mapsto \mathbf{p}^{\mathrm{NE}}(\theta)$, and then of ψ^{RPO}.

Our next result shows that the axioms of Pareto optimality and equal absolute gains uniquely define a notion of collusion, denoted by ψ^{APO} (A for absolute, PO for Pareto optimal), and establishes that ψ^{APO} is smooth and mutually beneficial. Function values can again be approximated to arbitrary precision using a bisection algorithm. The proof of Theorem 2 is of a similar structure as the proof of Theorem 1.

Theorem 2. *There exists a unique notion of collusion* ψ^{APO} *that satisfies both Axiom* 2 (*Pareto optimality*) *and Axiom* 4 (*equal absolute gains*). *For all* $\theta \in \Theta$ *and each* $j \in \{1, 2\}$ *it holds that* $\psi_j^{\mathrm{APO}}(\theta)$ *is the unique zero of the continuous function*

$$x \mapsto f_j^{\mathrm{APO}}(x;\theta) := \Big(r_j\Big([x, f_{\neg j}^{\mathrm{PO}}(x;\theta)]_j;\theta\Big) - r_j^{\mathrm{NE}}(\theta)\Big) - \Big(r_{\neg j}\Big([x, f_{\neg j}^{\mathrm{PO}}(x;\theta)]_j;\theta\Big) - r_{\neg j}^{\mathrm{NE}}(\theta)\Big), \tag{9}$$

which strictly decreases from a positive value to a negative value on its domain $(p_j^{\mathrm{MONO}}(\theta), \infty)$*. Moreover,* ψ^{APO} *is infinitely differentiable and mutually beneficial* (*Axiom* 1).

The third and final notion of collusion that we consider is the Nash bargaining solution, denoted by ψ^{NB}, where, for all $\theta \in \Theta$, $\psi^{\mathrm{NB}}(\theta)$ is defined as the maximizer of the function

$$\mathbf{p} \mapsto f^{\mathrm{NB}}(\mathbf{p};\theta) := \big(r_1(\mathbf{p};\theta) - r_1^{\mathrm{NE}}(\theta)\big)\big(r_2(\mathbf{p};\theta) - r_2^{\mathrm{NE}}(\theta)\big) \tag{10}$$

on the domain $\mathcal{P}(\theta) := \{\mathbf{p} \in [0,\infty)^2 : r_1(\mathbf{p};\theta) > r_1^{\mathrm{NE}}(\theta)$ and $r_2(\mathbf{p};\theta) > r_2^{\mathrm{NE}}(\theta)\}$. This notion of collusion is uniquely defined by assuming Pareto optimality, symmetry, and the independence of irrelevant alternatives property (see Section EC.2 in the Supporting Information for details). Theorem 3 establishes that ψ^{NB} is well-defined, and characterizes every function value $\psi^{\mathrm{NB}}(\theta)$ as the unique maximizer of a strictly log-concave function on a convex domain, which implies that it can be computed to arbitrary precision using a gradient ascent method. The theorem also establishes smoothness properties and that the Nash bargaining solution satisfies Axioms 1 and 2.

Theorem 3. *Let* $\theta \in \Theta$*. There exists a unique price pair that maximizes the function* $\mathbf{p} \mapsto f^{\mathrm{NB}}(\mathbf{p};\theta)$ *on* $\mathcal{P}(\theta)$*. The domain* $\mathcal{P}(\theta)$ *is a* (*nonempty*) *convex set, and the function* $\mathbf{p} \mapsto f^{\mathrm{NB}}(\mathbf{p};\theta)$ *is strictly log-concave on* $\mathcal{P}(\theta)$*. The notion of collusion* ψ^{NB} *is infinitely differentiable, mutually beneficial* (*Axiom* 1)*, and Pareto optimal* (*Axiom* 2).

To prove Theorem 3, we first establish that the revenue functions $\mathbf{p} \mapsto r_j(\mathbf{p};\theta)$ are strictly log-concave by explicitly computing the Hessians and showing that they are negative definite. This allows us to derive that the function $\mathbf{p} \mapsto f^{\mathrm{NB}}(\mathbf{p};\theta)$ is strictly log-concave, and that its domain $\mathcal{P}(\theta)$ is convex; these properties in turn ensure existence and uniqueness of the maximizer. We then recall that the Nash equilibrium is not Pareto optimal, which implies that $\mathcal{P}(\theta)$ is nonempty, and thus shows that ψ^{NB} is mutually beneficial. Pareto optimality of ψ^{NB} follows immediately from the observation that a Pareto improvement of $\psi^{\mathrm{NB}}(\theta)$ would give a strictly higher value of f^{NB}. Finally, infinite differentiability of ψ^{NB} follows by applying the implicit function theorem to the gradient of the logarithm of f^{NB}.

2.3 | Effects of collusion on prices, revenues, and consumer welfare

We measure consumer welfare (CW) by the expected post-purchase utility obtained by the consumer:

$$\mathrm{CW}(\mathbf{p};\theta) := \mathbb{E}[\max\{U_0, U_1(p_1;\theta), U_2(p_2;\theta)\}], \tag{11}$$

for any underlying parameter $\theta \in \Theta$ and any price pair $\mathbf{p}$ charged by the firms. Our next result shows that Pareto optimal price pairs are detrimental to consumers because they are component-wise strictly higher than the Nash equilibrium price pair, and strictly reduce consumer welfare compared to the Nash equilibrium. Consequently, Pareto optimal notions of collusion are a threat to consumers; this includes the three examples from Subsection 2.2.2.

Theorem 4. *Let* $\theta \in \Theta$*. For any price pair* $\mathbf{p}$*, it holds that* $\mathrm{CW}(\mathbf{p};\theta) = -\ln(\lambda_0(\mathbf{p};\theta))$*. Moreover, if* $\mathbf{p}$ *is Pareto optimal for* θ*, then* $p_j > p_j^{\mathrm{NE}}(\theta)$ *for each* $j \in \{1, 2\}$*, and* $\mathrm{CW}(\mathbf{p};\theta) < \mathrm{CW}(\mathbf{p}^{\mathrm{NE}}(\theta);\theta)$.

TABLE 1 Performance results of four different notions of collusion compared to the Nash equilibrium

Notion of collusion	RPO	APO	NB	JRM
Mutually profitable	100%	100%	100%	19%
Average increase in prices	49%	46%	46%	124%
Average increase in revenues	14%	18%	17%	0%
Average decrease in consumer welfare	30%	32%	31%	32%

In the proof of Theorem 4, we derive that $\mathrm{CW}(\mathbf{p};\theta)$ is Gumbel distributed with mean $-\ln(\lambda_0(\mathbf{p};\theta))$ and scale parameter one. We then show that the optimal monopoly prices are strictly higher than the Nash equilibrium prices, which means that Pareto optimal price pairs are componentwise strictly higher than the Nash equilibrium price pair due to Lemma 1. The result regarding reduced consumer welfare directly follows because the no-purchase probability is a strictly increasing function of both prices.

To obtain more insight into the effects of collusion on prices, revenues, and consumer welfare, we now perform a simulation. We generate 1 million instances of $\theta = (a_1, b_1, a_2, b_2)$, with a_1 and a_2 drawn uniformly at random from the interval $[-1, 5]$, and with b_1 and b_2 drawn uniformly at random from the interval [0.001,0.019]. For each instance we compute the relative increase in prices and revenues, and the relative decrease in consumer welfare (compared to the Nash equilibrium) under RPO, APO, NB, and JRM; Table 1 reports the averages.

The results from Table 1 show that RPO, APO, and NB have roughly equivalent effects on prices, revenues, and consumer welfare, whereas JRM performs rather differently. First, JRM is mutually profitable compared to the Nash equilibrium in only 19% of the cases; this is in stark contrast with the 100% for the other collusive solutions. Second, the average effect on revenues is 0%; this is because JRM is not a mutually beneficial notion of collusion, so that revenues can in fact decrease by switching from the Nash equilibrium to JRM. Third, the average effect of JRM on prices is much more extreme than for the other notions of collusion, but because these enormous price increases generally do not occur for both firms simultaneously, consumer welfare is affected by roughly the same amount as the other notions of collusion.

3 | COLLUSION AND COMPETITION IN A DYNAMIC PRICING DUOPOLY

We now consider a sequential version of the pricing game where the parameters are initially *unknown* to the firms. Subsection 3.1 extends the model from Section 2 to this dynamic setting and provides the definition of a policy. In Subsection 3.2, we propose an algorithm called *Collude-or-Compete* that guarantees convergence to a built-in notion of collusion when used by both players, and performs optimally against a class of alternative policies that a competitor might deploy.

3.1 | Dynamic model

We consider a dynamic version of the static duopoly game introduced in Subsection 2.1. Although the firms can in principle change their selling prices at any instant, we divide the time horizon into periods and model the firms' pricing behavior as a discrete-time simultaneous-move game, as is common in the literature (see, e.g., Akçay et al., 2010, Cooper et al., 2015, Li & Huh, 2011, Maglaras & Meissner, 2006). This means that we divide the time horizon $[0, \infty)$ into subsequent periods $[t-1, t)$ of equal length, where the time unit is chosen small enough to ensure that the probability of two or more customers arriving in the same period is negligible, and assume that both players update their prices at the start of each period (cf. Talluri & van Ryzin, 2004, p. 58).

Let $p_j(s)$ be the price charged by firm $j \in \{1, 2\}$ during the sth period $[s-1, s)$, for $s \in \mathbb{N}$, and write $\mathbf{p}(s) := (p_1(s), p_2(s))$ for the corresponding price pair. Demand again follows the multinomial logit model, that is, if $\theta \in \Theta$ is the underlying parameter, the probability that a unit of product j is purchased during the sth period, given the price pair $\mathbf{p}(s)$, is given by $\lambda_j(\mathbf{p}(s);\theta)$. The probability of a no-purchase is given by $\lambda_0(\mathbf{p}(s);\theta)$. The (expected) revenue for firm j during the sth period, given the price pair $\mathbf{p}(s)$, is given by $r_j(\mathbf{p}(s);\theta)$. There exists a "true" underlying parameter θ^*, which is *unknown* to both firms but lies in a known convex and compact set $\Xi \subset \Theta$. Finally, let $\psi : \Theta \to [0, \infty)^2$ be a notion of collusion that is Lipschitz-continuous on Ξ, for example one of the notions discussed in Subsection 2.2.2.

Let $\mathcal{F}_{0,j}$ be the trivial σ-algebra. Denote the demand for product j during the sth period by $d_j(s) \in \{0, 1\}$, and let $\mathcal{F}_{s,j} := \sigma(\mathbf{p}(1), d_j(1), \ldots, \mathbf{p}(s), d_j(s))$ be the σ-algebra generated by the price pairs and firm j's demand observations up to the sth period. We call $(p_j(s))_{s\in\mathbb{N}}$ a price policy or algorithm for firm j if $p_j(s)$ is $\mathcal{F}_{s-1,j}$-measurable for all $s \in \mathbb{N}$.

3.2 | Collude-or-compete

We propose a dynamic price algorithm called Collude-or-Compete, which by design is able to switch between two policies, or *modules*. One of these policies (the *collusive* policy) is designed to guarantee convergence of prices to the collusive solution $\psi(\theta^*)$ if used by both firms and activated simultaneously, and is described in Subsection 3.2.1. Use of this policy requires demand information of *both* firms; we describe in Subsection 3.2.2 how the competitor's (private) demand realizations can be extracted from (public) prices. Subsection 3.2.3 introduces the other policy (the *competitive* policy), which is designed to optimize revenue when playing against a wide range of policies that an opponent could deploy. Subsection 3.2.4 describes how the Collude-or-Compete algorithm switches between the two modules based on the competitor's price path, and shows how simultaneous activation of the collusive module is guaranteed if the algorithm is used by both firms. Subsection 3.2.5 provides a comprehensive formulation of the proposed algorithm.

3.2.1 | Collusive policy

The collusive policy is based on ideas from den Boer and Zwart (2014). The key idea is to iteratively approximate θ^* using maximum-likelihood estimation and charge the estimated collusive price, possibly with a perturbation added to ensure consistency of the parameter estimates.

In particular, if firm j activates the collusive policy at time $\tau \geq 0$, it starts by charging $p_j(\tau+1) = p_{\text{start},j}$ and $p_j(\tau+2) = \kappa_1 p_{\text{start},j}$, where the initial price $p_{\text{start},j} > 0$ is an input parameter and where $\kappa_1 \in (0,1)$. For every subsequent period $t \geq \tau+3$, the collusive policy uses the maximizer $\hat{\theta}_{t-1}$ of the likelihood function

$$L_{t-1}(\theta) := \prod_{s=\tau+1}^{t-1} \lambda_1(\mathbf{p}(s);\theta)^{d_1(s)} \lambda_2(\mathbf{p}(s);\theta)^{d_2(s)} \lambda_0(\mathbf{p}(s);\theta)^{1-d_1(s)-d_2(s)} \tag{12}$$

on Ξ; the following lemma establishes existence and uniqueness.

Lemma 2. *Let $t \geq \tau+3$. If $p_k(\tau+1) \neq p_k(\tau+2)$ for each $k \in \{1,2\}$, then there exists a unique maximizer of L_{t-1} on Ξ.*

To ensure consistency of the parameter estimates, the policy imposes for every $t \geq \tau+2$ that the sample variance of firm j's prices satisfies

$$\text{Var}(p_j(\tau+1), \ldots, p_j(t)) \geq c_j (t-\tau)^{\kappa_0}, \tag{13}$$

where $\kappa_0 \in (-1,0)$ and $c_j := 2^{-2-\kappa_0}(p_{\text{start},j})^2(\kappa_1-1)^2$, and where the sample variance is defined by

$$\text{Var}(p_j(\tau+1), \ldots, p_j(t)) := \frac{1}{t-\tau}\sum_{s=\tau+1}^{t} p_j(s)^2 - \left(\frac{1}{t-\tau}\sum_{s=\tau+1}^{t} p_j(s)\right)^2. \tag{14}$$

It achieves this by charging the estimated collusive price $\psi_j(\hat{\theta}_{t-1})$, for all $t \geq \tau+3$, unless this would imply that the lower bound (13) on the sample variance of the prices is not satisfied. In that case, the next price should be chosen not too close to the average of the prices charged thus far. To this end, we define a *taboo interval* around the average price, and perturb the estimated collusive price if it falls within the taboo interval. In particular, the taboo interval is given by

$$TI_j(t) := \left(\frac{1}{t-\tau-1}\sum_{s=\tau+1}^{t-1} p_j(s) - \mu_j(t), \frac{1}{t-\tau-1}\sum_{s=\tau+1}^{t-1} p_j(s) + \mu_j(t)\right), \tag{15}$$

where the half-width $\mu_j(t)$ is given by

$$\mu_j(t) := \sqrt{c_j\left((t-\tau)^{1+\kappa_0} - (t-\tau-1)^{1+\kappa_0}\right)\frac{t-\tau}{t-\tau-1}}. \tag{16}$$

The price that the collusive policy charges for firm j during the tth period, for $t \geq \tau+3$, is given by[1]

$$\text{DCVP}_j(t) := \psi_j(\hat{\theta}_{t-1}) + \delta_j(t), \tag{17}$$

where the price perturbation $\delta_j(t)$ is given by

$$\delta_j(t) := 2\mu_j(t)\mathbb{1}\{\psi_j(\hat{\theta}_{t-1}) \in \text{TI}_j(t)\}. \tag{18}$$

Remark 1. The values of κ_0 and κ_1 are hard-coded into the algorithm by the software developer, cannot be altered by the user, and are the same for all users. We show in Subsection 4.1 that the choice $\kappa_0 = -1/2$ leads to asymptotically optimal performance. The choice of κ_1 can be market-dependent.

3.2.2 | Extracting demand information from prices

Computing the prices determined by the collusive policy requires demand observations from *both* firms. We explain in this section how the competitor's demand realizations can be inferred from their price path. The key feature underlying our inference process is that different demand observations by the competitor correspond to different selling prices. Using the same price algorithm as the competitor enables one to "reverse engineer" the (private) demand data that has resulted in the (public) selling prices. By continuously applying this procedure, both users of the algorithm ensure that they base their decisions on the same data set consisting of price *and* demand data. As a result, from an informational viewpoint, the situation is then equivalent to the situation where demand information is publicly available.

We now explain in more detail the mechanism of demand inference from price paths that is incorporated in the Collude-or-Compete algorithm. This mechanism utilizes that both firms use the algorithm and are *synchronized*, that is, that they simultaneously activate the collusive module at some time τ—we explain in Subsection 3.2.4 that synchronization is guaranteed per construction of the algorithm. The mechanism is based on the following idea. If firm j observes demand $d_j(t) = 1$, then she knows with certainty that $d_{\neg j}(t) = 0$. On the other hand, if firm j observes no demand, that is, $d_j(t) = 0$, then both $d_{\neg j}(t) = 0$ and $d_{\neg j}(t) = 1$ are possible. For both of these cases, firm j computes the corresponding maximum-likelihood estimate and the corresponding prices that the collusive policy would set for each firm,[2] and sets her own price $p_j(t+1)$ assuming that $d_{\neg j}(t) = 0$. After observing $p_{\neg j}(t+1)$, firm j immediately adapts her price if $p_{\neg j}(t+1)$ reveals that $d_{\neg j}(t) = 1$. We now describe the mechanism in more detail.

At the beginning of each period $t+1$, where $t \geq \tau + 3$, firm j computes the maximizers $\hat{\theta}_{t,j}^{(0,0)}$, $\hat{\theta}_{t,j}^{(0,1)}$, $\hat{\theta}_{t,j}^{(1,0)}$ of the likelihood function L_t on Ξ, corresponding to each of the three demand possibilities $(d_j(t), d_{\neg j}(t)) \in \{(0,0),(0,1),(1,0)\}$. She also computes the prices that the collusive policy would set, $(p_j^{(0,0)}(t+1), p_{\neg j}^{(0,0)}(t+1))$, $(p_j^{(0,1)}(t+1), p_{\neg j}^{(0,1)}(t+1))$, $(p_j^{(1,0)}(t+1), p_{\neg j}^{(1,0)}(t+1))$, corresponding to each of these three possibilities. If $d_j(t) = 1$, then firm j sets $p_j(t+1) = p_j^{(1,0)}(t+1)$. If $d_j(t) = 0$, then firm j assumes that $d_{\neg j}(t) = 0$ and sets $p_j(t+1) = p_j^{(0,0)}(t+1)$; but, once firm j observes that $p_{\neg j}(t+1) = p_{\neg j}^{(0,1)}(t+1)$, firm j infers that $d_{\neg j}(t) = 1$ and immediately changes her price to $p_j(t+1) = p_j^{(0,1)}(t+1)$. In this way, firm j can reconstruct $d_{\neg j}(t)$ for all $t \geq \tau + 3$.

The competitor's demand during periods $\tau + 1$ and $\tau + 2$ are inferred from $p_{\neg j}(\tau+3)$ in a similar manner. In particular, before determining $p_j(\tau+3)$, firm j computes the maximizers $\hat{\theta}_{\tau+2,j}^{(x_1,y_1),(x_2,y_2)}$ of the likelihood function $L_{\tau+2}$ on Ξ, now corresponding to each of the *nine* demand possibilities $(d_j(\tau+1), d_{\neg j}(\tau+1)) = (x_1, y_1)$ and $(d_j(\tau+2), d_{\neg j}(\tau+2)) = (x_2, y_2)$, where $(x_1,y_1),(x_2,y_2) \in \{(0,0),(0,1),(1,0)\}$. She also computes the prices that the collusive policy would set, $(p_j^{(x_1,y_1),(x_2,y_2)}(\tau+3), p_{\neg j}^{(x_1,y_1),(x_2,y_2)}(\tau+3))$, corresponding to each of these nine possibilities. Then, at the start of period $\tau+3$, firm j assumes that $(d_{\neg j}(\tau+1), d_{\neg j}(\tau+2)) = (0,0)$ and sets $p_j(\tau+3) = p_j^{(d_j(\tau+1),0),(d_j(\tau+2),0)}(\tau+3)$; but, once firm j observes that $p_{\neg j}(\tau+3) = p_{\neg j}^{(0,y_1),(0,y_2)}(\tau+3)$ for some $(y_1,y_2) \in \{(1,1),(0,1),(1,0)\}$ with $d_j(\tau+1) + y_1 \leq 1$ and $d_j(\tau+2) + y_2 \leq 1$, firm j infers that $(d_{\neg j}(\tau+1), d_{\neg j}(\tau+2)) = (y_1, y_2)$ and immediately changes her price to $p_j(\tau+3) = p_j^{(d_j(\tau+1),y_1),(d_j(\tau+2),y_2)}(\tau+3)$.

This mechanism, to infer (private) demand information from (public) selling prices, hinges on two assumptions: a technical and a mathematical one. On the technical side, we need to assume that the time (say, $\varepsilon \in (0,1)$ time units) required to observe the competitor's price and potentially adapt one's price in response to this observation, is sufficiently small such that the probability of a customer arrival during $[t, t+\varepsilon)$ is negligible. Whether this is possible in practice is difficult to determine without being engaged in actual implementation; at the same time, it is not obvious that it is technically impossible, now or in the near future. Investigating the technical feasibility of this scheme is therefore an important direction for future research, but outside the scope of this paper.

The second assumption underlying our demand inference scheme is that different demand observations correspond to different prices. Formally stated, the mechanism works if, with probability one, $p_{\neg j}^{(0,0)}(t+1) \neq p_{\neg j}^{(0,1)}(t+1)$ for all $t \geq \tau + 3$, as well as $p_{\neg j}^{(0,y_1),(0,y_2)}(\tau+3) \neq p_{\neg j}^{(0,\tilde{y}_1),(0,\tilde{y}_2)}(\tau+3)$ for all distinct $(y_1,y_2),(\tilde{y}_1,\tilde{y}_2) \in \{(0,0),(0,1),(1,0),(1,1)\}$. We have unfortunately not been able to formally prove this property, and we believe that this constitutes a relevant question for future research. On the other hand, throughout all our simulations we observed not a single instance that violated this condition. We conjecture that the required conditions needed to infer demand from price paths are satisfied, and in the rest of the paper we assume that this is the case.

3.2.3 | Competitive policy

The competitive policy is based on ideas from Broadie et al. (2011). The idea is to maximize the (unknown) revenue function using a stochastic gradient ascent method introduced by Kiefer and Wolfowitz (1952), which applies price perturbations around a so-called pivot price to obtain a local estimate of the derivative, and updates the pivot price accordingly. Choosing appropriate perturbation sizes and step sizes between subsequent pivot prices ensures convergence of prices to the maximizer of the revenue function.

In particular, from the perspective of firm j, the competitive policy divides the time horizon into consecutive cycles, each cycle consisting of two subsequent time periods: in the nth cycle, for $n \in \mathbb{N}$, it charges the price $p_j^{(n)} + \beta_{n,j}$ during the first period, and $p_j^{(n)} - \beta_{n,j}$ during the second period, where $p_j^{(n)}$ is called the nth *pivot* price. The price perturbations are given by

$$\beta_{n,j} := K_{\beta,j}/n^{1/4}, \tag{19}$$

where $K_{\beta,j}$ is a positive constant. We require that $K_{\beta,j} < p_{\max,j}/2$, where

$$p_{\max,j} := \sup_{p \geq 0, \theta \in \Xi} \{\mathrm{BR}_j(p;\theta)\} \tag{20}$$

determines a price ceiling for the competitive policy, and where, for all $\theta = (a_1, b_1, a_2, b_2) \in \Theta$ and $p \geq 0$,

$$\mathrm{BR}_j(p;\theta) := \frac{1}{b_j}\left(W_0\left(\frac{\exp(a_j - 1)}{1 + \exp(a_{\neg j} - b_{\neg j}p)}\right) + 1\right) \tag{21}$$

denotes the *best-response* price of firm j given that her opponent charges the price p (see Section EC.12 in the Supporting

Information for the derivation). The demand observations from these two periods, say, periods t and $t+1$, are used to compute the next pivot price

$$p_j^{(n+1)} := \Pi_{n+1}\left(p_j^{(n)} + \alpha_{n,j}\left(\frac{(p_j^{(n)} + \beta_{n,j})d_j(t) - (p_j^{(n)} - \beta_{n,j})d_j(t+1)}{\beta_{n,j}}\right)\right), \tag{22}$$

where $\Pi_{n+1,j}$ denotes the Euclidean projection function onto the interval $[\beta_{n+1,j}; p_{\max,j} - \beta_{n+1,j}]$, and where the "step sizes" are given by

$$\alpha_{n,j} := K_{\alpha,j}/n, \tag{23}$$

with $K_{\alpha,j}$ a positive constant. The projection step ensures that the prices charged by the policy are contained in the interval $[0, p_{\max,j}]$. The constants $K_{\alpha,j}$ and $K_{\beta,j}$, as well as the initial pivot price $p_j^{(1)} \in [\beta_{1,j}; p_{\max,j} - \beta_{1,j}]$, are input parameters that can be chosen by firm j.

3.2.4 | Switching between modules and achieving synchronization

In this section, we explain how the Collude-or-Compete algorithm switches between the collusive policy and the competitive policy based on the opponent's price path. We speak of the collusive/competitive "module" instead of "policy" when we view it as part of the overarching Collude-or-Compete algorithm, and not as a stand-alone policy.

The algorithm always starts by running the collusive module; but, as soon as the opponent charges a price that cannot be the output of the collusive module, the algorithm immediately switches to the competitive module. The first time that the competitive module is activated, it runs $K_{\gamma,j}$ Kiefer–Wolfowitz cycles (for firm j), where $K_{\gamma,j} \in \mathbb{N}$ is an input parameter. After that, the algorithm (temporarily) switches back to the collusive module. The goal of this is to make collusion possible if the opponent recently started using the Collude-or-Compete algorithm. But, again, when the opponent deviates from the prices determined by the collusive module, the algorithm immediately switches back to the competitive module, continuing with the $(n+1)$th cycle if it last performed the nth cycle. The mth time that the competitive module is activated it runs $mK_{\gamma,j}$ cycles before activating the collusive module again. The "batch size" is increased in this manner to ensure that the collusive module runs less frequently over time (relative to the competitive module) against an opponent that does not use the same algorithm.

If both firms use the Collude-or-Compete algorithm, their collusive modules need to be activated *simultaneously* in order for prices to converge to the collusive solution $\psi(\theta^*)$. To achieve this, the algorithm can activate the collusive module prematurely, overriding the aforementioned routine of switching between modules. On a high level the mechanism works as follows. Suppose that both firms use the Collude-or-Compete algorithm, consider the perspective of firm j, and let the mth *phase* be the time interval between the mth and $(m+1)$th moment in time that firm $\neg j$ activates the collusive module. The start of a phase is marked by two prices charged subsequently by firm $\neg j$ that have a ratio of κ_1, and concludes with a certain number of Kiefer–Wolfowitz cycles.[3] Because these prices are highly structured, firm j can recognize when her opponent performs a phase, but does not know the duration of the phase in advance. However, after observing two subsequent phases, firm j can infer from the opponent's price path that the opponent ran mk and $(m+1)k$ Kiefer–Wolfowitz cycles during the subsequent phases, in that order, for some $m, k \in \mathbb{N}$. This reveals to firm j that $K_{\gamma,\neg j} = k$, and therefore reveals that the next phase contains $(m+2)k$ Kiefer–Wolfowitz cycles. This enables firm j to exactly determine the starting time of phase $m+3$ in advance, allowing her to activate the collusive module at exactly this time.

Formally, the mechanism is as follows. Suppose that there exist $t, n \geq 0$, $K > 0$, and $m, k \in \mathbb{N}$ such that

$$\frac{p_{\neg j}(t+2)}{p_{\neg j}(t+1)} = \kappa_1, \quad \text{and} \quad \frac{p_{\neg j}(t+4+2mk)}{p_{\neg j}(t+3+2mk)} = \kappa_1, \quad \text{and}$$
$$\frac{p_{\neg j}(t+6+2(2m+1)k)}{p_{\neg j}(t+5+2(2m+1)k)} = \kappa_1; \tag{24}$$

$$\frac{p_{\neg j}(t+2+2s-1) - p_{\neg j}(t+2+2s)}{(n+s)^{-1/4}} = K, \ \forall s \in \{1, \ldots, mk\}; \tag{25}$$

$$\frac{p_{\neg j}(t+4+2mk+2s-1) - p_{\neg j}(t+4+2mk+2s)}{(n+mk+s)^{-1/4}} = K, \quad \forall s \in \{1, \ldots, (m+1)k\}; \tag{26}$$

$$\frac{p_{\neg j}(t+6+2(2m+1)k+2s-1) - p_{\neg j}(t+6+2(2m+1)k+2s)}{(n+(2m+1)k+s)^{-1/4}} = K, \quad \forall s \in \{1, \ldots, (m+2)k\}; \tag{27}$$

then firm j activates the collusive module at time $t+6+2(3m+3)k$.

The first equality in (24) indicates activation of the collusive module by firm $\neg j$ at time t, and (25) indicates mk Kiefer–Wolfowitz cycles starting at time $t+2$, concluding phase m. Similarly, phase $m+1$ corresponds to the second inequality in (24) combined with (26); this phase starts at time $t+2+2mk$ and ends at time $t+4+2(2m+1)k$. At time $t+6+2(2m+1)k$, two periods after the end of the previous phase, firm j has observed the third activation of the collusive module by firm $\neg j$ (from time t onward), therefore knows that two phases have been completed, and is thus able to infer the values of k, m and that $K_{\gamma,\neg j} = k$. At this point, firm j knows that her opponent will now run $(m + 2)k$ Kiefer–Wolfowitz cycles (which can be confirmed using (27)) and will activate the collusive module at time $t + 6 + 2(3m + 3)k$.

In the rest of the paper, we suppose that the synchronization mechanism is implemented in full generality in a function $\texttt{SYNC}_j$; for all $t \geq 0$, $\texttt{SYNC}_j(t)$ equals 1 if the synchronization mechanism dictates that firm j should activate the collusive module at time t, and equals 0 otherwise.

3.2.5 | Comprehensive formulation of the algorithm

In the preceding sections, we have described in detail the individual components of our algorithm. In the current section, we synthesize these into a compact description of the complete algorithm. We take the perspective of firm j. In the description we write $p_{j,\varepsilon}(s)$ for the price charged by firm j during $[s - 1, s - 1 + \varepsilon)$ and $p_j(s)$ for the price charged during $[s - 1 + \varepsilon, s)$, for all $s \geq 0$, where ε denotes the time required to observe the competitor's price and potentially adapt one's price in response to this observation (see Subsection 3.2.2).

Collude-or-Compete algorithm $\text{CC}_j(p_{\text{start},j}; p_j^{(1)}; K_{\alpha,j}; K_{\beta,j}; K_{\gamma,j})$

(A) Initialize parameters and state variables.

Let $K_{\gamma,j} \in \mathbb{N}$ and $p_{\text{start},j}, p_j^{(1)}, K_{\alpha,j}, K_{\beta,j} > 0$ such that $K_{\beta,j} < p_{\max,j}/2$ and $p_j^{(1)} \in [K_{\beta,j}; p_{\max,j} - K_{\beta,j}]$. Additionally, let $k := 0$, $\gamma := K_{\gamma,j}$, and let τ be the time at which firm j has activated the Collude-or-Compete algorithm.

(B) Run the collusive module.

Charge $p_j(\tau + 1) = p_{\text{start},j}$. If $\texttt{SYNC}_j(\tau + 1) = 1$, let $\tau := \tau + 1$ and go to part B. Charge $p_j(\tau + 2) = \varkappa_1 p_{\text{start},j}$. If $\texttt{SYNC}_j(\tau + 2) = 1$, let $\tau := \tau + 2$ and go to part B. If $p_{\neg j}(\tau + 2)/p_{\neg j}(\tau + 1) \neq \varkappa_1$, let $\tau := \tau + 2$ and go to part C. Charge $p_{j,\varepsilon}(\tau + 3) = p_j^{(d_j(\tau+1),0),(d_j(\tau+2),0)}(\tau + 3)$. If $p_{\neg j,\varepsilon}(\tau + 3) = p_{\neg j}^{(0,y_1),(0,y_2)}(\tau + 3)$ for some $(y_1, y_2) \in \{(0,0), (1,1), (0,1), (1,0)\}$ with $d_j(\tau + 1) + y_1 \leq 1$ and $d_j(\tau + 2) + y_2 \leq 1$, charge $p_j(\tau + 3) = p_j^{(d_j(\tau+1),y_1),(d_j(\tau+2),y_2)}(\tau + 3)$, store the demand data $d_{\neg j}(\tau + 1) = y_1$, $d_{\neg j}(\tau + 2) = y_2$ for future computations, let $t := \tau + 3$ and go to part B1. Otherwise, charge $p_j(\tau + 3) = p_{j,\varepsilon}(\tau + 3)$, let $\tau := \tau + 3$ and go to part C.

(B1) Main loop of the collusive module.

Charge $p_{j,\varepsilon}(t + 1) = p_j^{(d_j(t),0)}(t + 1)$. If $p_{\neg j,\varepsilon}(t + 1) = p_{\neg j}^{(0,y)}(t + 1)$ for some $y \in \{0, 1\}$ with $d_j(t) + y \leq 1$, charge $p_j(t + 1) = p_j^{(d_j(t),y)}(t + 1)$, store the demand data $d_{\neg j}(t) = y$ for future computations, let $t := t + 1$ and repeat part B1. Otherwise, charge $p_j(t + 1) = p_{j,\varepsilon}(t + 1)$, let $\tau := t + 1$ and go to part C.

(C) Run the competitive module.

Let $t := \tau$ and $n := k + 1$.

(C1) Main loop of the competitive module.

If $\texttt{SYNC}_j(t) = 1$, let $\tau := t$ and go to part B. Charge $p_j(t + 1) = p_j^{(n)} + K_{\beta,j}/n^{1/4}$. If $\texttt{SYNC}_j(t + 1) = 1$, let $\tau := t + 1$ and go to part B. Charge $p_j(t + 2) = p_j^{(n)} - K_{\beta,j}/n^{1/4}$. Let $t := t + 2$. If $n - k = \gamma$, let $k := k + \gamma$, $\gamma := \gamma + K_{\gamma,j}$, $\tau := t$ and go to part B. Otherwise, let $n := n + 1$ and repeat part C1.

Part A initializes the hyper-parameters of the algorithm. It also introduces three state variables: τ keeps track of when the algorithm switches between modules, k stores how many Kiefer–Wolfowitz cycles have been executed so far, and γ determines how many cycles the competitive module needs to run the next time it is activated.

Part B consists of the collusive module. During the initial two periods it charges subsequent prices with a fixed ratio of $\varkappa_1 \neq 1$ to enable maximum-likelihood estimation. At the start of the third period it sets the price that corresponds to the scenario where the opponent made no sales during the previous two periods. If the opponent's price reveals that a different scenario occurred, the "wrong" price is adjusted accordingly after ε time units, and the inferred demand data are stored for future computations. Part B1 consists of the collusive main loop: assume that the opponent made no sale during the previous period, set the corresponding price, adjust the price after ε time units if the opponent's price reveals that the no-sale assumption was wrong, store the inferred demand data, and repeat. During these steps the state variable t is used to keep track of time. If the synchronization mechanism dictates at any point that the collusive module has to be initialized, the algorithm immediately re-activates part B. Similarly, if the opponent charges a price that cannot be the output of the collusive module, the algorithm switches to part C.

Part C consists of the competitive module. It first introduces the state variable t, which keeps track of time, and n, which determines the competitive cycle that needs to be executed next. Part C1 consists of the competitive main loop: perform the nth Kiefer–Wolfowitz cycle, and update the state variables (i.e., increase t by 2 and n by 1). Again, if the synchronization mechanism requires it, the loop is interrupted and part B is activated. If the required number of cycles (i.e., γ cycles) have been completed without interruption, the state variables are updated accordingly (i.e., k is increased by γ and γ is increased by $K_{\gamma,j}$), and part B is activated.

4 | POLICY PERFORMANCE

In this section, we analyze the performance of the Collude-or-Compete algorithm against an opponent that uses the same algorithm (Subsection 4.1), and against an opponent whose prices are determined by an alternative policy within a class of "reasonable" competitive policies (Subsection 4.2). In both cases, we define regret and prove optimal regret bounds. The main results of this section, Theorems 6 and 7, can be summarized as follows. If the Collude-or-Compete algorithm is (eventually) used by both firms, prices converge to the collusive solution $\psi(\theta^*)$ with probability one, and the expected cumulative regret is of order $t^{3/4}$ (up to logarithmic terms). On the other hand, if one firm uses the Collude-or-Compete algorithm while the opponent prices according to a noisy reaction function, the prices generated by CC converge to a best-response price under some mild assumptions, and the expected cumulative regret is of order $\sqrt{t}$. Subsection 4.3 provides numerical illustrations of these results.

4.1 | Policy performance against a collusive opponent

Throughout Subsection 4.1, we consider the case that both firms use the Collude-or-Compete algorithm, possibly activated at different starting times. In particular, firm $j \in \{1,2\}$ uses the policy $\mathrm{CC}_j(p_{\mathrm{start},j}; p_j^{(1)}; K_{\alpha,j}; K_{\beta,j}; K_{\gamma,j})$ from some point onward—where the hyper-parameters can be different for each firm. Due to the mechanism described in Subsection 3.2.4 there exists a time of synchronization τ at which the firms simultaneously initiate the collusive module, after which both firms price according to the policy described in Subsection 3.2.1.

Recall that the collusive policy is designed to learn the collusive price pair $\psi(\theta^*)$, under mutual use of the policy, by estimating the unknown underlying model parameter θ^* based on accumulating price and demand data. Had θ^* been known, the firms could insist on charging the price pair $\psi(\theta^*)$ from time τ onward, in which case the expected revenue for firm j per period would be $r_j(\psi(\theta^*); \theta^*)$. We therefore measure the loss or *regret* for firm j at time t compared to the revenue under the "true" collusive solution by

$$R_j^{\mathrm{col}}(t) := \sum_{s=\tau+1}^{t} \left(r_j(\psi(\theta^*); \theta^*) - d_j(s) p_j(s) \right). \tag{28}$$

We show in this section that the firms, under mutual use of the Collude-or-Compete algorithm, learn the collusive price pair $\psi(\theta^*)$ and achieve sublinear expected regret.

Recall that $\hat{\theta}_t$ denotes the maximum-likelihood estimator of θ^* based on the price and demand data from periods $\tau+1, \dots, t$. Our first result in this section entails that $\hat{\theta}_t$ is a consistent estimator, and provides a high-probability bound on the estimation error.

Theorem 5. *Let $t \geq \tau + 2$. There exist positive constants K_1, K_2, and $y_{\max}$ (which do not depend on t, τ, or θ^*) such that, for all $y \in (0, y_{\max}]$,*

$$\mathbb{P}(\|\hat{\theta}_t - \theta^*\| < y) \geq 1 - K_1(t-\tau) \cdot \exp(-K_2 y^2 (t-\tau)^{1+\kappa_0}). \tag{29}$$

To prove this theorem, we apply Brouwer's invariance of domain theorem to derive conditions which imply that $\hat{\theta}_t$ lies in a small neighborhood of θ^*. Using matrix inequalities and Hoeffding's concentration bounds we show that the lower bound on the sample variance induced by the collusive policy implies that these conditions hold with high probability.

Our main result in this section is the following regret bound[4].

Theorem 6. *It holds that $\mathbb{E}[R_j^{\mathrm{col}}(t)] = \tilde{O}((t-\tau)^{\kappa_0/2+1} + (t-\tau)^{1/2-\kappa_0/2})$ for each $j \in \{1,2\}$. In particular, if $\kappa_0 = -1/2$, then $\mathbb{E}[R_j^{\mathrm{col}}(t)] = \tilde{O}((t-\tau)^{3/4})$.*

To prove Theorem 6, we first show for all $s \geq \tau + 3$ that the instantaneous absolute revenue loss during the sth period, compared to the revenue under the true collusive price pair $\psi(\theta^*)$, is bounded from above by a constant times the distance between $\psi(\theta^*)$ and the charged price pair. This latter term can be decomposed into two terms: the norm of the price pair perturbation induced by the collusive policy, and the distance between the true and estimated collusive price pair, which is bounded from above by a constant times the estimation error $||\theta^* - \hat{\theta}_{s-1}||$. By construction of the collusive policy, the cumulative expected price perturbations are of the order $O((t-\tau)^{\kappa_0/2+1})$, and, by application of Theorem 5, the cumulative expected estimation errors are of the order $\tilde{O}((t-\tau)^{1/2-\kappa_0/2})$. The total expected regret is therefore $\tilde{O}((t-\tau)^{\kappa_0/2+1} + (t-\tau)^{1/2-\kappa_0/2})$.

The parameter κ_0 captures the trade-off between exploration and exploitation. On the one hand, increasing κ_0 leads to more price dispersion and hence lower estimation errors, captured by the term $\tilde{O}((t-\tau)^{1/2-\kappa_0/2})$. On the other hand, price experiments are costly, which is captured by the term $O((t-\tau)^{\kappa_0/2+1})$. The optimal balance between the two competing objectives of earning and learning is struck by choosing $\kappa_0 = -1/2$.

Remark 2. Kleinberg et al. (2008) show that the regret rate of $\tilde{O}(t^{(d+1)/(d+2)})$ is asymptotically optimal for Lipschitz-continuous bandits with d-dimensional action space, where $d \geq 1$. The rate $t^{3/4}$ for $d = 2$ (because there are two products our action space is two-dimensional) corresponds with the regret bound derived in Theorem 6.

4.2 | Policy performance against a competitive opponent

Throughout Subsection 4.2, we consider the case that firm $j \in \{1,2\}$ uses the Collude-or-Compete algorithm, while the opponent's prices are given by a noisy reaction function. In particular, firm j uses the policy $\mathrm{CC}_j(p_{\mathrm{start},j}, p_j^{(1)}, K_{\alpha,j}, K_{\beta,j}, K_{\gamma,j})$ from the start of the time horizon. For the opponent we consider the general framework that, for all $s \in \mathbb{N}$,

$$p_{\neg j}(s) = \phi(p_j(s)) + \varepsilon_s, \tag{30}$$

where $\phi : [0,\infty) \to [0,\infty)$ is a function unknown to firm j, and where ε_s are $\mathcal{F}_{s-1,\neg j}$-measurable disturbance terms. In words, firm $\neg j$ "reacts" on firm j, possibly with disturbance terms.

In Theorem 7, we prove an $O(\sqrt{t})$ regret bound. To appropriately define regret we need to make some mild assumptions on ϕ and ε_s. For all *reaction functions* $\varphi : [0,\infty) \to [0,\infty)$, which determine a price based on the price charged by the opponent, and all $p \geq 0$, let

$$f_\varphi(p) := r_j([p, \varphi(p)]_j; \theta^*) \tag{31}$$

denote the expected revenue for firm j given that she charges the price p and the opponent charges $\varphi(p)$. Let Φ_{K_0} be the class of all twice continuously differentiable reaction functions φ such that

(F1) f_φ is strictly log-concave at its stationary points,

which implies that f_φ has a unique maximizer p^*_φ (Lemma 3), and

(F2) $|f'_\varphi(p)| \geq K_0|p - p^*_\varphi|$ for all $p \in [0, p_{\max,j}]$,

where K_0 is a given positive constant. We assume that ϕ is an element of Φ_{K_0} and that $p^*_\phi < p_{\max,j}$.

Lemma 3. *If φ is a twice continuously differentiable reaction function that satisfies Condition (F1), then there exists a unique maximizer p^*_φ of f_φ. If, additionally, $p^*_\varphi < p_{\max,j}$, then Condition (F2) is satisfied for all sufficiently small $K_0 > 0$.*

We now show for two reasonable reaction functions that they are elements of our reaction-function class. Define for all $q \geq 0$ the constant (fixed-price) function $\mathrm{FP}_q(p)$ that equals q for any $p \geq 0$.

Lemma 4. *Let $\theta \in \Theta$ and $q \geq 0$. The functions $p \mapsto \mathrm{BR}_{\neg j}(p; \theta)$ and FP_q are elements of Φ_{K_0} for all sufficiently small $K_0 > 0$.*

Furthermore, we assume that $\mathbb{E}[|\varepsilon_s|]$ converges to zero, which means that the opponent gradually learns to price according to the reaction function ϕ, rendering p^*_ϕ the asymptotically optimal price for firm j, and $f_\phi(p^*_\phi)$ the asymptotically optimal expected revenue for firm j per period. We therefore measure the regret for firm j compared to the asymptotically optimal revenue by

$$R_j^{\mathrm{comp}}(t) := \sum_{s=1}^{t} \left(f_\phi(p^*_\phi) - d_j(s)p_j(s) \right). \tag{32}$$

The main result of this section is that the Collude-or-Compete algorithm learns the asymptotically optimal price p^*_ϕ and achieves $O(\sqrt{t})$ expected regret under appropriate conditions on ε_s. To formulate our main result, define, for all $s \in \mathbb{N}$,

$$\xi_s := r_j([p_j(s), \phi(p_j(s)) + \varepsilon_s]_j; \theta^*) - r_j([p_j(s), \phi(p_j(s))]_j; \theta^*). \tag{33}$$

Note that ξ_s is a random variable that captures how the opponent's noise ε_s on the reaction function distorts the expected revenue for firm j. We finally make the mild and reasonable assumption that our Collude-or-Compete algorithm can detect in finitely many time periods that the prices of firm $\neg j$ are not determined by Collude-or-Compete. Under this assumption, we have the following result:

Theorem 7. *Let $K_{\alpha,j} > (\sqrt{2} - 1)/(2K_0)$. If there exists a positive constant K_ξ such that $\mathbb{E}[|\xi_s|] \leq K_\xi/\sqrt{s}$ for all $s \in \mathbb{N}$, and $\mathbb{E}[|\xi_s|^2]$ is bounded uniformly in s, then $\mathbb{E}[R_j^{\mathrm{comp}}(t)] = O(\sqrt{t})$.*

To prove Theorem 7, we first show that the expected absolute revenue loss per period, compared to the asymptotically optimal revenue $f_\phi(p^*_\phi)$, is bounded from above by the sum of three separate terms involving the mean square error of the pivot price, the square of the price perturbation induced by our competitive policy, and the expected absolute noise on the revenue function, yielding the following regret bound:

$$\left|\mathbb{E}\left[R_j^{\mathrm{comp}}(t)\right]\right| \leq \sum_{s=1}^{t} \left(2K_1 \mathbb{E}\left[\left(p^*_\phi - p_j^{(\lceil s/2 \rceil)}\right)^2\right] + 2K_1 \beta^2_{\lceil s/2 \rceil, j} + \mathbb{E}\left[|\xi_s|\right] \right), \tag{34}$$

where K_1 is a positive constant. By adapting the proof of Theorem 1 in Broadie et al. (2011) to the case where (ξ_s) is not necessarily a martingale difference sequence, we show that there exists a positive constant C such that, for all $s \in \mathbb{N}$,

$$\mathbb{E}\left[\left(p^*_\phi - p_j^{(\lceil s/2 \rceil)}\right)^2\right] \leq C/\sqrt{s}. \tag{35}$$

By substituting (19), (35), and the assumed upper bound of $\mathbb{E}[|\xi_s|]$ into the regret bound (34), we arrive at the final conclusion $\mathbb{E}[R_j^{\mathrm{comp}}(t)] = O(\sqrt{t})$.

Broadie et al. (2011, Proposition 1) show that $O(1/\sqrt{s})$ is the minimal order of the upper bound in (35) over a wide range of sequences $(\alpha_{n,j})$ and $(\beta_{n,j})$, rendering $O(\sqrt{t})$ the minimal order of the upper bound in (34). This rate is asymptotically optimal (cf. Keskin & Zeevi, 2014 or Broder & Rusmevichientong, 2012).

4.3 | Numerical illustrations

4.3.1 | Collusive opponent

We now provide a numerical illustration of the theoretical results from Subsection 4.1. In particular, we show by means of simulation that the prices determined by the Collude-or-Compete algorithm converge to the collusive solution $\psi(\theta^*)$ when used by both firms, and that the growth rate of the regret is in accordance with Theorem 6.

To illustrate convergence to the collusive price pair $\psi(\theta^*)$, we run a simulation consisting of 50,000 time periods. We set $\varkappa_0 = -1/2$, $\varkappa_1 = 5/11$, $\theta^* = (2, 0.010, 3, 0.004)$, $\psi = \psi^{\mathrm{RPO}}$ (defined in Subsection 2.2.2) and set the hypercube $\Xi = [-1, 5] \times [0.001, 0.019] \times [-1, 5] \times [0.001, 0.019]$ as the compact and convex parameter space, in line with the

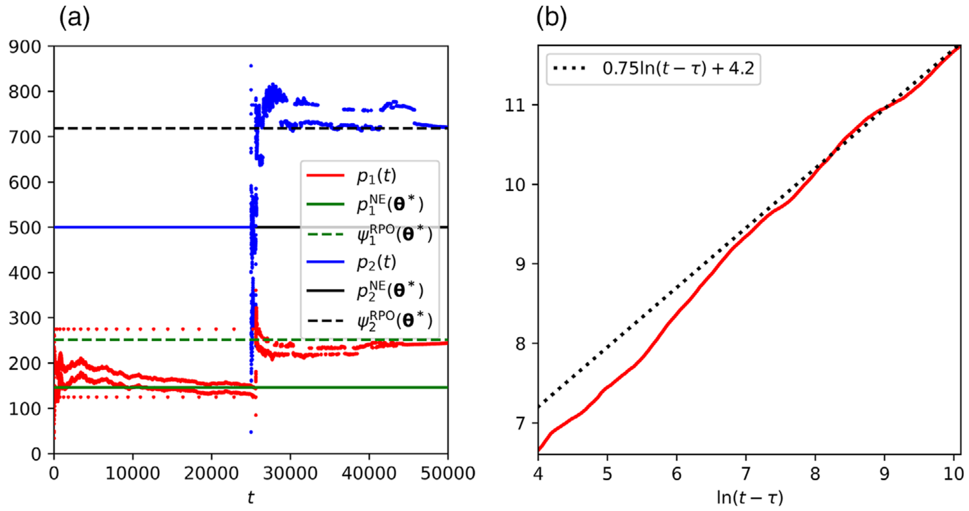

FIGURE 1 Convergence of prices (left panel) and regret on a log–log scale (right panel) when both firms use the Collude-or-Compete algorithm [Color figure can be viewed at wileyonlinelibrary.com]

literature mentioned in Subsection 2.2.1. For this choice of θ^*, the joint-revenue-maximizing prices cause revenue losses of almost 80% compared to the Nash equilibrium for one of the firms, whereas the RPO prices increase revenue by almost 20% for both firms. The RPO prices are also significantly higher than the Nash equilibrium prices: $\mathbf{p}^{\mathrm{NE}}(\theta^*) \approx (146, 500)$ and $\psi^{\mathrm{RPO}}(\theta^*) \approx (252, 718)$.

Regarding the price policies applied by the firms, suppose that firm one applies the policy $\mathrm{CC}_1(p_{\mathrm{start},1}; p_1^{(1)}; K_{\alpha,1}; K_{\beta,1}; K_{\gamma,1})$ throughout the entire time horizon (i.e., from time 0 onward), with $p_{\mathrm{start},1} = 275$, $p_1^{(1)} = 150$, $K_{\alpha,1} = 100$, $K_{\beta,1} = 100$, and $K_{\gamma,1} = 60$. Firm two fixes its price to 500 for the initial 25,000 time periods, and then starts using the Collude-or-Compete algorithm; this setup illustrates that the Collude-or-Compete algorithm is self-synchronizing, that is, prices converge to $\psi(\theta^*)$ even if the firms activate the Collude-or-Compete algorithm at different starting times. In particular, firm two activates the policy $\mathrm{CC}_2(p_{\mathrm{start},2}; p_2^{(1)}; K_{\alpha,2}; K_{\beta,2}; K_{\gamma,2})$ at time 25,000, with $p_{\mathrm{start},2} = 770$, $p_2^{(1)} = 450$, $K_{\alpha,2} = 300$, $K_{\beta,2} = 300$, and $K_{\gamma,2} = 50$.

Figure 1a shows the price paths resulting from our simulation. The isolated (red) dots are the result of firm one switching to the collusive module (charging $p_{\mathrm{start},1} = 275$ and $\varkappa_1 p_{\mathrm{start},1} = 125$, in that order), which occurs with decreasing frequency as long as firm two keeps her price fixed to 500. Firm one predominantly uses the competitive module during this phase, and Figure 1a shows that her prices initially converge to the best-response price $\mathrm{BR}_1(500; \theta^*) \approx 146$ (conform Theorem 7). Synchronization is achieved at time $\tau = 25606$, after which the prices charged by both firms converge to the collusive solution $\psi^{\mathrm{RPO}}(\theta^*)$. The taboo intervals of the collusive policy (Subsection 3.2.1) can be discerned in the figure as white regions that the price paths occasionally "jump" over.

We repeat this simulation 1000 times in order to illustrate the growth rate of the regret. Figure 1b shows that the logarithm of the sample average of $\tilde{R}_1^{\mathrm{col}}(t) := \sum_{s=\tau+1}^{t} |r_1(\psi(\theta^*); \theta^*) - r_1(\mathbf{p}(s); \theta^*)|$ asymptotically resembles a line with gradient 3/4 when plotted against $\ln(t-\tau)$, where $\tau = 25606$ denotes the time of synchronization. This illustrates that $\mathbb{E}[R_1^{\mathrm{col}}(t)] = O(\mathbb{E}[\tilde{R}_1^{\mathrm{col}}(t)]) = \tilde{O}((t-\tau)^{3/4})$, conform Theorem 6.

4.3.2 | Competitive opponent

We now provide a numerical illustration of the theoretical results from Subsection 4.2. In particular, we show by means of simulation that the prices determined by the Collude-or-Compete algorithm converge to the asymptotically optimal price when playing against an explore-then-commit policy, and that the growth rate of the regret is in accordance with Theorem 7.

To illustrate convergence to the asymptotically optimal price we run a simulation consisting of 1 million time periods. We again set $\varkappa_0 = -1/2$, $\varkappa_1 = 5/11$, $\theta^* = (2, 0.010, 3, 0.004)$, $\psi = \psi^{\mathrm{RPO}}$, and $\Xi = [-1, 5] \times [0.001, 0.019] \times [-1, 5] \times [0.001, 0.019]$. Regarding the price policies applied by the firms, we suppose that firm one applies the policy $\mathrm{CC}_1(p_{\mathrm{start},1}; p_1^{(1)}; K_{\alpha,1}; K_{\beta,1}; K_{\gamma,1})$ throughout the entire time horizon, with $p_{\mathrm{start},1} = 275$, $p_1^{(1)} = 150$, $K_{\alpha,1} = 100$, $K_{\beta,1} = 100$, and $K_{\gamma,1} = 1000$. Firm two uses an explore-then-commit policy: for each of the initial half million time periods, one of the five experimental prices 300, 400, 500, 600, and 700 is chosen uniformly at random and charged during that period. After these half million periods, firm two selects an experimental price p_2^* that provided the highest mean revenue per period, and commits to that price for the rest of the time horizon. This policy fits the

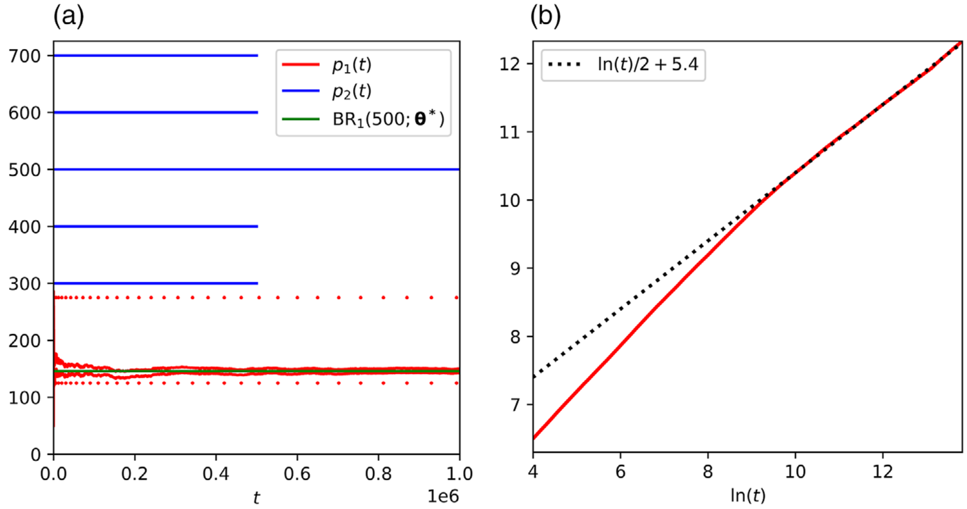

FIGURE 2 Convergence of prices (left panel) and regret (right panel) on a log–log scale when one firm uses the Collude-or-Compete algorithm and the other firm uses an explore-then-commit policy [Color figure can be viewed at wileyonlinelibrary.com]

competitive framework from Subsection 4.2 with the fixed-price reaction function $\phi = \mathrm{FP}_{p_2^*}$.

Figure 2a shows the price paths resulting from our simulation. It illustrates that firm two initially explores five different prices and eventually commits to the price $p_2^* = 500$. The prices charged by firm one converge to the asymptotically optimal price $\mathrm{BR}_1(500; \theta^*)$. The isolated (red) dots indicate the moments at which firm one activates the collusive module.

We repeat this simulation 1000 times in order to illustrate the growth rate of the regret. It turns out that $p_2^* = 500$ in all thousand simulations. It therefore holds in the context of these simulations that $\mathbb{E}[R_1^{\mathrm{comp}}(t)] = \mathbb{E}[\tilde{R}_1^{\mathrm{comp}}(t)]$, where $\tilde{R}_1^{\mathrm{comp}}(t) := \sum_{s=1}^{t}(r_1((\mathrm{BR}_1(500; \theta^*), 500); \theta^*) - r_1(\mathbf{p}(s); \theta^*))$. Figure 2b illustrates that the logarithm of the resulting sample mean of $\tilde{R}_1^{\mathrm{comp}}(t)$ asymptotically resembles a line with gradient 1/2 when plotted against $\ln(t)$, indicating that $\mathbb{E}[\tilde{R}_1^{\mathrm{comp}}(t)] = O(\sqrt{t})$, and hence that expected regret is $O(\sqrt{t})$, conform Theorem 7.

5 | DISCUSSION

Contributions. This work makes several contributions to the field of dynamic pricing and demand learning. First, in a pricing duopoly with multinomial logit purchase probabilities, we show for several axiomatic notions of collusion that they reduce consumer welfare and component-wise increase sellers' revenues and prices compared to the Nash equilibrium. These properties hold regardless of the model parameters, so that these constitute stable notions of collusion in all markets modeled in this way. This result demonstrates that even in markets where the demand functions are very asymmetric, firms can define cartel prices in a way that is mutually beneficial and detrimental to consumer welfare. Second, we construct a price algorithm that learns the cartel prices from data when used by both players in the duopoly, and learns to respond optimally against a class of competitive policies. In both cases, we analyze the performance by proving theoretical regret bounds. To learn to collude, the firms do not need to know that the competitor is using the same algorithm, and the algorithms do not have to be initiated at exactly the same moment. The algorithm enables the firms to learn to price collusively without engaging in illicit forms of communication, and is thus an example of *data-driven* algorithmic collusion. The algorithm includes a mechanism to infer sales data from price paths so that our algorithm works well even if sellers do not share their sales data with competitors. This has the important consequence that legal measures that forbid firms to share their sales data, directly or indirectly, will not fully mitigate the threat of algorithmic collusion. It also shows that defining "illicit communication" in the context of price cartels is a complex question: our work shows that price paths generated by algorithms inevitably contain information that can be "decoded" by firms that use the same algorithm to facilitate collusion. Our results on the performance of the algorithm when playing against a competitive opponent are of independent interest as such results are scarce in the literature.

The role of demand information. Our results also apply when demand information is shared publicly; Meylahn and den Boer (2022) provide several real-life examples, including web-shops that explicitly disclose inventory information ("*x* units in stock," "*x* units sold in the last *y* hours"). Aouad and den Boer (2021) demonstrate in an assortment optimization problem that observing the competitor's demand realizations may not even be necessary in order to learn the underlying model. However, the decisions of both players would no longer be based on the same data set, which makes it more challenging to discern whether the opponent is using the same algorithm.

Extensions. Our model can be extended to multiple products, time-varying parameters, and limits on the maximum number of price changes. Ideas from den Boer (2014) (multi-product policies), Chen et al. (2020) and Cheung et al. (2017) (limited price changes), and Keskin and Zeevi (2017) (nonstationary environments) can be adapted to construct well-performing price policies, both in a competitive and in a collusive setting. A second important extension that is left for future research is to allow for more than two players, each of which may either price competitively or be willing to form a cartel. This extension requires generalizing our notion of collusion to accommodate larger cartels, extending our results on synchronization and statistical aspects, and studying the stability properties of (sub)coalition structures in an oligopoly.

ACKNOWLEDGMENTS

The authors are grateful to the reviewers and editors for constructive suggestions. This research was supported by NWO Klein, Grant OCENW.KLEIN.016.

ORCID

Thomas Loots https://orcid.org/0000-0002-7753-2705
Arnoud V. den Boer https://orcid.org/0000-0003-4779-0436

ENDNOTES

[1] The abbreviation DCVP stands for "Duopolistic Controlled Variance Pricing," based on the policy name "Controlled Variance Pricing" from den Boer and Zwart (2014) for a monopolist.

[2] Note that firm j requires the correct value of $c_{\neg j}$ in order to compute the prices that the collusive policy would set for the competitor. Because $c_{\neg j} = 2^{-2-\kappa_0}(p_{\neg j}(\tau+1))^2(\kappa_1 - 1)^2$ if firm $\neg j$ activates the collusive module at time τ, this value can be computed by firm j immediately after period $\tau + 1$.

[3] Our description of the synchronization mechanism uses the property that the collusive module of firm $\neg j$ is active for only two periods at a time. However, it is worth emphasizing that this property is not necessary for the mechanism to work; it only makes for a simpler description. That is, if firm $\neg j$ runs the collusive module for more than two periods, firm j can still check whether the opponent's prices conform to the collusive module by applying the demand-information extraction-mechanism from Subsection 3.2.2. Therefore, firm j can still distinguish two subsequent phases, infer $K_{\gamma,\neg j}$, and derive when the opponent will activate the collusive module.

[4] Here we use the notation $f(t) = \tilde{O}(g(t))$ as shorthand for $f(t) = O(g(t)\ln(g(t))^k)$ for some arbitrary $k > 0$.

REFERENCES

Akçay, Y., Natarajan, H. P., & Xu, S. H. (2010). Joint dynamic pricing of multiple perishable products under consumer choice. *Management Science*, *56*(8), 1345–1361.

Aouad, A., & den Boer, A. V. (2021). Algorithmic collusion in assortment games. Working paper. https://ssrn.com/abstract=3930364

Bertsimas, D., & Perakis, G. (2006). Dynamic pricing: A learning approach. In S. Lawphongpanich, D. W. Hearn, & M. J. Smith (Eds.), *Mathematical and computational models for congestion charging* (pp. 45–79). US: Springer.

Besbes, O., & Zeevi, A. (2009). Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Operations Research*, *57*(6), 1407–1420.

Besbes, O., & Zeevi, A. (2015). On the (surprising) sufficiency of linear models for dynamic pricing with demand learning. *Management Science*, *61*(4), 723–739.

Bos, I., & Harrington, J. E., Jr. (2010). Endogenous cartel formation with heterogeneous firms. *The RAND Journal of Economics*, *41*(1), 92–117.

Broadie, M., Cicek, D., & Zeevi, A. (2011). General bounds and finite-time improvement for the Kiefer-Wolfowitz stochastic approximation algorithm. *Operations Research*, *59*(5), 1211–1224.

Broder, J., & Rusmevichientong, P. (2012). Dynamic pricing under a general parametric choice model. *Operations Research*, *60*(4), 965–980.

Calvano, E., Calzolari, G., Denicolo, V., & Pastorello, S. (2020). Artificial intelligence, algorithmic pricing, and collusion. *American Economic Review*, *110*(10), 3267–3297.

Chen, B., Chao, X., & Wang, Y. (2020). Data-based dynamic pricing and inventory control with censored demand and limited price changes. *Operations Research*, *68*(5), 1445–1456.

Chen, Q., Jasin, S., & Duenyas, I. (2019). Nonparametric self-adjusting control for joint learning and optimization of multiproduct pricing with finite resource capacity. *Mathematics of Operations Research*, *44*(2), 601–631.

Chen, Q., Jasin, S., & Duenyas, I. (2021). Joint learning and optimization of multi-product pricing with finite resource capacity and unknown demand parameters. *Operations Research*, *69*(2), 560–573.

Cheung, W. C., Simchi-Levi, D., & Wang, H. (2017). Dynamic pricing and demand learning with limited price experimentation. *Operations Research*, *65*(6), 1722–1731.

Cooper, W. L., Homem-de-Mello, T., & Kleywegt, A. J. (2015). Learning and pricing with models that do not explicitly incorporate competition. *Operations Research*, *63*(1), 86–103.

Delahaye, T., Acuna-Agost, R., Bondoux, N., Nguyen, A.-Q., & Boudia, M. (2017). Data-driven models for itinerary preferences of air travelers and application for dynamic pricing optimization. *Journal of Revenue and Pricing Management*, *16*(6), 621–639.

den Boer, A. V. (2014). Dynamic pricing with multiple products and partially specified demand distribution. *Mathematics of Operations Research*, *39*(3), 863–888.

den Boer, A. V. (2015). Dynamic pricing and learning: Historical origins, current research, and new directions. *Surveys in Operations Research and Management Science*, *20*(1), 1–18.

den Boer, A. V., Meylahn, J. M., & Schinkel, M. P. (2022). Artificial collusion: Examining supracompetitive pricing by Q-learning algorithms. Working paper No. 2022–06, Amsterdam Center for Law & Economics, University of Amsterdam, The Netherlands. https://ssrn.com/abstract=4213600

den Boer, A. V., & Zwart, B. (2014). Simultaneously learning and optimizing using controlled variance pricing. *Management Science*, *60*(3), 770–783.

den Boer, A. V., & Zwart, B. (2015). Dynamic pricing and learning with finite inventories. *Operations Research*, *63*(4), 965–978.

Ezrachi, A., & Stucke, M. E. (2016). *Virtual competition: The promise and perils of the algorithm-driven economy*. Harvard University Press.

Ezrachi, A., & Stucke, M. E. (2017a). Algorithmic collusion: Problems and counter-measures. https://one.oecd.org/document/DAF/COMP/WD(2017)25/en/pdf

Ezrachi, A., & Stucke, M. E. (2017b). Artificial intelligence & collusion: When computers inhibit competition. *University of Illinois Law Review*, *2017*(5), 1775–1810.

Ezrachi, A., & Stucke, M. E. (2017c). Two artificial neural networks meet in an online hub and change the future (of competition, market dynamics and society). https://ssrn.com/abstract=2949434

Ezrachi, A., & Stucke, M. E. (2020). Sustainable and unchallenged algorithmic tacit collusion. *Northwestern Journal of Technology and Intellectual Property*, *17*(2), 217–260.

Fischer, C., & Normann, H.-T. (2019). Collusion and bargaining in asymmetric Cournot duopoly: An experiment. *European Economic Review*, *111*, 360–379.

Gal, M. S. (2019). Algorithms as illegal agreements. *Berkeley Technology Law Journal*, *34*(1), 67–118.

Gallego, G., & Topaloglu, H. (2019). *Revenue management and pricing analytics*. Springer.

Hansen, K. T., Misra, K., & Pai, M. M. (2021). Frontiers: Algorithmic collusion: Supra-competitive prices via independent algorithms. *Marketing Science*, *40*(1), 1–12.

Harrington, J. E. (2018). Developing competition law for collusion by autonomous artificial agents. *Journal of Competition Law and Economics*, *14*(3), 331–363.

Harrison, J. M., Keskin, N. B., & Zeevi, A. (2012). Bayesian dynamic pricing policies: Learning and earning under a binary prior distribution. *Management Science*, *58*(3), 570–586.

Hoffman, B. (2018). Competition and consumer protection implications of algorithms, artificial intelligence, and predictive analytics. https://www.ftc.gov/system/files/documents/public_statements/1431041/hoffman_-_ai_intro_speech_11-14-18.pdf

Keskin, N. B., & Zeevi, A. (2014). Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies. *Operations Research*, *62*(5), 1142–1167.

Keskin, N. B., & Zeevi, A. (2017). Chasing demand: Learning and earning in a changing environment. *Mathematics of Operations Research*, *42*(2), 277–307.

Kiefer, J., & Wolfowitz, J. (1952). Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, *53*(3), 462–466.

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

Kleinberg, R., Slivkins, A., & Upfal, E. (2008). Multi-armed bandits in metric spaces. *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing* (pp. 681–690). New York: Association for Computing Machinery.

Kühn, K., & Tadelis, S. (2017). Algorithmic collusion. Presentation, 12th International Conference on Competition and Regulation, CRESSE, Center for Economic Research and Competition Strategy, Athens, Greece. https://www.cresse.info/wp-content/uploads/2020/02/2017_sps5_pr2_Algorithmic-Collusion.pdf

Li, H., & Huh, W. T. (2011). Pricing multiple products with the multinomial logit and nested logit models: Concavity and implications. *Manufacturing and Service Operations Management*, *13*(4), 549–563.

Li, H., Webster, S., Mason, N., & Kempf, K. (2019). Product-line pricing under discrete mixed multinomial logit demand. *Manufacturing and Service Operations Management*, *21*(1), 14–28.

Maglaras, C., & Meissner, J. (2006). Dynamic pricing strategies for multiproduct revenue management problems. *Manufacturing and Service Operations Management*, *8*(2), 136–148.

Mandrescu, D. (2018). When algorithmic pricing meets concerted practices—The case of Partneo. https://www.lexxion.eu/coreblogpost/when-algorithmic-pricing-meets-concerted-practices-the-case-of-partneo/

Meylahn, J. M., & den Boer, A. V. (2022). Learning to collude in a pricing duopoly. *Manufacturing & Service Operations Management*, *24*(5), 2577–2594.

OECD. (2017a). Algorithms and collusion—Summaries of contributions. https://www.oecd.org/daf/competition/algorithms-and-collusion.htm

OECD. (2017b). Algorithms and collusion: Competition policy in the digital age. http://www.oecd.org/competition/algorithms-collusion-competition-policy-in-the-digital-age.htm

OECD. (2019). Hub-and-spoke arrangements—Background note by the secretariat. http://www.oecd.org/daf/competition/hub-and-spoke-arrangements.htm

Rayfield, W. Z., Rusmevichientong, P., & Topaloglu, H. (2015). Approximation methods for pricing problems under the nested logit model with price bounds. *INFORMS Journal on Computing*, *27*(2), 335–357.

Roth, A. E. (1979). *Axiomatic models of bargaining*. Springer.

Schrepel, T. (2017). Here's why algorithms are NOT (really) a thing. https://leconcurrentialiste.com/algorithms-based-practices-antitrust/

Schwalbe, U. (2018). Algorithms, machine learning, and collusion. *Journal of Competition Law and Economics*, *14*(4), 568–607.

Spiridonova, A., & Juchnevicius, E. (2020). Price algorithms as a threat to competition under the conditions of digital economy: Approaches to antimonopoly legislation of BRICS countries. *BRICS Law Journal*, *7*(2), 94–117.

Talluri, K. T., & van Ryzin, G. (2004). *The theory and practice of revenue management*. Springer.

van de Geer, R., & den Boer, A. V. (2022). Price optimization under the finite-mixture logit model. *Management Science*, *68*(10), 7480–7496.

Veljanovski, C. (2020). Algorithmic antitrust. Working paper. https://ssrn.com/abstract=3644363

Wang, Y., Chen, B., & Simchi-Levi, D. (2021). Multimodal dynamic pricing. *Management Science*, *67*(10), 6136–6152.

Wang, Z., Deng, S., & Ye, Y. (2014). Close the gaps: A learning-while-doing algorithm for single-product revenue management problems. *Operations Research*, *62*(2), 318–331.

Yang, Y., Lee, Y.-C., & Chen, P.-A. (2020). Competitive demand learning: A data-driven pricing algorithm. Working paper. https://arxiv.org/abs/2008.05195

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Loots, T., & den Boer, A. V. (2023). Data-driven collusion and competition in a pricing duopoly with multinomial logit demand. *Production and Operations Management*, *32,* 1169–1186. https://doi.org/10.1111/poms.13919