NBER WORKING PAPER SERIES

ALGORITHMIC COERCION WITH FASTER PRICING

Zach Y. Brown
Alexander MacKay

Working Paper 34070
http://www.nber.org/papers/w34070

NATIONAL BUREAU OF ECONOMIC RESEARCH
1050 Massachusetts Avenue
Cambridge, MA 02138
July 2025

We thank Jim Dana, William Fuchs, Maxim Engers, Joe Harrington, Scott Kominers, Matt Leisten, Nathan Miller, and Andrew Sweeting for helpful comments. We thank participants at the Northwestern Antitrust Conference, Utah Winter Business Economics Conference, IIOC, Barcelona Summer Forum Digital Economics Workshop, and ZEW Conference on Economics of Information and Communication Technologies. We are grateful for the research assistance of Harry Kleyer. The views expressed herein are those of the authors and do not necessarily reflect the views of the National Bureau of Economic Research.

Algorithmic Coercion with Faster Pricing
Zach Y. Brown and Alexander MacKay
NBER Working Paper No. 34070
July 2025
JEL No. D43, L13, L40, L81, L86

ABSTRACT

We examine a model in which one firm uses a pricing algorithm that enables faster pricing and multi-period commitment. We characterize a coercive equilibrium in which the algorithmic firm maximizes its profits subject to the incentive compatibility constraint of its rival. By adopting an algorithm that enables faster pricing and (imperfect) commitment, a firm can unilaterally induce substantially higher equilibrium prices even when its rival maximizes short-run profits and cannot collude. The algorithmic firm can earn profits that exceed its share of collusive profits, and coercive equilibrium outcomes can be worse for consumers than collusive outcomes. In extensions, we incorporate simple learning by the rival, and we explore the implications for platform design.

Zach Y. Brown
University of Michigan
Department of Economics
and NBER
zachb@umich.edu

Alexander MacKay
University of Virginia
Department of Economics
mackay@virginia.edu

1 Introduction

In classic models of collusion, all firms are forward-looking and take into account their rivals' dynamic strategies. Short-sighted behavior by rivals is a source of fragility in these models.[1] For example, when all firms set prices simultaneously, a single firm facing rivals that maximize short-run profits cannot profitably choose a price above the competitive (Bertrand-Nash) level in equilibrium.

The increasing adoption of pricing algorithms across many markets—including online retail, gasoline, food delivery, among others—introduces behavior that departs from the simultaneous price-setting or quantity-setting assumption that is standard in the literature. Pricing algorithms change the nature by which firms update prices, allowing them to select rules that automatically react to price changes by rivals. Algorithm providers advertise that firms can "set it and forget it," implying commitment to a strategy over time.[2] Moreover, the use of algorithms has not been uniform across firms within a market. Recent empirical work has documented that some firms employ high-speed pricing algorithms that provide a significant advantage in terms of the ability to monitor and react to price changes by rivals,[3] and many large firms, including the online retailer Amazon, have invested substantial resources to obtain this advantage.[4]

Motivated by these observations, we develop a model where a firm commits (imperfectly) to an algorithm across periods. The algorithm is a function of the rival's price and can update prices more quickly than its rival within a period. In our baseline analysis, we assume that the rival simply maximizes profits in the current period. By making this assumption about the rival, we explore the potential for "robust" supracompetitive prices that occur even when collusive equilibria are ruled out. We define and characterize a *coercive* equilibrium where the algorithmic firm acts unilaterally to maximize its own profits, subject to the incentive compatibility constraint of its rival.

Our model characterizes pricing technology as the algorithm's relative reaction time to rival prices and the probability that the firm can update its algorithm each period. A faster reaction time provides a speed advantage that can be interpreted as either the ability to more quickly update prices or the ability to monitor the price changes of rivals at a higher frequency. When the firm updates its algorithm, it flexibly chooses an initial price and an update function that responds to the rival's price. Despite the fact that our model rules out standard collusive strate-

[1] Several factors are viewed as important for facilitating collusion, including similarity in size and costs, predictability of demand, observability of all rivals' prices, and the possibility of frequent direct communication. See, e.g., Scherer (1980), Tirole (1988), or Porter (2005).

[2] See https://www.informedrepricer.com/ (accessed February 2025).

[3] Brown and MacKay (2023) show that the pricing technology for large online retailers varies from once-per-week updates to updates that occur multiple times per hour. The adoption of high-speed pricing algorithms has also been observed in other settings (e.g., Assad et al., 2024; Aparicio et al., 2021).

[4] The U.S. Federal Trade Commission notes that "Amazon has estimated that for thousands of the most popular products on Amazon it can detect any price change virtually anywhere on the internet within hours," and asserts that Amazon automatically reacts to price changes by other online retailers and by sellers on its marketplace. See FTC v. Amazon (September, 2023).

gies, the algorithmic firm can incentivize higher prices by committing to an update function that punishes deviations from a target price by its rival. Under an assumption that the algorithmic firm sets the punishment function to be the one-shot best-response function, the equilibrium is unique.

The model nests the standard Bertrand and sequential-move equilibria as special cases. When there is no speed advantage and full commitment, the model yields the sequential equilibrium where the algorithmic firm is the leader. Conversely, when the algorithm can react instantly but there is no commitment across periods, the equilibrium is that where the algorithmic firm is the sequential follower.[5]

In the presence of both a speed advantage and multi-period commitment, the algorithmic firm can obtain prices and profits that are substantially higher than the benchmark cases above. The speed advantage allows the algorithmic firm to coerce its rival into setting higher prices; commitment across periods prevents the algorithmic firm from deviating from its own optimal long-run strategy. Intuitively, the algorithmic firm uses the threat to quickly undercut its rival before its rival can react to push its rival to set a higher price. Commitment to an algorithm across periods does not directly affect the incentives of the rival, but rather allows the algorithmic firm to "tie its hands."

These results provide context for an understanding of the potential impact of pricing algorithms. Multi-period commitment alone, regardless of the nature of the algorithm, can lead to higher prices, though profits are disproportionately accrued by the uncommitted rival. By contrast, a speed advantage enables the algorithmic firm to coerce its rival to raise prices, increasing joint profits relative to the Bertrand equilibrium while also shifting more profits to the algorithmic firm. With both (even imperfect) commitment across periods and a speed advantage, the algorithmic firm can raise prices such that it earns more than its share of collusive profits. In some cases, the algorithmic firm may dictate prices high enough such that there is greater deadweight loss and lower welfare than what would be obtained if the firm and its rival were able to collude.

We show how coercive equilibrium extends to settings where (a) the rival is forward-looking, (b) the algorithm employs alternative punishments, including price matching, and (c) the algorithmic firm faces several rivals. In all cases, the coercive equilibrium demonstrates a potential to substantially increase prices. With a price matching algorithm, the collusive outcome is the unique equilibrium when the algorithm enables an immediate reaction. As we discuss, the algorithmic firm may obtain higher profits with our baseline punishment.

In our baseline results, we assume the algorithmic firm's rival is fully informed about the punishment strategy encoded in the algorithm. In an extension, we consider an alternative where the rival firm observes only its own prices and profits and uses gradient learning to max-

[5] In earlier work, we considered this latter case and the general possibility of a speed advantage to increase prices in Markov perfect equilibrium (Brown and MacKay, 2023). This paper did not address multi-period commitment.

imize profits. With this alternative assumption about naive behavior, the rival does not consider the price set by the algorithmic firm (or even necessarily its presence). We show conditions in which the algorithmic firm can still coerce the naive firm to raise prices to supracompetitive levels using a high-speed pricing algorithm that is linear in the rival's price, even though this type of learning results in competitive (Bertrand) prices in a simultaneous game. We derive the optimal linear pricing rule that maximizes the algorithmic firm's profits with commitment and show that gradient learning will always result in supracompetitive prices. Intuitively, the algorithmic firm is able to make it appear as if residual demand facing the naive firm is more inelastic by, e.g., quickly decreasing price when the naive firm tries a price that is lower. The faster the algorithmic firm can adjust prices in response to the rival, the easier it is for the firm to coerce the rival into setting higher prices.

In a second extension, we consider the implications of our model for platform design. We endogenize the algorithm technology parameters that govern speed and commitment, allowing a platform to determine these features for sellers that compete in its market. This is motivated by our observation that platforms have the ability to determine how frequently sellers can update prices and what pricing software they can access. Our analysis shows that a platform that prioritizes producer surplus has an incentive to allow some sellers to have pricing algorithms with commitment and a speed advantage. By doing so, the platform can soften competition on the platform without coordinating behavior of the sellers. If a platform is vertically integrated and competes with a seller on the platform, the platform will have an incentive to use faster pricing and commitment to obtain a competitive advantage.

In practice, pricing algorithms (or "repricing") providers in online markets offer and advertise the two features we focus on. Whether using third-party or proprietary algorithms, firms update the software and pricing rules infrequently, even if the algorithm itself updates prices at high speed. This implies a level of commitment to a strategy across periods. Algorithm providers describe the benefits of commitment in terms of automation to save time and protection against short-sighted behavior.[6]

Algorithm providers also emphasize the value of reacting faster to competitors' price changes. For instance, a firm offering an algorithmic pricing tool notes that "businesses can compete more effectively by responding quickly".[7] Another pricing tool advertises that "a fast reaction to your competitors' price variations is essential to be aggressive and competitive in the world of online commerce."[8] One pricing algorithm provider offers a basic version that updates prices hourly and a premium version that "reacts to changes your competitors make in 90 seconds" in order to "beat competitors with super-fast repricing."[9] Similar pricing algorithms exist in offline mar-

[6]According to one source, a positive aspect of a pricing algorithm is that it "eliminates any rash pricing decisions that you might have made at the moment." See https://www.feedbackwhiz.com/blog/pros-and-cons-of-amazons-automate-pricing-tool/ (accessed February 2025).

[7]See https://dealhub.io/glossary/dynamic-pricing/ (accessed February 2025).

[8]See https://competitoor.com/pricing/the-importance-of-dynamic-pricing/ (accessed February 2025).

[9]See https://www.repricer.com/ (accessed October 2023).

kets, such as retail gasoline. Providers in these markets advertise methods to “automate your process for tracking competitive fuel prices” in “real time.”[10]

Despite the fact that the ability to quickly respond to competitors’ price changes is a key feature of algorithms, there is little theoretical work examining these issues. One exception is Brown and MacKay (2023), who provide a theoretical analysis of competitive (Markov perfect) outcomes when firms may differ in the speed at which they set prices, while documenting that there are substantial differences in pricing frequencies across major online retailers. Empirically, Assad et al. (2024) find that the adoption of pricing algorithms among retail gasoline stations in Germany leads to more frequent price changes, and also higher prices. Byrne et al. (2025) examine what happens when one gasoline retailer could no longer quickly observe and react to rival price changes due to a legal settlement. They find that asymmetries in pricing frequency increase prices, highlighting the importance of differences in pricing speed on equilibrium outcomes.

Our paper is the first to consider the implications when a speed advantage is combined with multi-period commitment. Commitment (with no speed advantage) has been studied in recent theoretical papers about algorithms (Salcedo, 2015; Leisten, 2024; Levine, 2024). Brown and MacKay (2023) show that simultaneous commitment to rules that depend on rival’s prices does not yield competitive (Bertrand-Nash) prices. Lamba and Zhuk (2025) consider an alternating-move setting where commitment yields supracompetitive prices. In empirical work, Musolff (2024) shows that third-party sellers on Amazon demonstrate some degree of commitment to pricing rules, including rules that undercut a rival’s price and reset to a higher price. Our results show that, together, commitment and a speed advantage provide the potential for a much more severe impact on prices, including outcomes that are worse than collusion for consumers.[11]

There is growing concern that autonomous pricing algorithms can learn collusive strategies, potentially facilitating collusion in markets such as online retail (Harrington, 2018). The literature has largely focused on simultaneous-move games with symmetric agents, including a literature considering the potential for collusion with artificial intelligence (Calvano et al., 2020; Asker et al., 2024; Banchio and Mantegazza, 2022).[12] Hansen et al. (2021) examine symmetric firms using misspecified learning algorithms. Miklós-Thal and Tucker (2019) and O’Connor and Wilson (2021) consider the impact of algorithms that provide better demand forecasts on the sustainability of collusion. Asymmetries among agents, e.g., in costs or demand, are generally

[10]See https://www.taigadata.com/front-office-platform/competitive-fuel-pricing/ (accessed February 2025). Historically, retail gasoline managers often observed their competitor’s prices once per day on the way to work in the morning and then manually adjusted prices. See https://www.priceadvantage.com/resources/white-papers/10-fuel-pricing-best-practices/ (accessed February 2025).

[11]Our coercive equilibrium concept does not rely on software algorithms per se. One can imagine other environments with commitment and a speed advantage where coercion may characterize the market outcome, such as when one firm has an information advantage.

[12]Waltman and Kaymak (2008) consider Q-learning in a quantity-setting game. A smaller literature has focused on alternating-move games as in Maskin and Tirole (1988). In particular, Klein (2021) examines collusion with machine learning algorithms in a sequential game.

believed to make collusion more difficult (Scherer, 1980; Tirole, 1988).

We argue that the coercive equilibrium in this paper is more robust than standard repeated-game models of supracompetitive prices, such as collusion. It is well known that in simultaneous-move games, high-frequency pricing implies a larger per-period discount factor, making collusion easier to sustain when firms have perfect monitoring (e.g., Abreu et al., 1991). Yet, all firms must be forward-looking to sustain collusion in these models. In contrast, we make the conservative assumption that the rival simply maximizes current-period profits.[13] Our analysis of pricing algorithms that are linear in rival's price indicates that even naive firms with no knowledge that their rival is using a pricing algorithm can be coerced into raising prices to supracompetitive levels. This shares elements with the idea of strategic manipulation of rival perceptions, as in Fudenberg and Tirole (1986) theory of signal-jamming. In this way, there is broad scope for high-speed pricing algorithms to raise prices.

Our analysis of platform incentives relates to questions pertaining to platform design and third-party incentives when sellers use algorithms. Johnson et al. (2023) examine how platforms can use demand-steering policies to increase competition when sellers use algorithms that collude. Harrington (2022) and Calder-Wang and Kim (2023) address the incentives of third-party pricing algorithm providers to recommend supracompetitive prices to subscribing firms. Our contribution is to consider the regulation of features of independent pricing algorithms.

The paper proceeds as follows. We introduce the model in Section 2. In Section 3, we discuss the equilibrium concept and provide benchmark cases. We characterize the general problem and provide examples in Section 4. In Section 5, we introduce the idea of learning, and we show how the algorithmic firm can obtain coercive equilibria even with simple linear strategies. Section 6 examines the incentives for platform design. Section 7 concludes.

2 Model

We present a duopoly model in which an algorithmic firm can commit to a high-speed algorithm that automatically updates prices, while its rival simply maximizes current-period profits.

2.1 Environment

Two firms, a and b, each produce a single product with prices given by (p_a, p_b). Firm a has a high-speed pricing algorithm while firm b does not. Time is continuous and is given by

[13] Our paper also relates to a previous literature examining games in which a single long-run player faces a succession of repeated short-run players. The classic application is one in which an incumbent faces a new (short-run) potential entrant in each period, as studied by Milgrom and Roberts (1982) and Kreps and Wilson (1982). Fudenberg and Levine (1989) and Fudenberg et al. (1990) provide folk-theorem style analysis for feasible payoffs for a general class of games with a single long-run player. In our setting, the algorithmic firm plays the role of the long-run player while the rival is modeled as a short-run player.

$t \in [0, \infty)$, while periods are defined by discrete intervals indexed by $t \in \{0, 1, 2, ...\}$. We define the length of a period as the frequency with which firm b, the slower firm, can update prices. Thus, firm b can update prices at the beginning of each period.

Demand arrives in continuous time. The instantaneous profit flow function for firm i is time-invariant and is given by $\pi_i(p_i, p_{-i}, x_t)$, where x_t are state variables that affect demand. We assume the profit functions are quasi-concave and have a unique maximum with respect to a firm's own price. We also assume that the products are substitutes and are strategic complements in prices, such that $\frac{\partial \pi_i}{\partial p_{-i}} > 0$ and $\frac{\partial^2 \pi_i}{\partial p_i \partial p_{-i}} > 0$.

2.2 The Algorithmic Firm

The algorithmic firm a maximizes discounted profits over an infinite horizon. Firm a has a discrete discount factor of profits in future periods given by $\beta \in [0, 1)$.[14]

At time $t = 0$, and at future opportunities indexed by τ, firm a chooses a pricing algorithm, $\mathcal{A}_\tau^{\alpha,\gamma}(p_{bt}, x_t, t)$. The algorithm is characterized by speed and commitment parameters, α and γ, which we describe below. In general, an algorithm sets prices according to

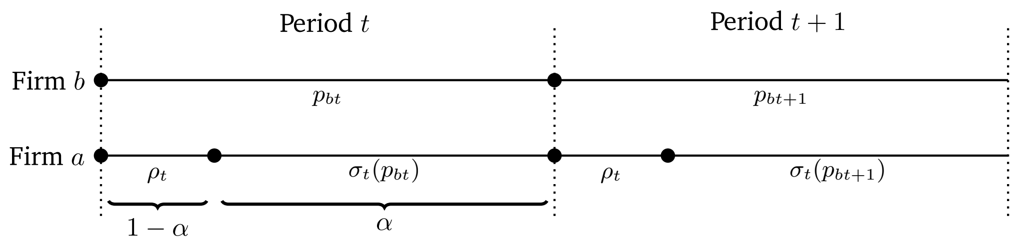
$$p_{at} = \mathcal{A}_\tau^{\alpha,\gamma}(p_{bt}, x_t, t) \equiv \begin{cases} \rho_\tau(x_t) & t - \lfloor t \rfloor \leq 1 - \alpha \\ \sigma_\tau(p_{bt}, x_t) & t - \lfloor t \rfloor > 1 - \alpha \end{cases} \quad (1)$$

where $\lfloor t \rfloor$ is the floor function that yields the greatest integer less than t, i.e., the beginning of the period. The algorithm is characterized by an initial price-setting function $\rho_\tau(x_t)$ and an update function $\sigma_\tau(\hat{p}_{bt}, x_t)$ that can depend on observable state variables, x_t. The initial price-setting component $\rho_\tau(x_t)$ is used to set prices at the beginning of each period, e.g., $p_{a0} = \rho_0(x_0)$ in period 0. Within each period, the faster firm can observe its rivals price, p_{bt}, and the algorithm adjusts price according to the update function, e.g., $p_{at} = \sigma_0(\hat{p}_{bt}, x_t)$. The parameter $\alpha \in [0, 1]$ captures the speed advantage of the algorithmic firm when updating price in response to the rival. The algorithmic firm's initial price is relevant for fraction $1 - \alpha$ of the period and the update function is relevant for fraction α of the period.

These assumptions reflect features of software that is used to update prices automatically. At the beginning of each period, the algorithm may update the price in response to new information captured by x_t. At the same time as these updates, firm b can also change its price, but the decision of firm b is not known in advance by the algorithm. However, the algorithm can observe p_{bt} chosen at the beginning of period t and then update with a lag $1 - \alpha$. One can interpret our equilibrium analysis as conditional on a sequence of state variables; therefore,

[14] For simplicity, we assume there is no within-period discounting, consistent with the idea that the period is very short in many real-world settings with pricing algorithms. However, it is straightforward to redefine α to account for within-period discounting. Consider any α and any instantaneous within-period discount rate ν. Then there exists an objective (non-discounted) speed advantage $\tilde{\alpha}$ such that $\frac{\int_0^{1-\tilde{\alpha}} e^{-\nu t} dt}{\int_0^1 e^{-\nu t} dt} = \frac{1 - e^{-\nu(1-\tilde{\alpha})}}{1 - e^{-\nu}}$. For a given ν, the mapping of α to $\tilde{\alpha}$ is one-to-one.

Figure 1: Timing with Differences in Pricing Speed



Notes: Figure shows timing of pricing when firm a, the algorithmic firm, sets algorithm $\mathcal{A}_t^{\alpha,\gamma}$ at the start of period t and remains committed to the algorithm in period $t+1$. Labels show the relevant prices for each segment given speed advantage $\alpha = 3/4$.

going forward, we suppress x_t in our notation. We return to examining unobserved demand shocks in Section 5.

Figure 1 shows the timing of price adjustments when $\alpha = 3/4$. Both firms can adjust prices at t, firm a can adjust at $t+1-\alpha$, and firm b must wait fraction α until $t+1$ to adjust its price in response. When $\alpha = 0$, the algorithmic firm has no speed advantage and there is simultaneous pricing. When $\alpha = 1$, the algorithmic firm has maximum pricing speed and can instantly react to p_{bt}. In this way, the timing assumptions can be seen as a generalization of standard simultaneous pricing and sequential pricing that nests both as special cases.

There are two ways to motivate the timing of the game. Firms may have differences in pricing systems that allow one firm to set prices more frequently than the other, even when both firms observe their rival's price changes immediately. For instance, if firm a can update price once every 15 minutes and firm b can update price once per hour, then $\alpha = 3/4$, as only the first opportunity for firm a to change its price after observing firm b's price is consequential. An alternative motivation for the timing in our model is that firms differ in terms of how quickly they can observe and react to their rival's price. For example, suppose that both firms can update prices at any time, but firm a observes firm b's price with a 15 minute lag and firm b observes firm a's price with a 1 hour lag. In this case, the only consequential opportunities for firm a to adjust price are concurrently with firm b and 15 minutes later. Under this interpretation of the model, $\alpha = 3/4$ reflects the ratio of firm a's time after observing its rival's price (and before its rival can react) to firm b's time to observe their rival's price. Large differences in the frequency with which firms can update or observe rival prices have been observed in markets such as online retail and retail gasoline (Brown and MacKay, 2023; Byrne et al., 2025).

Firm a has an indefinite commitment to the algorithm over future periods, which is captured by γ. This reflects the fact that a firm re-programs an algorithm only infrequently, even if the algorithm itself can respond at high speed. When the firm is not updating the algorithm, the price is set automatically by the algorithm at each price change opportunity, e.g., $t \in \{1-\alpha, 1, 2-\alpha, 2, 3-\alpha...\}$. At the beginning of each period, with probability γ, the firm remains committed to the algorithm and cannot manually update the algorithm or its price. With probability $1-\gamma$, the firm can update the algorithm, at which point it also picks a new initialization price.

The algorithmic firm's decisions can be expressed as a dynamic problem:

$$V_0(t) = \max_{\mathcal{A}|p_{bt}} (1-\alpha)\pi_a(\rho, p_{bt}) + \alpha\pi_a(\sigma(p_{bt}), p_{bt}) + \beta\gamma V_1(t+1, \mathcal{A}) + \beta(1-\gamma)V_0(t+1) \quad (2)$$

$$V_1(t, \mathcal{A}) = (1-\alpha)\pi_a(\rho, p^*_{bt}) + \alpha\pi_a(\sigma(p^*_{bt}), p^*_{bt}) + \beta\gamma V_1(t+1, \mathcal{A}) + \beta(1-\gamma)V_0(t+1) \quad (3)$$

starting with $t = 0$ and for each integer t thereafter. $V_0(t)$ provides the value function when firm a can update the algorithm, and $V_1(t, \mathcal{A})$ provides the value function when the firm is committed to algorithm $\mathcal{A}$. Here, p^*_{bt} gives the optimal reaction by firm b to $\mathcal{A}$. Firm a can anticipate the optimal response of firm b in future periods while it remains committed to the algorithm.

2.3 The Rival without an Algorithm

We assume that firm b, the rival, sets prices to simply maximize current-period profits.

Assumption A1. *In each period $t \in \{0, 1, 2, ...\}$, firm b solves the problem*

$$\max_{p_b|\mathcal{A}_\tau} (1-\alpha)\pi_b(p_b, \rho_\tau) + \alpha\pi_b(p_b, \sigma_\tau(p_b)) \quad (4)$$

As discussed previously, the period length is defined by the frequency with which firm b updates prices. Therefore, firm b's pricing decision is made with respect to the entire period, and its profit function is based on the full-period average of outcomes resulting from the initial price and any within-period responses by the algorithmic firm. We assume that firm b does not take into account future periods, which, in standard models, implies collusion is not attainable. In this way, assuming firm b does not value future profits is a conservative assumption when analyzing the potential for supracompetitive prices. We discuss the case in which firm b is forward looking in Section 4.3.

Under this assumption, firm b does not respond to the fact that firm a makes a commitment across periods. It does imply that firm b internalizes the algorithmic firm's reaction within the current period, i.e., it understands σ_τ. We consider an extension in which firm b is naive about the use of the algorithm in Section 5.

2.4 Equilibrium

Equilibrium is characterized by a sequence of realized algorithms $\{\mathcal{A}_t\}$ and prices $\{p_{bt}\}$ such that equations (2), (3), and (4) are satisfied for all t. We use a condition akin to the standard

Nash equilibrium condition. When choosing an algorithm, firm a takes as given the current price of firm b. When choosing a price, firm b takes as given the algorithm chosen by a.

With time-invariant profit functions, the problem is stationary. We can exploit the fact that $V_0(t) = V_0(t')$ for $t, t' \in \mathbb{N}$ to express the algorithmic firm's problem from equations (2) and (3) as:

$$\begin{aligned}\tilde{V}(t) = &\max_{\mathcal{A}|p_{bt}} (1-\alpha)\pi_a(\rho, p_{bt}) + \alpha\pi_a(\sigma(p_{bt}), p_{bt}) \\ &+ \sum_{s=t+1}^{\infty} (\beta\gamma)^{s-t} \left((1-\alpha)\pi_a(\rho, p_{bt}^*) + \alpha\pi_a(\sigma(p_{bt}^*), p_{bt}^*)\right)\end{aligned} \tag{5}$$

where $\tilde{V}(t) = \frac{1-\beta}{1-\beta\gamma}V_0(t)$. Going forward, we will make use of the fact that $\sum_{s=t+1}^{\infty}(\beta\gamma)^{s-t} = \frac{\beta\gamma}{1-\beta\gamma}$ to simplify notation. Because the profit function π_a is quasi-concave (with a unique maximum), $\tilde{V}(t)$ is quasi-concave.

Taking into account the response of the rival firm, we can reformulate equation (5) as a constrained optimization problem that specifies whether a target price pair $(p_a^\dagger, p_b^\dagger)$ can be maintained in equilibrium. Specifically, we have firm a choose the target prices $(p_a^\dagger, p_b^\dagger)$ that maximize its discounted profits, subject to the algorithm technology and the incentive compatibility constraint for firm b. This yields the following objective:

$$\max_{(p_a^\dagger, p_b^\dagger)|p_{bt}} (1-\alpha)\pi_a(\rho, p_{bt}) + \alpha\pi_a(\sigma(p_{bt}), p_{bt}) + \frac{\beta\gamma}{1-\beta\gamma}\pi_a(p_a^\dagger, p_b^\dagger) \tag{6}$$

$$\text{s.t.} \quad (i) \quad \rho = p_a^\dagger \tag{7}$$

$$(ii) \quad \sigma(p_b) = \begin{cases} p_a^\dagger & \text{if } p_b = p_b^\dagger \\ P_a(p_b) & \text{if } p_b \neq p_b^\dagger \end{cases} \tag{8}$$

$$(iii) \quad \pi_b(p_b^\dagger, p_a^\dagger) \geq (1-\alpha)\pi_b(\hat{p}_b, \rho) + \alpha\pi_b(\hat{p}_b, \sigma(\hat{p}_b)) \quad \forall \hat{p}_b \tag{9}$$

which is obtained by plugging in the target prices into equation (5).

The objective function is subject to three constraints: (i) firm a chooses the initial price ρ to be equal to its target price, (ii) the update function provides firm a's target price as long as firm b follows its target price, and (iii) the target price for firm b satisfies its incentive compatibility constraint.

When firm b does not choose $p_{bt} = p_b^\dagger$, the update function of the algorithm follows a potentially arbitrary punishment function, $P_a(p_b)$. For our main results, we assume that the punishment function is simply the one-shot best-response function for firm a:

Assumption A2. *The punishment function is equal to firm a's static best-response function, $P_a(\cdot) = R_a(\cdot)$.*

It is typical in the literature on collusion to assume punishment strategies that are consistent with short-run non-cooperative behavior, and we follow that convention. We consider alternative punishment functions in Section 4.3.

There is a unique Markov perfect equilibrium characterized by the choice of $\mathcal{A}$ and p_b that satisfy the above conditions. Uniqueness is obtained under assumptions A1 and A2 when the profit functions are well-behaved.[15] The rival, by assumption, does not account for future profits or respond to the history of play. Thus, its presence eliminates a large class of equilibria that can be supported in repeated games. We illustrate this with benchmark cases in the following section.

3 Benchmark Cases

Here, we present four benchmark equilibria under different assumptions about pricing speed and commitment. These benchmark cases help to build intuition for these features and motivate the general analysis that we present in Section 4. First, we consider the case when the algorithmic firm has no speed advantage and no commitment. We then consider commitment only and a speed advantage only as separate cases. The case with a speed advantage but no multi-period commitment corresponds to the asymmetric commitment model analyzed in Brown and MacKay (2023). We then consider commitment and speed advantage together in the limiting case of maximal coercion.

For comparison, we will consider the outcome that maximizes joint profits, which we refer to as the *collusive outcome* or *collusion*. This is the outcome if the rivals could create a perfect cartel or if they merged and became a multi-product monopolist. The collusive outcome is obtained when solving the objective function $\max_{(p_a,p_b)} \pi_a(p_a,p_b) + \pi_b(p_b,p_a)$. We define each firm's *share of collusive profits* as the profits it earns at the collusive prices.

3.1 Simultaneous Pricing

Consider the case when $\alpha = 0$ and $\gamma = 0$, so that the algorithm provides no commitment and no speed advantage. The objective functions become

$$\text{Firm } a : \max_{\mathcal{A}|p_{bt}} \pi_a(\rho, p_{bt}) \tag{10}$$

$$\text{Firm } b : \max_{p_b|\mathcal{A}_\tau} \pi_b(p_b, \rho_\tau) \tag{11}$$

which corresponds to the one-shot simultaneous price-setting game. Thus, the only subgame perfect equilibrium is the Bertrand-Nash equilibrium. Though dynamic price-setting games

[15] Under more general forms for the punishment function, multiple equilibria can be obtained; thus, assumption A2 can alternatively be viewed as a device for equilibrium selection.

may, in general, yield multiple equilibria, the presence of a firm that maximizes current-period profits greatly reduces the set of outcomes that can be sustained in equilibrium.

3.2 Multi-Period Commitment Only

In some settings, the algorithm may provide a commitment advantage but no speed advantage. This is the case in the literature on learning algorithms and competition (e.g., Calvano et al., 2020; Asker et al., 2024; Johnson et al., 2023), in which firms commit to learning algorithms that set prices simultaneously. The analysis here can be roughly thought of as an extension of these models, where one firm can endogenously choose the optimal algorithm and the other firm uses a learning algorithm that recovers its true payoffs. We will not discuss learning here but instead describe the long-run payoffs.

With no speed advantage, $\alpha = 0$. Firm a has the objective function

$$\max_{\mathcal{A}|p_{bt}} \underbrace{\pi_a(\rho, p_{bt})}_{\substack{\text{Simultaneous}\\ \text{Pricing Incentive}}} + \underbrace{\frac{\beta\gamma}{1-\beta\gamma}\pi_a(\rho, p^*_{bt})}_{\substack{\text{Leader}\\ \text{Pricing Incentive}}} \tag{12}$$

while firm b maximizes $\max_{p_b|\mathcal{A}_\tau} \pi_b(p_b, \rho_\tau)$. The objective function for firm a differs from that of the one-shot benchmark due to the term $\frac{\beta\gamma}{1-\beta\gamma}\pi_a(\rho, p^*_{bt})$, which is positive as long as $\beta\gamma > 0$. The algorithmic firm balances the profits in the current period, conditional on p_{bt}, against the profits in future periods where the rival with slower pricing might update its price.

The second component in the objective captures the pricing incentive of a leader in a sequential pricing game. In equilibrium, p^*_{bt} will be given by firm b's static best-response function, $p^*_{bt} = R_b(\rho)$. Because a anticipates profits in future periods that it is committed to the algorithm, it internalizes the reaction of its rival. We refer to the combined term $\beta\gamma$ as *commitment*, as β and γ enter the model as a pair. We define *full commitment* as the limit as $\beta\gamma$ goes to 1. In this case, firm a becomes infinitely patient and has perfect commitment and the outcome is equivalent to a sequential price-setting game in which firm a is the (Stackelberg) leader. For example, in a setting in which firms update prices once per day, and firm a updates its algorithm once per month on average, the outcome may be similar to sequential pricing, as firm a will primarily internalize profits over future periods.

More generally, the equilibrium can be characterized as follows:

Proposition 1. *When the algorithm enables commitment ($\gamma > 0$) but no speed advantage ($\alpha = 0$), the equilibrium lies along firm b's best response function between the simultaneous price-setting equilibrium and the sequential price setting equilibrium where the algorithmic firm is the leader.*

When firms produce substitute goods and prices are strategic complements, both firms realize higher prices compared to the price-setting (Bertrand-Nash) equilibrium. This follows similar logic to Proposition 2 of Brown and MacKay (2023). In contrast to the asymmetric commitment model of Brown and MacKay (2023), when the firms have identical profit functions, the algorithmic firm has a higher price than the rival.

All proofs are in Appendix A.1.

3.3 Faster Pricing Only

We now consider the case in which firm a has no multi-period commitment ($\gamma = 0$) but the algorithm enables faster pricing updates ($\alpha > 0$). The period t objective function for firm a when it can update its algorithm becomes

$$\max_{\mathcal{A}|p_{bt}} \underbrace{(1-\alpha)\pi_a(\rho, p_{bt})}_{\substack{\text{Simultaneous} \\ \text{Pricing Incentive}}} + \underbrace{\alpha\pi_a(\sigma(p_{bt}), p_{bt})}_{\substack{\text{Follower} \\ \text{Pricing Incentive}}} \tag{13}$$

while the objective function for firm b remains as given in equation (4).

This problem is equivalent to the asymmetric commitment model analyzed by Brown and MacKay (2023). Following that analysis, it is weakly dominant for firm a to choose an update function that corresponds to its static best-response function, $\sigma(\cdot) = R_a(\cdot)$, which satisfies our assumption A2.

In contrast to the above case with only multi-period commitment, the algorithmic firm balances the simultaneous price-setting incentive with the sequential price-setting incentive where it acts as the *follower*. Thus, following Proposition 2 from Brown and MacKay (2023), the equilibrium lies on the algorithmic firm's best-response function, between the simultaneous and the sequential equilibrium. The parameter α indicates how much weight the rival with slower pricing puts on the portion of the period after the algorithm update. As above, this results in higher prices for both firms when the products are substitutes and prices are strategic complements. However, this case will yield lower prices for the algorithmic firm, instead of higher prices, when the firms have identical profit functions.

In the limiting case where $\alpha = 1$, the algorithm yields the sequential price-setting equilibrium, with firm b acting as the leader and firm a acting as the follower. Thus, the two features of the algorithms we study—speed and multi-period commitment—can generate sequential equilibria where the algorithmic firm takes on either the leader or follower role.

3.4 Maximal Coercion

Our model introduces general *coercive* equilibria in which firm a may have both a speed advantage ($\alpha > 0$) and commitment ($\beta\gamma > 0$). As we show, the combination of these features generates distinct outcomes that benefit the algorithmic firm. Before considering the general model in Section 4, we focus on the limiting case in which $\alpha = 1$ and $\beta\gamma$ becomes arbitrarily

Figure 2: Benchmark Equilibria, with Examples

	No Commitment ($\beta\gamma = 0$)	**Full Commitment** ($\beta\gamma \to 1$)
No Speed Advantage ($\alpha = 0$)	Simultaneous Bertrand-Nash $(p_a, p_b) = (1.00, 1.00)$	Sequential, Firm a is Leader $(p_a, p_b) = (1.14, 1.05)$
Fastest Pricing ($\alpha = 1$)	Sequential, Firm a is Follower $(p_a, p_b) = (1.05, 1.14)$	Maximal Coercion $(p_a, p_b) = (1.93, 2.15)$

close to 1. In other words, firm a is infinitely patient, has full commitment, and can instantaneously react to the price of firm b in any period. We call this limiting case *maximal coercion*.

Under maximal coercion, firm a's constrained optimization problem can be expressed as

$$\begin{aligned} &\max_{(p_a^\dagger, p_b^\dagger)} \pi_a(p_a^\dagger, p_b^\dagger) \\ &\text{s.t.} \quad \pi_b(p_b^\dagger, p_a^\dagger) \geq \max_{p_b} \pi_b(p_b, P_a(p_b)) \end{aligned} \tag{14}$$

where $P_a(\cdot) = R_a(\cdot)$ following Assumption A2. Firm a chooses a target price pair $(p_a^\dagger, p_b^\dagger)$ such that the incentive compatibility constraint for firm b, given by the second line, holds. The incentive compatibility constraint indicates that firm b can deviate from the target price $p_b^\dagger$ and receive profits that depend on the punishment function $P_a(\cdot)$ of firm a's algorithm. Given that the punishment function is equal to firm a's static best-response function, firm b earns profits at least as great as the sequential leader.[16]

3.5 Numerical Example

We illustrate our results with examples generated from a simple symmetric linear demand system given by

$$D_i(p_i, p_{-i}) = 1 - \left(\frac{1}{4} + \frac{d}{2}\right) p_i + \frac{d}{2} p_{-i} \tag{15}$$

where $d \geq 0$ is an inverse measure of product differentiation. This demand system can be derived from the quasilinear quadratic utility model (Singh and Vives, 1984). The goods do not compete when $d = 0$ and are perfect substitutes when $d = \infty$. Without loss of generality, we normalize marginal costs to zero.

Figure 2 provides the benchmark cases when $d = 1$. Prices are significantly higher under maximal coercion compared to the other cases. In the sequential benchmark, prices are 5 percent and 14 percent higher for the two firms relative to the Bertrand-Nash equilibrium.

[16]Under standard smoothness conditions, the condition will bind exactly.

With maximal coercion, the prices are 93 percent and 115 percent higher than Bertrand-Nash.

The maximal coercion equilibrium is obtained at the target price pair $(p_a^\dagger, p_b^\dagger)$ that solves equation (14). Here, the solution is $(p_a^\dagger, p_b^\dagger) = (1.93, 2.15)$, which is obtained with the update function

$$\sigma(p_{bt}) = \begin{cases} 1.93 & \text{if } p_{bt} = 2.15 \\ (2 + p_{bt})/3 & \text{if } p_{bt} \neq 2.15 \end{cases} \tag{16}$$

Firm b sets price knowing that firm a's algorithm will instantly update according to this rule. If firm b sets the target price of 2.15, then its profit is $\pi_b(2.15, 1.93) = 0.76$. If firm b deviates from the target price, then firm a punishes by best responding, given by $(2 + p_{bt})/3$. Firm b has no incentive to deviate from the target price since solving $\max_{p_b} \pi_b(p_b, (2 + p_b)/3)$ yields the same profit as setting the target price.[17]

The implications of maximal coercion for profits are substantial. The algorithmic firm earns profits of 0.75 in the Bertrand-Nash, 0.76 as the sequential leader, and 0.82 as the sequential follower. In the coercive equilibrium, it earns 1.21. Thus, moving from the Bertrand-Nash benchmark to maximal coercion increases the profits for firm a by 61 percent.

The maximal coercion outcome also benefits the algorithmic firm relative to the collusive outcome. Given demand, collusive prices are $(p_a, p_b) = (2, 2)$. With maximal coercion, firm b is incentivized to set a price higher than the collusive price (2.15 versus 2) and the algorithmic firm obtains higher profits for itself than its share of collusive profits (1.21 versus 1). In contrast, firm b earns lower profits than its share of collusive profits.

4 Equilibrium Outcomes

4.1 Prices and Profits

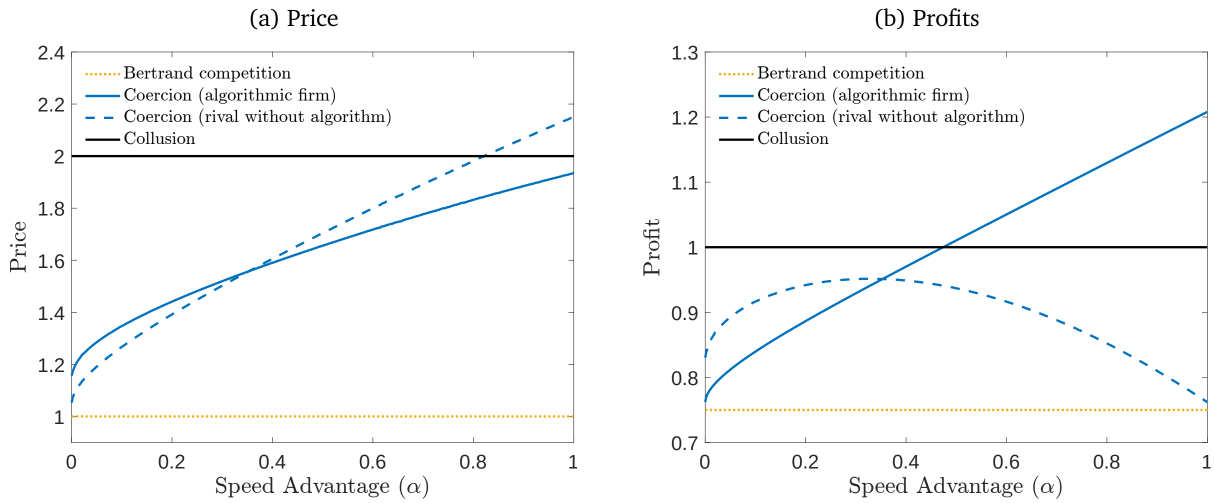
We examine results for the coercive equilibrium. For illustrations, we use the demand system from Section 3.5 with a differentiation parameter of $d = 1$.[18] Figure 3 presents equilibrium prices (panel (a)) and profits (panel (b)) in the case with full commitment ($\beta\gamma \to 1$) and different values for the algorithmic firm's speed advantage. The solid blue line represents the algorithmic firm, while the dashed blue line represents the firm without the algorithm. The black line shows joint profit-maximizing prices and profits that could be obtained under collusion or if there was a multi-product monopolist. For comparison, we also plot the prices and profits for Bertrand competition (yellow dotted line).

From this comparison, we obtain the following two results:

[17] Firm b's optimal deviation from the maximal coercion equilibrium is to choose a price of 1.14, which would imply that firm a's algorithm sets a price of 1.05. These are the sequential leader and follower prices.

[18] We also simulate the same plot using logit demand and the patterns are quite similar. See Appendix Figure A-7.

Figure 3: Prices and Profits in Coercive Equilibrium, by Algorithmic Firm's Pricing Speed



Notes: Panel (a) shows prices and panel (b) shows profits under Bertrand competition with simultaneous pricing, coercion with full commitment ($\beta\gamma \to 1$), and joint profit maximization. Figure shows the equilibrium for different values of speed advantage α. Assumes $d = 1$ under linear demand given by equation (15).

Remark 1. *In the coercive equilibrium, prices for some products can be higher than the prices that maximize joint profits.*

Remark 2. *In the coercive equilibrium, the algorithmic firm can earn greater than its share of profits under joint profit maximization.*

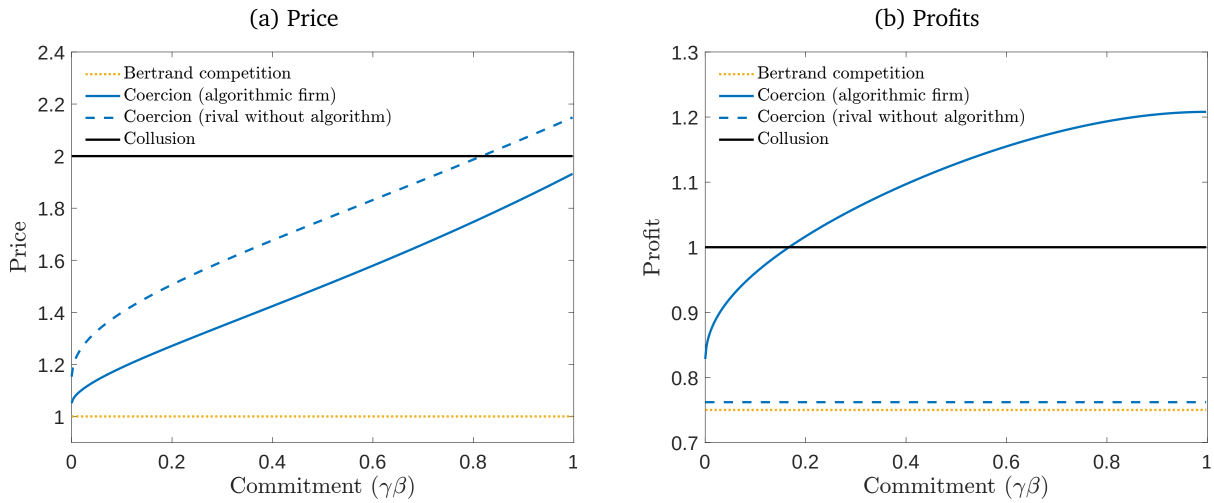
For our example, joint profits are maximized when prices are set to 2, yielding profits of 1 for each product. Panel (a) shows that the price of the rival firm exceeds this level under full commitment when the speed advantage for the algorithmic firm is sufficiently high (roughly $\alpha > 0.8$). The algorithmic firm has an incentive to coerce its rival to set a price greater than the collusive level so that consumers substitute to the algorithmic firm's product. With full commitment and a speed advantage such that α is roughly more than 0.5, the algorithmic firm earns more than its share of collusive profits.

We have shown that a speed advantage within periods and commitment across periods can change the feasible profit set. The next result establishes sufficient conditions under which the algorithmic firm can capture more than its share of collusive profits.

Proposition 2. *There exists values $\overline{\alpha}$ and $\overline{\beta\gamma}$ such that, for $\alpha > \overline{\alpha}$ and $\beta\gamma > \overline{\beta\gamma}$, the algorithmic firm earns profits greater than its share of profits under joint profit maximization, provided that profits for firm b are higher with collusive prices than when firm b is the sequential leader.*

When the algorithm can respond quickly and the level of commitment is high, the collusive outcome is incentive compatible for firm b if it yields more profits than what it would obtain by deviating (and facing the "punishment" by firm a's algorithm). Hence, we require the firm's

Figure 4: Prices and Profits in Coercive Equilibrium, by Algorithmic Firm's Commitment



Notes: Panel (a) shows prices and panel (b) shows profits under Bertrand competition with simultaneous pricing, coercion with maximum speed ($\alpha = 1$), and joint profit maximization. Figure shows the equilibrium for different values of commitment, $\beta\gamma$. Assumes $d = 1$ under linear demand given by equation (15).

share of collusive profits to exceed those received when it is sequential leader, which occurs for common models of demand. If the incentive compatibility constraint is slack at collusive prices, firm a can raise its own target price or lower the rival's target price to increase its own profits.

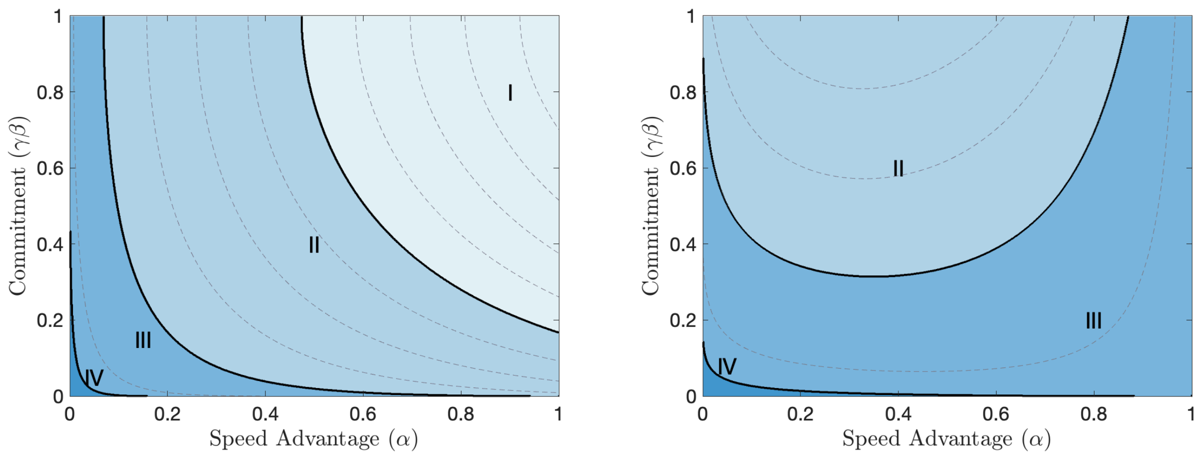
Figure 3 indicates that there are three regions that characterize the relative prices of the algorithmic firm and its rival. In the limiting case with no speed advantage, we obtain the sequential-move equilibrium as discussed in Section 2. With a weak speed advantage, it is optimal for the algorithmic firm to lead with a higher price than the rival and to use its threat of punishment to prevent the rival from lowering its price further. For an intermediate range of pricing speed, the algorithmic firm is able to incentivize the slower rival to set a higher price than the algorithmic firm but cannot coerce the rival to set a price above the fully collusive price. With very fast pricing, the algorithmic firm can coerce its rival into setting prices above the fully collusive price. In all cases, prices of both firms are above the Bertrand price.

When the algorithmic firm's speed advantage is small, the rival earns greater profits than the algorithmic firm. Around $\alpha = 0.4$, the lines for firm a and firm b intersect. These values reflect the case when the incentives from commitment and the pricing advantage balance each other out, yielding symmetric prices and profits.

Figure 4 shows the equilibrium cases when commitment ($\beta\gamma$) varies and $\alpha = 1$. For these cases, the algorithmic firm always prices lower than the rival and earns greater profits. Panel (b) shows that the rival only earns profits equal to the sequential leader profits—close to the competitive profits—in all cases. Thus, the algorithmic firm can use the threat of an immediate reaction to incentivize the rival to set higher prices and extract nearly all of the resulting

Figure 5: Profits by Pricing Speed and Commitment

(a) Algorithmic firm (firm a) (b) Rival without algorithm (firm b)



Notes: Panels (a) and (b) show profit regions for firm a and firm b for different values of the speed advantage (α) and commitment ($\beta\gamma$). Region I indicates profits greater than under joint profit maximization (splitting collusive profits), region II indicates profits greater than those of a sequential follower, region III indicates profits greater than those of a sequential leader, and region IV indicates profits greater than under Bertrand competition. Dashed lines show additional isoprofit curves, each indicating increments equal to one-fourth of the difference between the collusive profits and the sequential follower profits. Assumes $d = 1$ under linear demand given by equation (15).

producer surplus. The extent to which the algorithmic firm can do this in equilibrium depends on its ability to commit. When commitment is low, firm a has a short-run incentive to reduce prices given the high prices of its rival.

For this example, equilibrium prices are supermodular in speed and commitment. With only commitment or only a speed advantage, the algorithmic firm is bounded in its ability to raise prices to the sequential prices and payoffs. Appendix Figures A-3 and A-4 illustrate the cases of $\beta\gamma = 0$ and $\alpha = 0$, respectively, showing a modest increase in prices and profits. In the presence of both features—as illustrated by moving left to right in Figure 3 or Figure 4—an algorithmic firm can obtain substantially higher prices and profits. These complementarities are apparent at intermediate values of commitment and speed advantage, as illustrated in Appendix Figures A-5 and A-6. For example, the algorithmic firm can obtain profits greater than collusive profits if either $\beta\gamma = 0.5$ or $\alpha = 0.5$.

Figure 5 plots the profits obtained under all combinations of commitment and pricing speed. Profit regions for the algorithmic firm are shown in panel (a) and those for the slower rival are shown in panel (b). Region I indicates profits greater than symmetric collusion, region II indicates profits greater than those obtained by a sequential follower, region III indicates profits greater than those obtained by a sequential leader, and region IV indicates profits greater than under Bertrand competition.

For this demand system, the algorithmic firm can obtain profits greater than the sequential leader for almost all values of α and $\beta\gamma$. Moreover, many combinations of commitment and speed can allow the algorithmic firm to obtain greater profits than the symmetric collusion payoffs (region I). The slower rival also earns profits greater than the sequential leader for almost all values of α and $\beta\gamma$, but the potential increase in profits is lower than that of firm a, never exceeding the share of profits obtained by joint profit maximization. Appendix Figure A-8 shows that we obtain similar patterns for the maximal coercion equilibrium for different values of (inverse) product differentiation, d. When the products are closer substitutes, coercion has a larger effect, as measured by the price increase over the Bertrand equilibrium, the decrease in consumer surplus relative to the Bertrand equilibrium, and the increase in profits over symmetric collusion for the faster firm.

These examples illustrate how speed and commitment benefit the algorithmic firm. We formalize this for the generic case with the following propositions:

Proposition 3. *For the algorithmic firm, profits are increasing in the speed advantage (α).*

Proposition 4. *For the algorithmic firm, profits are increasing in commitment ($\beta\gamma$).*

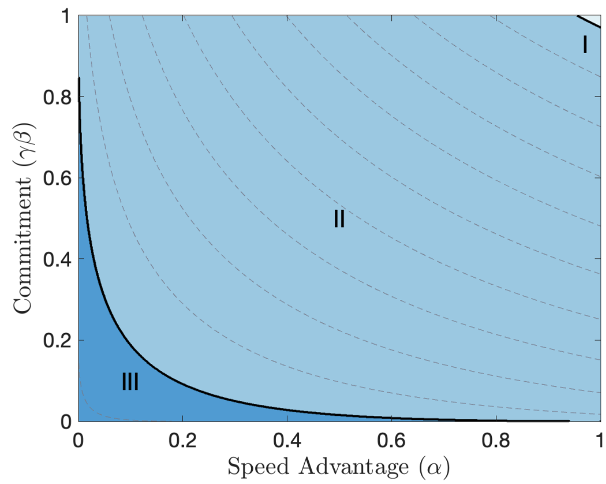
Intuitively, a speed advantage within each period and commitment across periods are tools that confer greater coercive power to the algorithmic firm. Greater speed allows the algorithmic firm to punish more rapidly and coerce its rival to set higher prices. Greater commitment allows the algorithmic firm to set a high price at the beginning of the period for its long-run benefit at the expense of short-run profits.

As a corollary, firm a's profits are greatest with maximal coercion: $(\alpha,\beta\gamma)=(1, 1)$. Equation (14) illustrates why this is the case. Firm a maximizes long-run profits subject to a single constraint, which is the least restrictive when $\alpha = 1$.

4.2 Consumer Surplus

Figure 6 plots consumer surplus regions for different combinations of commitment and speed advantage. Here, the regions are ordered from lowest to greatest consumer surplus. Region I indicates consumer surplus less than under collusion, region II indicates consumer surplus less than the sequential-move equilibrium, and region III indicates consumer surplus less than under Bertrand competition. In this environment, consumers are always weakly worse off than they would be under the Bertrand-Nash equilibrium. For most cases with this demand system, consumer surplus is lower than under the sequential-move equilibrium. The dashed lines indicate increments of consumer surplus equal to one-tenth of the difference between consumer surplus under collusion and under the sequential-move equilibrium. Within region II, there is a gradient reflecting a steady decline in consumer surplus as the speed advantage or commitment increases. Consumer surplus can be closer to collusive levels than Bertrand levels even with a modest speed advantage or modest degree of commitment.

Figure 6: Consumer Surplus by Pricing Speed and Commitment



Notes: Figure shows consumer surplus regions for different values of the speed advantage (α) and commitment ($\beta\gamma$). Region I indicates consumer surplus less than under collusion, region II indicates consumer surplus less than the sequential-move equilibrium, and region III indicates consumer surplus less than under Bertrand competition. Dashed lines indicate increments of consumer surplus equal to one-tenth of the difference between consumer surplus under collusion and under the sequential-move equilibrium. Assumes $d = 1$ under linear demand given by equation (15).

When $\beta\gamma$ and α are both close to one, consumer surplus is lower than that obtained under symmetric collusion (indicated by region I). Since the outcomes in region I also yield lower joint profits than collusion, total welfare is also lower than the collusive outcome. We state this as our next result:

Remark 3. *Consumer surplus and total welfare can be lower in the coercive equilibrium than under joint profit maximization.*

This result reflects the fact that the algorithmic firm has the power to coerce the rival to set a price higher than the collusive price when the speed advantage and degree of commitment are high. Even though the algorithmic firm sets a price slightly lower than the collusive price, consumers are harmed overall.

This analysis highlights how two features of algorithms—speed and commitment—provide a substantial advantage to the adopting firm, especially when combined. In the following sections we highlight that this finding is robust to a number of alternative assumptions.

4.3 Alternative Assumptions

We now consider how alternative assumptions affect our analysis. We consider a rival that places value on future profits, alternative punishment functions used by the algorithm, and moving from a duopoly setting to an oligopoly setting in which the algorithmic firm faces several rivals.

Forward-Looking Rival

Assumption A1 that firm b is short-sighted helps deliver a unique equilibrium and rules out dynamic strategies that are used to sustain collusive outcomes. Here, we demonstrate that the algorithmic coercion equilibrium persists even when the rival is forward-looking.

Suppose instead that firm b maximizes the sum of discounted profits. There exists an equilibrium strategy where neither firm b nor firm a conditions on the history of play. The optimal strategy of this form has firm b solving the following dynamic problem:

$$V_b(t, \mathcal{A}_\tau) = \max_{p_b|\mathcal{A}_\tau}(1-\alpha)\pi_b(p_b, \rho_\tau) + \alpha\pi_b(p_b, \sigma_\tau(p_b)) + \hat{\beta}\gamma V_b(t+1, \mathcal{A}_\tau) + \hat{\beta}(1-\gamma)V_b(t+1, \mathcal{A}_{t+1})$$

where $\hat{\beta}$ is firm b's discount factor and $\mathcal{A}_{t+1}$ denotes the algorithm chosen by firm a in period $t+1$. The evolution of the state to period $t+1$ is independent of the choice of p_b in period t. Thus, a policy of choosing p_b to maximize current-period profits is an equilibrium strategy and the coercive equilibrium we characterize persists when firm b is forward-looking. When firm b is forward-looking, we can no longer guarantee that there is a unique equilibrium. There may be equilibria where both firms condition on the history of play, similar to traditional collusive equilibria.

Alternative Punishment Functions

Our baseline model shows that a best-response punishment by the algorithmic firm is sufficient to coerce its rival to set a significantly higher price. The assumption that the punishment is the static best response is conservative and the algorithmic firm could potentially commit to a harsher punishment. Appendix Figure A-9 shows the equilibrium under an alternative to Assumption A2 in which firm a can punish by setting price equal to marginal cost ($p_a = 0$ in the example). The relationship between prices and pricing technology is similar; however, greater punishment allows firm a to coerce its rival into setting even higher prices, resulting in higher profits for the algorithmic firm for a given level of commitment.

Next, we consider an alternative assumption that the algorithmic firm employs price matching for its punishment function, $P_a(p_b) = p_b$. Price matching, anecdotally, has been used in online markets, and the literature has noted that such strategies could increase prices (e.g., Salop, 1986). Price matching is a less aggressive punishment than our baseline best-response function. Equilibrium outcomes are shown in Appendix Figure A-10.

Similar to the other punishment strategies, prices and profits are increasing in the speed advantage. However, when the algorithm can immediately react and the profit functions are symmetric, the unique outcome is the collusive price, regardless of the level of commitment. Intuitively, the algorithmic firm has to offer firm b profits greater than what it could obtain by deviating. When firm b deviates and faces a price-matching punishment, it deviates to the collusive price. These simulations imply that the algorithmic firm would prefer to use a more aggressive punishment, such as the static best response, when commitment is high because it provides the firm with more than the collusive profits. However, there are cases with low commitment in which the algorithmic firm may prefer to use a price matching approach.

We also consider a punishment function that is linear in rival's price in the following section. As we show, this punishment may be particularly relevant when the rival uses a naive learning strategy.

N-firm Oligopoly

Our analysis can be extended to a more general oligopoly setting in which a single algorithmic firm faces several rivals that do not use algorithms. We characterize the objective function of the algorithmic firm facing multiple rivals in Appendix Section A.2. We show that speed and commitment play similar roles for the case of a single algorithmic firm and two rivals. We then examine how the maximal coercion equilibrium changes as the number of rivals increases. For illustration, we use an extension of the demand system in equation (15). Appendix Figure A-2 shows that prices under maximal coercion are decreasing in the level of competition, but the decrease is gradual. The coercive equilibrium leads to prices that are still 18 percent higher than Bertrand prices with 10 rival firms.

5 Extension: Incorporating Learning

In the above analysis, we show how a firm with a high-speed pricing algorithm may unilaterally implement supracompetitive prices when the slower rival understands (explicitly or implicitly) the potential punishment strategy and resulting profits. We now consider the case in which the slower rival is potentially uninformed about the strategy used by the algorithmic firm or its own profit function. Instead, the rival learns over time by optimizing over price. We ask whether the algorithmic firm can induce higher prices without announcing a strategy. We then simulate learning to shed light on the speed of convergence to the long-run equilibrium. We show that these strategies still converge on average in the presence of time-varying demand shocks.

5.1 Simple Learning and Linear Strategies

In many settings, it may be reasonable to assume that an equilibrium arises not from introspection by the players but rather from an iterative process of adaptive learning (Fudenberg and Levine, 2016). Firms may not even internalize the fact that they are playing a game with a strategic rival.[19] Models of learning in economic games include fictitious play, reinforcement

[19]Given the complexity and number of products, online retailers are known to conduct pricing experiments assuming no strategic response by rivals (Hansen et al., 2021).

learning, and gradient learning. Different learning models can have different convergence properties. The recent literature has shown that different classes of learning algorithms may or may not converge to Bertrand-Nash equilibria.[20]

We focus on gradient learning by the slower firm. Gradient learning is a particularly naive strategy that requires minimal inputs. The slower rival does not take into account the actions or strategies of the algorithmic firm, nor does it form beliefs about the economic environment. Gradient learning captures the fact that many firms simply adjust prices in the direction that increases current-period profits until profits are maximized. In particular, we assume that the slow firm starts with two guesses for its optimal price, $p_{b0} = \hat{p}_b^0$ and $p_{b1} = \hat{p}_b^1$, with $\hat{p}_b^0 \neq \hat{p}_b^1$. For period $t \geq 1$, the firm updates its price following

$$p_{b(t+1)} = p_{bt} + \lambda \widehat{\frac{\partial \pi_b}{\partial p_b}}\Big|_{p_{bt}} \tag{17}$$

where $\widehat{\frac{\partial \pi_b}{\partial p_b}}$ is the firm's estimate of the gradient of its profits with respect to its own price.

Gradient learning is also closely related to A/B testing in which firms use price experiments to determine whether to raise or lower prices, a commonly used approach in online markets. It has desirable properties for our purposes. In a simultaneous game in which both firms use gradient learning, the strategies converge to the Bertrand-Nash equilibrium as long as price adjustments are relatively smooth (λ is not too large).[21] We consider the case in which the slower rival employs gradient learning and ask whether the algorithmic firm can implement a pricing strategy that results in supracompetitive profits. Throughout, we assume λ is small enough to ensure convergence.

In contrast to the previous sections, we assume here that the algorithmic firm adopts a linear punishment strategy instead of the discontinuous trigger strategy in Assumption A2. In particular, for a target price pair $(p_a^\dagger, p_b^\dagger)$, the faster firm chooses $p_a = p_a^\dagger$ at the beginning of the period and then updates its price according to the linear pricing rule:

$$\sigma(p_b) = \begin{cases} p_a^\dagger - \phi(p_b^\dagger - p_b) & \text{if } p_a^\dagger - \phi(p_b^\dagger - p_b) \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

Thus, the punishment depends on how far from the target price the slower rival deviates, and the degree of punishment is captured by ϕ. Note that, by assumption, $\sigma(p_b^\dagger) = p_a^\dagger$. We focus on linear strategies because the linearity helps ensure that experimentation by the slower firm will converge to the desired price of the faster firm as we show below. In practice, pricing strategies

[20]For example, the Q-learning algorithms studied in Calvano et al. (2020) do not generally converge to Bertrand-Nash, while Asker et al. (2024) show that imposing additional assumptions on this class of algorithms does lead to convergence.

[21]See Anufriev et al. (2013).

that are linear in rivals' prices are common.[22]

Throughout, we assume that the algorithmic firm can fully commit to the linear pricing rule given by equation (18). This also implies that the firm does not adjust its strategy to manipulate the rate of learning of the slower firm.

5.2 Coercive Linear Strategies with Simple Learning

We now solve for the pricing rule and equilibrium prices. The algorithmic firm attempts to induce the target price vector $(p_a^\dagger, p_b^\dagger)$. To constrain the slope of the reaction by the algorithmic firm, we assume that the linear slope of the pricing rule passes through the point $(0, 0)$ and the target price vector. Linear strategies of this form have the property that the faster firm's price changes in response to any non-negative price chosen by the slower rival. This implies $\phi = \frac{p_a^\dagger}{p_b^\dagger}$. For expositional clarity, we assume $p_b \geq 0$.

Under these assumptions, the objective function can be written as:

$$\max_{(p_a^\dagger, p_b^\dagger)} \pi_a(p_a^\dagger, p_b^\dagger) \tag{19}$$

$$\text{s.t.} \quad p_b^\dagger = \underset{p_b|\mathcal{A}}{\operatorname{argmax}} (1-\alpha)\pi_b(p_b, p_a^\dagger) + \alpha\pi_b(p_b, \phi p_b) \tag{20}$$

where the initial price in each period, ρ, is equal to $p_a^\dagger$, and where the update function is given by $\sigma(p_b) = \phi p_b$. As before, the algorithmic firm chooses a target price vector subject to the incentive compatibility constraint of the slower rival.

The slower rival's first-order condition can be expressed as

$$(1-\alpha)\frac{\partial \pi_b(p_b, p_a^\dagger)}{\partial p_b} + \alpha\frac{\partial \pi_b(p_b, \phi p_b)}{\partial p_b} + \alpha\phi\frac{\partial \pi_b(p_b, \phi p_b)}{\partial p_a} = 0 \tag{21}$$

Because $\phi = \frac{p_a^\dagger}{p_b^\dagger}$, the algorithmic firm's pricing rule can be written in terms of the implicit function $p_b^*(p_a^\dagger)$ that solves the above first-order condition. Let $p_a^*(p_b^\dagger)$ be the algorithmic firm's price as a function of the target $p_b^\dagger$ when the constraint $p_b^\dagger = p_b^*(p_a^\dagger)$ holds. Given the implicit functions p_a^* and p_b^*, we can express the algorithmic firm's problem as

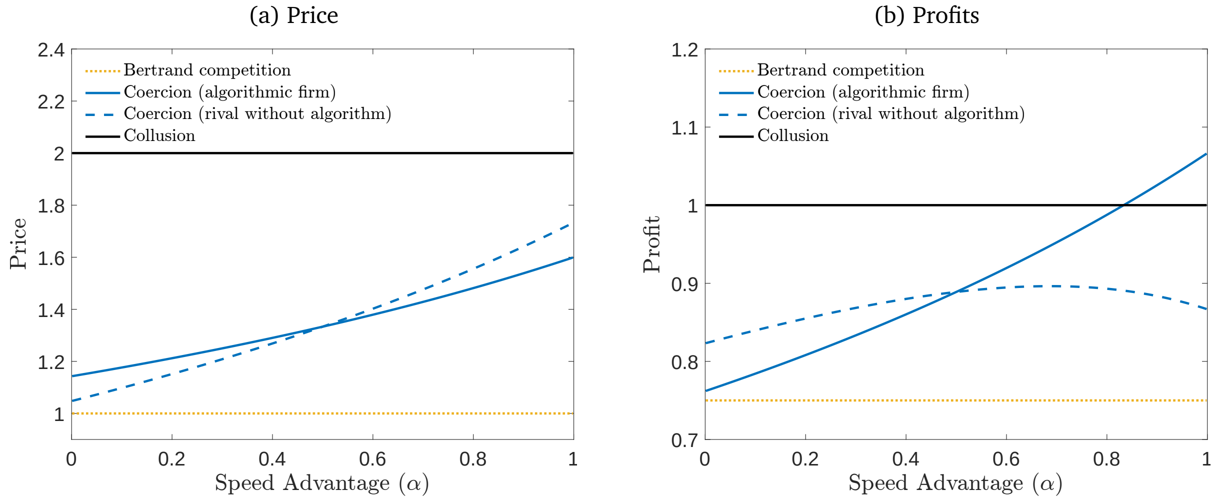
$$\max_{p_b^\dagger} \pi_a(p_a^*(p_b^\dagger), p_b^\dagger). \tag{22}$$

The solution is given by the first-order condition

$$\frac{\partial \pi_a(p_a^*(p_b^\dagger), p_b^\dagger)}{\partial p_a}\frac{\partial p_a^*(p_b^\dagger)}{\partial p_b^\dagger} + \frac{\partial \pi_a(p_a^*(p_b^\dagger), p_b^\dagger)}{\partial p_b} = 0 \tag{23}$$

[22] Pricing rules that undercut a competitor's price by a specific amount are an example of linear strategies and have been observed in a variety of settings (Chen et al., 2016; Musolff, 2024).

Figure 7: Coercive Strategies with Linear Pricing Rule



Notes: Panel (a) shows prices and Panel (b) shows profits under Bertrand competition with simultaneous pricing, coercion in which the faster firm uses a linear punishment rule and has full commitment ($\beta\gamma \to 1$), and joint profit maximization. In the coercion case, the firms simultaneously set prices at the beginning of the period and then the algorithmic firm has speed advantage α. Assumes $d = 1$ under linear demand given by equation (15).

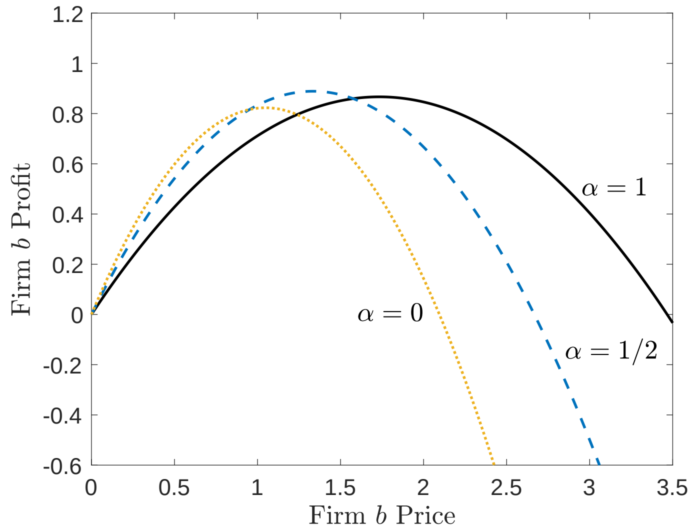
This first-order condition provides the optimal target price, $p_b^\dagger$, that the algorithmic firm chooses for the slower rival. The optimal price for the algorithmic firm is then $p_a^\dagger = p_a^*(p_b^\dagger)$. The solution reflects the fact that the algorithmic firm chooses to coerce the slower rival to set the target price knowing that the rival will maximize profit.

We depict the (long-run) equilibrium for different values of firm b's pricing speed in Figure 7. Panel (a) displays prices for both the algorithmic firm and the slower rival. Prices are consistently higher than Bertrand prices, and they increase with a greater speed advantage α. The algorithmic firm's price is lower than the rival's price when the speed advantage is sufficiently high. These patterns are similar to the model in the previous section assuming the firm uses a trigger strategy (Figure 3). However, given the linear restriction on punishment, the algorithmic firm cannot coerce its slower rival to set prices higher than the collusive price, indicating that the linearity of the strategies does limit the degree to which prices increase.

Panel (b) of Figure 7 displays the profits. Profits for the algorithmic firm are increasing in α, but profits for the slower rival are non-monotonic. With a large enough speed advantage, the algorithmic firm can make higher profits than (its share of) the full collusion profits, even with the linear restriction on the pricing rule. With $\alpha = 1$, the slower rival earns higher profits than in the case with coercive non-linear strategies (Figure 3).

One way to interpret the model is that a high-speed pricing algorithm can effectively modify a naive rival's perceived profit function. A naive rival that only considers the impact of its own price on its profit solves a simple (single-argument) objective function. Given that the

Figure 8: Perceived Profit Function for the Naive Rival



Notes: Figure shows perceived profits for firm b as a function of its own price, under three different values of the algorithmic firm speed advantage, α. $\alpha = 0$ indicates no pricing speed advantage while $\alpha = 1$ indicates fastest pricing. Assumes $d = 1$ under linear demand given by equation (15).

algorithmic firm's pricing rule punishes the rival less as the rival raises price, this encourages the firm to raise price above the Bertrand price. We illustrate this with examples in the following section.

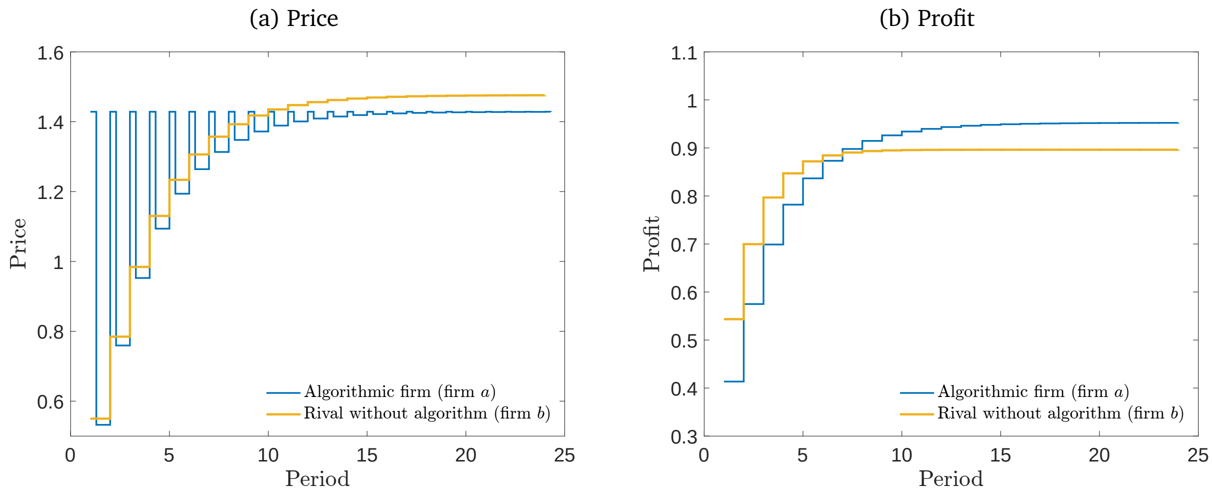
5.3 The Learning Process and Convergence

The linear strategies we analyze provide smooth, concave objective functions for our demand system and lead to rapid convergence for the learning firm.

Proposition 5. *Under linear demand, when firm a implements the optimal linear pricing rule, firm b profits are increasing in price for any $p_b < p_b^\dagger$ and decreasing in price for any $p_b > p_b^\dagger$. Therefore, provided λ is not too large, the use of gradient learning by firm b will result in convergence to the target price chosen by firm a for any initial price pair $(\hat{p}_b^0, \hat{p}_b^1)$.*

Figure 8 plots an example of the perceived profit function for the naive rival under different values of α for the case of linear demand. The dotted yellow line shows the benchmark case with simultaneous pricing ($\alpha = 0$) when the algorithmic firm chooses the Bertrand-Nash equilibrium price. In this case, firm b maximizes profit at $p_b = 1$. When firm a has faster pricing and uses the linear algorithm, firm b has a smooth, single-peaked perceived profit function much like under the simultaneous pricing case; however, the profit function is shifted to the right. Two examples, for $\alpha = 0.5$ and for $\alpha = 1$, are shown in Figure 8. The maximum of this function is obtained at a price of 1.33 when $\alpha = 0.5$. With $\alpha = 1$, firm b maximizes its perceived (and actual) profits at an even higher price, 1.73.

Figure 9: Simulated Learning with Linear Pricing Rule



Notes: Charts show simulated price paths when the naive firm is assumed to use gradient learning. The orange line shows price and profit for firm b, where price is updated in period $t+1$ following $\hat{p}_b^{t+1} = \hat{p}_b^t + \lambda \frac{\pi_b(\hat{p}_b^t, p_a^t) - \pi_b(\hat{p}_b^{t-1}, p_a^{t-1})}{\hat{p}_b^t - \hat{p}_b^{t-1}}$ where the price adjustment step size, λ, is 0.2. The blue line shows price and profit for firm a, which uses a linear pricing rule described in Section 5.2 with pricing speed given by $\alpha = 0.7$. Assumes $d = 1$ under linear demand given by equation (15).

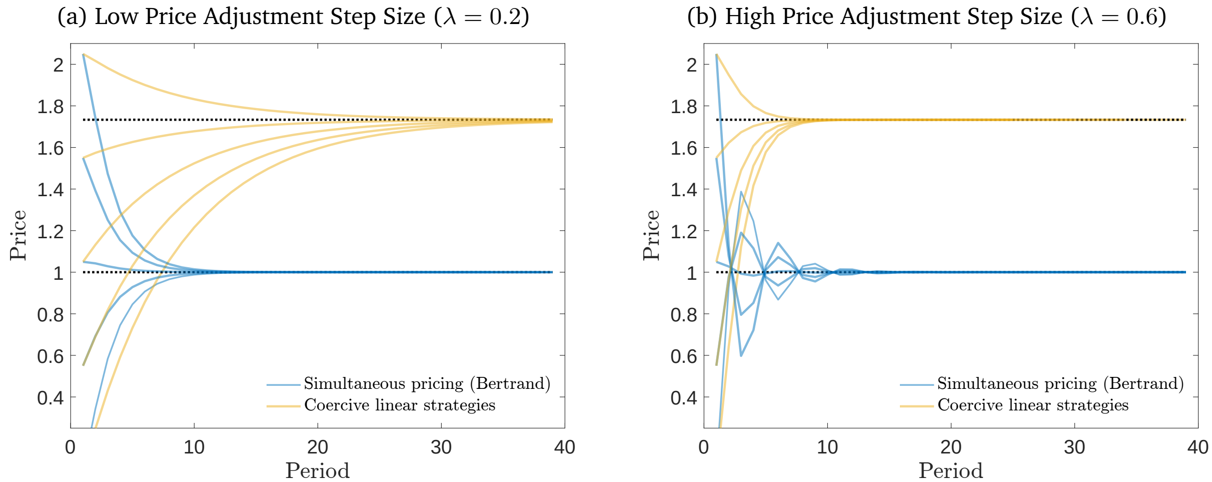
In Figure 9, we simulate an example in which firm a implements the linear pricing rule and has a moderate speed advantage ($\alpha = 0.7$). Firm b is naive and starts by setting an arbitrary price and attempts to maximize profits using gradient learning. We assume firm b approximates the derivative of profits using the most recent two prices and profit realizations. Therefore, after the firm sets price $\hat{p}_b^t$ at time t, the firm's beliefs about the derivative of its profit with respect to price is given by

$$\widehat{\frac{\partial \pi_b}{\partial p_b}}\bigg|_{p_{bt}} = \frac{\pi_{bt}^* - \pi_{b(t-1)}^*}{p_{bt} - p_{b(t-1)}} \tag{24}$$

where π_{bt}^* is the realized profit for period t. Note that firm b need not observe firm a, only the profits it received over the period. Then, the firm updates price according to equation 17. We assign a value of $\lambda = 0.2$.

For this simulation, we assume firm b initially tries a price below the Bertrand price. As shown in Figure 9 Panel (a), firm a initially sets the target price at the start of each period but undercuts firm b after observing the rival's price and applying the optimal long-run linear pricing rule (which we provide in Appendix equation (A-9)). Firm b observes that profits are increasing in its own price and therefore tries higher and higher prices. This is because the algorithmic firm's punishment becomes less severe as firm b tries higher prices. Panel (b) shows that raising prices above the Bertrand price continues to increase firm b's period profits until the equilibrium is reached after about 20 periods. Firm a's price initially oscillates to coerce the rival to set a higher price. However, after prices converge to the equilibrium, firm a's price at

Figure 10: Convergence of Simulated Learning with Linear Pricing Rule versus Simultaneous Pricing



Notes: Charts show simulated price paths with different initial prices when the naive firm is assumed to use gradient learning. Prices are updated in period $t+1$ following $\hat{p}_b^{t+1} = \hat{p}_b^t + \lambda \frac{\pi_b(\hat{p}_b^t, p_a^t) - \pi_b(\hat{p}_b^{t-1}, p_a^{t-1})}{\hat{p}_b^t - \hat{p}_b^{t-1}}$. The blue lines show a simultaneous pricing game that results in Bertrand prices. The orange lines show the game in which the algorithmic firm has maximum pricing speed ($\alpha = 1$) and uses the linear pricing rule described in Section 5.2. Panel (a) shows learning with price adjustment step size of 0.2 and Panel (b) shows learning with a price adjustment step size of 0.6. Assumes $d = 1$ under linear demand given by equation (15).

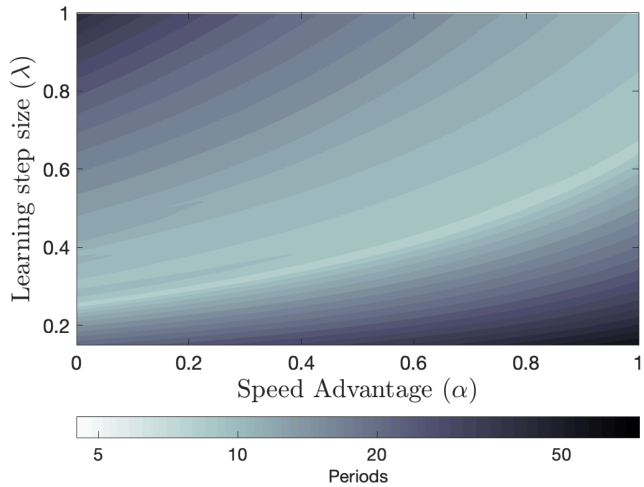
the beginning of the period, $p_a^\dagger$, is the same as the price calculated using the pricing rule, $r_a(p_b^\dagger)$.

Figure 10 examines convergence from multiple starting prices when $\alpha = 1$. First, we consider the standard case of a simultaneous pricing game as a benchmark for comparison. As is well known, prices converge to the Bertrand equilibrium from a variety of starting values in a standard simultaneous pricing game under gradient learning. This is shown by the blue lines. Convergence takes about 15 periods as seen in both panel (a) with a low step size and panel (b) with a higher step size. Under coercive linear strategies, the naive firm's price converges to a supracompetitive level for a variety of initial values, as illustrated by the orange lines in Figure 10. Comparing Panel (a) and Panel (b), convergence is somewhat slower than the standard simultaneous pricing case for a low price adjustment step size but is somewhat faster for a high price adjustment step size.

Regardless of the initial price, convergence occurs quickly, though it does depend on pricing speed and the choice of step size, λ. For each value of α and λ, we simulate the price paths for a grid of 1,000 initial prices. Figure 11 shows the average number of periods to convergence to within 0.1 percent of the target prices for a range of α and λ.[23] Regardless of the value of the learning parameter, the number of periods to convergence is reasonable in our simulations.

[23] To ensure that prices have converged, we identify the first instance for which prices fall within a given tolerance and stay within that tolerance for the next 10 periods in our simulations. Given the nature of gradient learning, this ensures that the price does not drift outside of this tolerance in future periods.

Figure 11: Convergence Speed by Pricing Speed and Price Step Size



Notes: Figure shows the number of periods until convergence for different simulation parameters when a naive firm is assumed to use gradient learning. The x-axis is the speed advantage of the algorithmic firm (α) and the y-axis is the learning step size (λ). For each value of α and λ, we simulate price paths from 1,000 values of initial starting prices from a grid between 0 and 2 and then average the number of periods until convergence. Convergence is defined as being within 0.1% of the equilibrium price. Darker color indicates a larger number of periods until convergence. Simulations assume the algorithmic firm uses the linear pricing rule described in Section 5.2. Assumes $d = 1$ under linear demand given by equation (15).

Moreover, for every value of α, there is a learning parameter λ such that the coercive strategies converge to the target prices in less than 10 periods on average. Higher step sizes tend to converge faster when α is large. Consistent with Proposition 5, the simulations always converge to the coercive equilibrium.[24]

To consider the sensitivity of our results to a static environment, we simulate learning by firm b in the presence of unobserved demand shocks. We modify our demand system to let $D_i(p_i, p_{-i}) = x_t - \left(\frac{1}{4} + \frac{d}{2}\right) p_i + \frac{d}{2} p_{-i}$ where demand shock x_t is distributed uniformly and $E[x_t] = 1$.[25] While demand shocks add randomness to the learning process, prices still converge to the same equilibrium on average. This can be seen in Appendix Figure A-12, which shows simulated prices for different magnitudes of the demand shock affecting the demand intercept. The presence of demand shocks adds error to firm b's estimated gradient, causing firm b to not always adjust prices in the direction that maximizes expected profits. Despite this, average prices across simulations still converge quickly (on average) to the equilibrium derived

[24] Appendix Table A-1 provides statistics on the number of periods to convergence under more and less stringent convergence criteria. When the convergence criterion is within 1 percent of the target price, λ can be chosen such that convergence takes no longer than 9 periods for a wide range of starting values.

Appendix Figure A-11 shows the mean number of periods until convergence for $\alpha = 1$ compared to the simultaneous pricing case for different learning parameters. The fact that convergence in the coercive linear strategy case is faster for large step sizes is due in part to the fact that there is less likely to be price oscillations during the learning process, as seen in Figure 10 Panel (b).

[25] For this example, if firm a observes demand shock x_t, the optimal linear pricing rule remains unchanged; it is the same as the case where $x_t = 1$.

in Section 5.2.

These results highlight that when an algorithmic firm uses the coercive linear strategy we outline above, the coercive equilibrium is robust to a naive firm using gradient learning. Overall, convergence is very fast, making it realistic for firms to implement in practice even in the presence of demand shocks. This can be contrasted with reinforcement learning algorithms, which may be impractical in some settings since convergence can be quite slow.[26]

We consider relatively simple pricing algorithms that are linear in rival's price. Pricing algorithms could in principle take more general forms, allowing greater flexibility in strategically modifying a rival's perceived profit function. For instance, a pricing algorithm that is a nonlinear function of rival's prices could potentially coerce rivals using naive learning strategies to raise prices to even higher levels than those under linear pricing rules. An algorithm that is a nonlinear function of rival's prices could also ensure that a naive firm is maximizing a concave profit function for more general demand systems, ensuring that there is convergence to supracompetitive prices under a variety of naive learning strategies.

6 Extension: Pricing Algorithms on Platforms

Large online platforms, such as online retailers, are often able to determine the technology available to different sellers that compete on their platform. Amazon, for example, allows marketplace retailers to use pricing algorithms provided by Amazon and also algorithms designed by third parties. Even when providing freedom about which pricing rules sellers employ, a large platform can regulate, or control, certain features of the algorithms, such as how often they can update prices and how frequently a firm may switch its pricing rule.

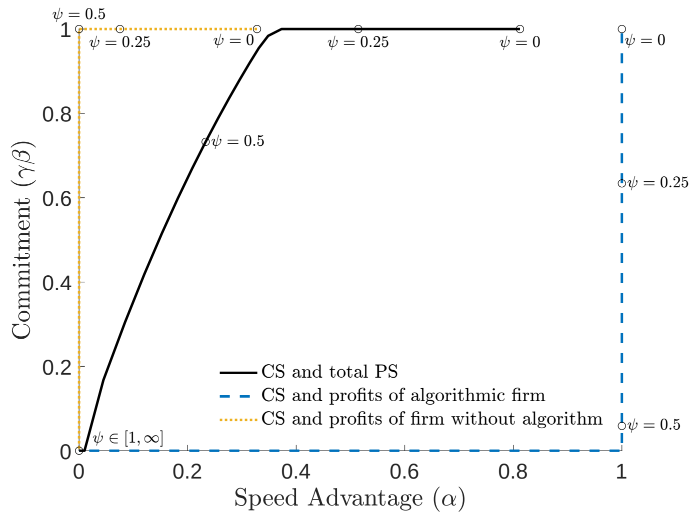
In this section, we consider an environment where a platform can endogenously choose algorithmic technology parameters governing speed and commitment, α and γ, for sellers on the platform. We consider this choice in the context of our baseline model, where seller a has an algorithm that enables faster pricing and commitment and seller b does not. We explore how different objective functions of a platform can lead to different incentives for algorithmic technology adoption.

Our analysis shows that a platform that prioritizes producer surplus has an incentive to allow some sellers to have pricing algorithms with commitment and a speed advantage. By doing so, the platform can soften competition on the platform without resorting to coordinating behavior of the sellers. If a platform is vertically integrated and competes with a seller on the platform, the platform will have an incentive to use faster pricing and commitment to obtain a competitive advantage.

We assume that the platform cares about profits earned by sellers. For simplicity, we assume

[26] For instance, den Boer et al. (2022) find that pricing algorithms based on Q-learning algorithms can take tens of thousands of iterations to converge and argue that these algorithms are "intrinsically slow."

Figure 12: Optimal Technology Parameters by Platform Objective Function



Notes: Shows the optimal speed advantage (α) and commitment ($\beta\gamma$) that solve different objective functions for a platform where ψ is the weight on consumer surplus. The black line shows the optimal algorithm characteristics given a platform maximizing a weighted sum of total producer surplus and consumer surplus where points on the line represent different weights. The dashed blue line shows the optimal algorithm characteristics given a platform maximizing a weighted sum of firm a's profits and consumer surplus. The dotted orange line shows the optimal algorithm characteristics given a platform maximizing a weighted sum of firm b's profits and consumer surplus. Assumes $d = 1$ under linear demand given by equation (15).

the platform earns non-distortionary commissions from these profits. The platform may also have dynamic incentives to retain consumers. We represent this in its objective function by placing some weight on consumer surplus.[27] We assume the platform commits to technology parameters for an indefinite period. Due to the stationarity of the model, we can represent the platform's surplus as being proportional to current profits and weighted consumer surplus, yielding the following objective:

$$\max_{\alpha,\gamma} \pi^*(\alpha, \gamma) + \psi CS^*(\alpha, \gamma). \tag{25}$$

Here, $\pi^*(\alpha, \gamma)$ represents per-period equilibrium profits and $CS^*(\alpha, \gamma)$ represents per-period equilibrium consumer surplus given algorithm speed and commitment α and γ.[28] The weight on consumer surplus is given by ψ. Though stylized, this objective function allows us to capture the key tradeoff a platform faces in terms of the surplus of different platform participants.

We initially assume $\pi^*(\alpha, \gamma)$ represents the joint profits of both sellers a and b. For each possible value of ψ, we solve for equilibrium prices, profits, and consumer surplus. We then solve for the pair (α, γ) that maximizes the platform's objective using the linear demand given by equation (15) and a value of $d = 1$.

[27] This can be microfounded by assuming, for example, that consumers have switching costs and may stop using the platform if not given enough surplus. See, e.g., Gutierrez (2022).

[28] Without loss of generality, we let $\beta \to 1$ and consider only variation in γ.

Figure 12 plots the optimal technology parameters as a function of ψ. The black line shows the optimal speed and commitment when $\pi^*(\alpha, \gamma)$ represents total producer surplus. For $\psi \in [1, \infty]$, the optimal technology parameters yield $\alpha = 0$ and $\gamma = 0$, generating the standard Bertrand-Nash equilibrium with no speed advantage and no commitment. Consistent with standard intuition in these models, the baseline "competitive" environment maximizes total surplus ($\psi = 1$).

When the platform puts less weight on consumer surplus, the platform chooses technology parameters that provide firm a with a speed advantage ($\alpha > 0$) and some multi-period commitment ($\gamma > 0$). When consumer surplus is valued half as much as profits ($\psi = 0.5$), the platform chooses $(\alpha, \gamma) = (0.23, 0.71)$. That is, it allows partial commitment and a modest speed advantage for firm a. When $\psi = 0.25$, the platform chooses full commitment ($\gamma = 1$) and α slightly higher than 0.5.

As the relative weight on profits increases, the platform increases α to give firm a an even greater speed advantage. When maximizing producer surplus only, the platform chooses a value of α close to 0.8. This yields the outcome with the greatest joint profits.

One can also consider a platform that prioritizes profits for only one seller. This could be the case if the platform is partially vertically integrated and sells a product that competes with products sold by an independent seller. Alternatively, the firm could have contractual arrangements that prioritize one seller. The case in which $\pi^*(\alpha, \gamma)$ includes only the profits of firm a is depicted by the dashed blue line. In contrast to the case where $\pi^*(\alpha, \gamma)$ is total producer surplus, the platform is incentivized to prioritize a speed advantage for firm a. This is illustrated by the fact that the dashed blue line lies to the left of the black line. As before, any value of $\psi \in [1, \infty]$ leads to simultaneous pricing. However, a consumer surplus weight of 0.5 yields $\alpha = 1$ and $\gamma \approx 0.05$ as the optimal choice. As ψ decreases, the platform would continue to increase the value of commitment. When $\psi = 0$ and the platform is tasked with maximizing firm a's profits only, the platform would choose the parameters that yield maximal coercion. The difference between the black line and the dashed blue line in Figure 12 indicates that a speed advantage can generate increased profits for firm a at the expense of total producer surplus.

Finally, one can consider the case in which the platform prioritizes firm b profits ($\pi^*(\alpha, \gamma)$ represents the profits of firm b only). This is depicted by the dotted orange line. In this case, it is optimal for the platform to prioritize commitment rather than a speed advantage. Even with $\psi = 0$, the platform would only allow a modest speed advantage with its pricing algorithms. A greater speed advantage would reduce the profits of firm b (but yield greater total profits).

This extension illustrates how different welfare weights can lead to different values for endogenously chosen technology parameters under our equilibrium concept. A social planner that puts equal weight on the profits of the two firms and consumers would choose simultaneous pricing—i.e., not allow algorithms that provide a speed advantage or commitment. In contrast, a platform placing more weight on producer surplus than consumer surplus has an incentive to enable pricing algorithms on the platform for a subset of sellers. By providing some firms with commitment and a speed advantage, the platform can generate coercive equilibria, softening competition among sellers even when they behave non-cooperatively. This is especially likely to be the case for a platform with significant market power that places little weight on consumer surplus. However, a platform aiming to maximize joint profits of sellers does not have an incentive to allow sellers to engage in maximal coercion ($\alpha = 1$ and $\gamma = 1$).

Another implication is that, if the platform sells its own products on the platform, it will have an incentive to give itself the pricing algorithm. In this case, the platform will give itself maximum pricing speed and commitment to gain a competitive advantage unless it puts significant weight on consumer surplus. To the extent that the platform cares about consumer surplus, the platform may only give itself a speed advantage and limited ability to commit. Finally, if the platform's objective were more strongly tied to the slow firm, it would prioritize commitment instead.

7 Conclusion

This paper examines how pricing algorithms that combine speed within a period and commitment across periods can fundamentally alter competitive outcomes. We characterize a coercive equilibrium in which a fast, algorithmic firm unilaterally induces its rival to set supracompetitive prices, even when that rival is short-sighted and cannot sustain collusion.

We highlight an important interaction between pricing speed and commitment. A firm with faster pricing than a rival can threaten to quickly undercut a rival's price unless it sets a high price. Commitment across periods makes this strategy optimal from the perspective of the algorithmic firm. An algorithm that enables both a speed advantage and commitment combines the ability to punish like a follower and the ability to commit to an initial high price like a leader. In this way, an algorithmic firm can extract a disproportionate share of industry profits. In some cases, coercion can result in prices that are "supracollusive", leading to worse outcomes for consumers and lower overall welfare than collusion. Our analysis of pricing rules that are a linear function of the slower rival's price demonstrates that such strategies can lead to supracompetitive prices while potentially raising less antitrust scrutiny and being robust to the use of simple learning strategies by the slower firm. We argue that the equilibria explored in this paper are more robust than standard collusive equilibria, which require all firms to be forward looking and understand the dynamic strategies of rivals. Coercive equilibrium can arise even when rivals are naive—they need not understand the nature of the game.

One implication of the model is that, consistent with claims by pricing algorithm providers, firms always benefit from reacting faster to their competitors' prices. While we take differences in pricing speed across competitors as given, there could be an "arms race" in which firms compete to price faster than rivals. In other settings, such as high-frequency trading, it has been noted that such an arms race to invest in technologies that allow for faster responses can be inefficient (Budish et al., 2015).

Overall, our results suggest a broad scope for firms to strategically increase prices using high-speed pricing algorithms. Algorithmic firms may be able to manipulate their rivals into setting prices above the competitive levels even when characteristics of the market would rule out traditional collusive strategies, such as short-termism or naive learning. There is an opportunity for future research to examine the extent to which pricing strategies and algorithms used in practice may raise prices based on the features we identify here.

References

Abreu, D., P. Milgrom, and D. Pearce (1991). Information and timing in repeated partnerships. *Econometrica*, 1713–1733.

Anufriev, M., D. Kopányi, and J. Tuinstra (2013). Learning cycles in bertrand competition with differentiated commodities and competing learning rules. *Journal of Economic Dynamics and Control 37*(12), 2562–2581.

Aparicio, D., Z. Metzman, and R. Rigobon (2021). The pricing strategies of online grocery retailers. Technical report, National Bureau of Economic Research Working Paper.

Asker, J., C. Fershtman, and A. Pakes (2024). The impact of artificial intelligence design on pricing. *Journal of Economics and Management Strategy 33*(2), 276–304.

Assad, S., R. Clark, D. Ershov, and L. Xu (2024). Algorithmic pricing and competition: Empirical evidence from the german retail gasoline market. *Journal of Political Economy 132*(3), 723–771.

Banchio, M. and G. Mantegazza (2022). Artificial intelligence and spontaneous collusion. *arXiv preprint arXiv:2202.05946*.

Brown, Z. Y. and A. MacKay (2023). Competition in pricing algorithms. *American Economic Journal: Microeconomics 15*(2), 109–156.

Budish, E., P. Cramton, and J. Shim (2015). The high-frequency trading arms race: Frequent batch auctions as a market design response. *The Quarterly Journal of Economics 130*(4), 1547–1621.

Byrne, D. P., N. de Roos, M. S. Lewis, L. M. Marx, and X. A. Wu (2025). Asymmetric information sharing in oligopoly: A natural experiment in retail gasoline. *Forthcoming, Journal of Political Economy*.

Calder-Wang, S. and G. H. Kim (2023). Coordinated vs efficient prices: The impact of algorithmic pricing on multifamily rental markets. *Available at SSRN 4403058*.

Calvano, E., G. Calzolari, V. Denicolo, and S. Pastorello (2020). Artificial Intelligence, Algorithmic Pricing, and Collusion. *American Economic Review 110*(10), 3267–97.

Chen, L., A. Mislove, and C. Wilson (2016). An Empirical Analysis of Algorithmic Pricing on Amazon Marketplace. In *Proceedings of the 25th International Conference on World Wide Web*, pp. 1339–1349.

den Boer, A. V., J. M. Meylahn, and M. P. Schinkel (2022). Artificial collusion: Examining supra-competitive pricing by q-learning algorithms. *Amsterdam Law School Research Paper* (2022-25).

Fudenberg, D., D. M. Kreps, and E. S. Maskin (1990). Repeated Games with Long-Run and Short-Run Players. *The Review of Economic Studies 57*(4), 555.

Fudenberg, D. and D. K. Levine (1989). Reputation and equilibrium selection in games with a patient player. *Econometrica 57*(4), 759–778.

Fudenberg, D. and D. K. Levine (2016). Whither game theory? towards a theory of learning in games. *Journal of Economic Perspectives 30*(4), 151–170.

Fudenberg, D. and J. Tirole (1986). A "signal-jamming" theory of predation. *The RAND Journal of Economics*, 366–376.

Gutierrez, G. (2022). The welfare consequences of regulating amazon. Working paper.

Hansen, K. T., K. Misra, and M. M. Pai (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 & Economics 14*(3), 331–363.

Harrington, J. E. (2022). The effect of outsourcing pricing algorithms on market competition. *Management Science 68*(9), 6889–6906.

Johnson, J. P., A. Rhodes, and M. Wildenbeest (2023). Platform design when sellers use pricing algorithms. *Econometrica 91*(5), 1841–1879.

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

Kreps, D. M. and R. Wilson (1982). Reputation and imperfect information. *Journal of Economic Theory 27*(2), 253–279.

Lamba, R. and S. Zhuk (2025). Pricing with algorithms. *arXiv preprint arXiv:2205.04661*.

Leisten, M. (2024). Algorithmic competition, with humans. Working paper.

Levine, D. K. (2024). Efficiently breaking the folk theorem by reliably communicating long term commitments. *Available at SSRN*. Working paper.

Maskin, E. and J. Tirole (1988). A Theory of Dynamic Oligopoly, I: Overview and Quantity Competition with Large Fixed Costs. *Econometrica 56*(3), 549–569.

Miklós-Thal, J. and C. Tucker (2019). Collusion by Algorithm: Does Better Demand Prediction Facilitate Coordination Between Sellers? *Management Science 65*(4), 1552–1561.

Milgrom, P. and J. Roberts (1982). Predation, reputation, and entry deterrence. *Journal of Economic Theory 27*(2), 280–312.

Musolff, L. (2024). Algorithmic pricing facilitates tacit collusion. Working paper.

O'Connor, J. and N. E. Wilson (2021). Reduced Demand Uncertainty and the Sustainability of Collusion: How AI Could Affect Competition. *Information Economics and Policy 54*.

Porter, R. H. (2005). Detecting collusion. *Review of Industrial Organization 26*(2), 147–167.

Salcedo, B. (2015). Pricing Algorithms and Tacit Collusion. Working paper.

Salop, S. (1986). Practices that (Credibly) Facilitate Oligopoly Coordination. In J. Stiglitz and F. Mathewson (Eds.), *New Developments in the Analysis of Market Structure*. MIT Press, Cambridge, MA.

Scherer, F. M. (1980). *Industrial Market Structure and Economic Performance: 2d Ed*. Rand Mcnally College.

Singh, N. and X. Vives (1984). Price and Quantity Competition in a Differentiated Duopoly. *The RAND Journal of Economics 15*(4), 546–554.

Tirole, J. (1988). *The Theory of Industrial Organization*. MIT Press, Cambridge, MA.

Waltman, L. and U. Kaymak (2008). Q-learning agents in a cournot oligopoly model. *Journal of Economic Dynamics and Control 32*(10), 3275–3293.

Appendix

A.1 Proofs

Proof of Proposition 1

In equilibrium, firm b chooses $p_{bt}^* = R_b(\rho)$, where $R_b(\rho)$ is firm b's best response. Thus, firm a's problem with commitment and no speed advantage can be expressed as

$$\max_{\rho | p_{bt}} \left[\pi_a(\rho, p_{bt}) + \frac{\beta\gamma}{1-\beta\gamma} \pi_a(\rho, R_b(\rho)) \right] \tag{A-1}$$

Firm a's problem in a simultaneous price-setting game and in a sequential price-setting game (where firm a is the leader) are given by $\max_\rho \pi_a(\rho, p_{bt})$ and $\max_\rho \pi_a(\rho, R_b(\rho))$, respectively. In (A-1), firm a maximizes a weighted sum of these two objective functions. Given that π_a is quasi-concave, firm a's equilibrium price lies between the ρ that solves the simultaneous price-setting game and the ρ that solves the sequential price-setting game where firm a is the leader. $\square$

Proof of Proposition 2

Denote the joint profit-maximizing (collusive) prices (p_a^C, p_b^C) and the sequential-move prices where firm b is the leader as (p_a^S, p_b^S), i.e.,

$$p_b^S = \arg\max_{p_b} \pi_b(p_b, R_a(p_b)) \tag{A-2}$$

$$p_a^S = R_a(p_b^S) \tag{A-3}$$

and $R_a(p_b)$ provides firm a's static best-response function. The collusive prices satisfy $(p_a^C, p_b^C) = \arg\max_{(p_a,p_b)} [\pi_a(p_a, p_b) + \pi_b(p_b, p_a)]$ and yield the first-order conditions $\frac{\partial \pi_a}{\partial p_a} + \frac{\partial \pi_b}{\partial p_a} = 0$, $\frac{\partial \pi_b}{\partial p_b} + \frac{\partial \pi_a}{\partial p_b} = 0$.

It is sufficient to consider the maximal coercion case ($\alpha = 1$, $\beta\gamma = 1$) where the algorithmic firm's objective is:

$$\max_{(p_a^\dagger, p_b^\dagger)} \pi_a(p_a^\dagger, p_b^\dagger) \tag{A-4}$$

$$\text{s.t.} \quad \pi_b(p_b^\dagger, p_a^\dagger) \geq \max_{p_b} \pi_b(p_b, R_a(p_b))$$

The additional condition used in the proposition is that $\pi_b(p_b^C, p_a^C) > \max_{p_b} \pi_b(p_b, R_a(p_b)) = \pi_b(p_b^S, p_a^S)$. This establishes that a candidate target price vector $(p_a^\dagger, p_b^\dagger) = (p_a^C, p_b^C)$ satisfies the incentive compatibility constraint under maximal coercion. Moreover, the constraint is slack because the inequality is strict.

Therefore, firm a can profitably deviate by either decreasing $p_a^\dagger$ or increasing $p_b^\dagger$, since:

$$\frac{\partial \pi_a(p_a, p_b^C)}{\partial p_a} < 0 \quad \text{at } p_a = p_a^C \tag{A-5}$$

$$\frac{\partial \pi_a(p_a^C, p_b)}{\partial p_b} > 0 \quad \text{at } p_b = p_b^C \tag{A-6}$$

The second inequality holds because products are substitutes, and the first inequality follows from the sign of the second inequality and the first-order condition for joint profit maximization. Thus, firm a can choose target prices to increase its profits beyond $\pi_a(p_a^C, p_b^C)$ without violating firm b's incentive compatibility constraint.

We have shown that a profitable deviation exists for $\alpha = 1$ and $\beta\gamma = 1$. Because Propositions 3 and 4 establish that profits are increasing in the speed advantage and commitment, there exist threshold values $\overline{\alpha}$ and $\overline{\beta\gamma}$ such that, for $\alpha > \overline{\alpha}$ and $\beta\gamma > \overline{\beta\gamma}$, the algorithmic firm earns profits greater than its share of profits under joint profit maximization. □

Proof of Proposition 3

We show that this holds for any $\alpha > 0$, including α arbitrarily close to zero. We establish that, in equilibrium, the target price $p_b^\dagger$ is greater than the optimal price firm b would choose when holding fixed firm a's price at $p_a^\dagger$. We proceed by contradiction. Denote this latter price as $\tilde{p}_b = \arg\max_{p_b | p_a^\dagger} \pi_b(p_b, p_a^\dagger)$. If $\tilde{p}_b > p_b^\dagger$, then, by the fact that the goods are substitutes, $\pi_a(p_a^\dagger, \tilde{p}_b) > \pi_a(p_a^\dagger, p_b^\dagger)$. This contradicts the fact that $(p_a^\dagger, p_b^\dagger)$ is a solution to firm a's problem, because the alternative target price vector $(p_a^\dagger, \tilde{p}_b)$ would provide greater profits for firm a while still satisfying firm b's incentive compatibility condition. We also have that $\tilde{p}_b \neq p_b^\dagger$ because $\tilde{p}_b = p_b^\dagger$ only when $p_b^\dagger$ lies on firm b's best-response function, and with $\alpha > 0$ this is not the case. Therefore, $\tilde{p}_b < p_b^\dagger$.

We can express the algorithmic firm's constrained optimization problem as the Lagrangian

$$\mathcal{L} = \pi_a(p_a^\dagger, p_{bt}) + \frac{\beta\gamma}{1-\beta\gamma}\pi_a(p_a^\dagger, p_b^\dagger) + \lambda\left[\pi_b(p_b^\dagger, p_a^\dagger) - \left[(1-\alpha)\pi_b(\hat{p}_b, p_a^\dagger) + \alpha\pi_b(\hat{p}_b, R_a(\hat{p}_b))\right]\right] \tag{A-7}$$

for $\hat{p}_b$ that maximizes firm b's deviation profits.[29] The corresponding optimization conditions

[29] Technically, the incentive compatibility constraint holds for any deviation price by firm b (not just a particular $\hat{p}_b$), but these constraints are non-binding for other choices of p_b and therefore the corresponding Lagrangian multipliers are zero. $\hat{p}_b$ reflects the optimally chosen deviation.

are:

$$
\begin{aligned}
[p_a^\dagger] &: 0 = \pi_a^1(p_a^\dagger, p_{bt}) + \frac{\beta\gamma}{1-\beta\gamma}\pi_a^1(p_a^\dagger, p_b^\dagger) + \lambda\left[\pi_b^2(p_b^\dagger, p_a^\dagger) - (1-\alpha)\pi_b^2(\hat{p}_b, p_a^\dagger)\right] \\
[p_b^\dagger] &: 0 = \frac{\beta\gamma}{1-\beta\gamma}\pi_a^2(p_a^\dagger, p_b^\dagger) + \lambda\left[\pi_b^1(p_b^\dagger, p_a^\dagger)\right] \\
[\lambda] &: \pi_b(p_b^\dagger, p_a^\dagger) - \left[(1-\alpha)\pi_b(\hat{p}_b, p_a^\dagger) + \alpha\pi_b(\hat{p}_b, R_a(\hat{p}_b))\right]
\end{aligned}
$$

where superscripts indicate derivatives with respect to the first or second argument.

From the second optimization condition, we solve for the Lagrange multiplier:

$$
\lambda = -\frac{\beta\gamma}{1-\beta\gamma}\frac{\pi_a^2(p_a^\dagger, p_b^\dagger)}{\pi_b^1(p_b^\dagger, p_a^\dagger)}
$$

By the assumption that the products are substitutes, $\pi_a^2(p_a^\dagger, p_b^\dagger) > 0$. Conversely, because $\tilde{p}_b < p_b^\dagger$, $\pi_b^1(p_b^\dagger, p_a^\dagger) < 0$. Therefore, $\lambda > 0$.

To determine the effect of α on the algorithmic firm's profits, we take the derivative of the Lagrangian with respect to α. This yields

$$
\begin{aligned}
\frac{d\mathcal{L}}{d\alpha} =& \pi_a^1(p_a^\dagger, p_{bt})\frac{\partial p_a^\dagger}{\partial\alpha} + \frac{\beta\gamma}{1-\beta\gamma}\pi_a^1(p_a^\dagger, p_b^\dagger)\frac{\partial p_a^\dagger}{\partial\alpha} + \frac{\beta\gamma}{1-\beta\gamma}\pi_a^2(p_a^\dagger, p_b^\dagger)\frac{\partial p_b^\dagger}{\partial\alpha} \\
&+ \lambda\left[\pi_b^1(p_b^\dagger, p_a^\dagger)\frac{\partial p_b^\dagger}{\partial\alpha} + \pi_b^2(p_b^\dagger, p_a^\dagger)\frac{\partial p_a^\dagger}{\partial\alpha} + \left[\pi_b(\hat{p}_b, p_a^\dagger) - \pi_b(\hat{p}_b, R_a(\hat{p}_b))\right]\right] \\
&- \lambda\left[(1-\alpha)\pi_b^2(\hat{p}_b, p_a^\dagger)\frac{\partial p_a^\dagger}{\partial\alpha}\right]
\end{aligned}
$$

where we have invoked the envelope theorem with respect to $\hat{p}_b$.

Plugging in the optimization conditions for $p_a^\dagger$ and $p_b^\dagger$, we obtain:

$$
\frac{d\mathcal{L}}{d\alpha} = \lambda\left[\pi_b(\hat{p}_b, p_a^\dagger) - \pi_b(\hat{p}_b, R_a(\hat{p}_b))\right]
$$

Note that $p_a^\dagger > R_a(\hat{p}_b)$ because the products are strategic complements in prices and it must be that $\hat{p}_b < p_b^\dagger$, following logic parallel to that at the beginning of this proof (that showed $\tilde{p}_b < p_b^\dagger$). Then, because the products are substitutes, $\pi_b(\hat{p}_b, p_a^\dagger) - \pi_b(\hat{p}_b, R_a(\hat{p}_b)) > 0$. Since $\lambda > 0$, we obtain $\frac{d\mathcal{L}}{d\alpha} > 0$. Therefore, profits are increasing in the speed advantage. □

Proof of Proposition 4

We begin with the Lagrangian introduced in the previous proof. Taking the derivative with respect to $\beta\gamma$, while invoking the envelope theorem for $p_a^\dagger$, $p_b^\dagger$, and $\hat{p}_b$, we obtain

$$\frac{\partial \mathcal{L}}{\partial \beta\gamma} = \frac{1}{1-\beta\gamma}\pi_a(p_a^\dagger, p_b^\dagger) + \frac{\beta\gamma}{(1-\beta\gamma)^2}\pi_a(p_a^\dagger, p_b^\dagger) = \frac{1}{(1-\beta\gamma)^2}\pi_a(p_a^\dagger, p_b^\dagger) \tag{A-8}$$

Because profits are positive, $\frac{\partial \mathcal{L}}{\partial \beta\gamma} > 0$. Therefore, profits are increasing in commitment. □

Proof of Proposition 5

For the purposes of the proof, we assume that firm b knows the derivative of its profits at the chosen price, i.e., its estimate $\left.\widehat{\frac{\partial \pi_b}{\partial p_b}}\right|_{p_{bt}}$ is accurate. For the estimation rule specified in equation (24), we will also require that the two initial prices $\hat{p}_b^0$ and $\hat{p}_b^1$ yield different profits and that λ is sufficiently small.

First, we solve explicitly for equilibrium target prices

$$p_a^\dagger = \frac{6d+2}{2(1-\alpha)d^2+4d+1}$$

$$p_b^\dagger = \frac{2+2(\alpha+5)d(d+1)}{(2d+1)(2d((1-\alpha)d+2)+1)}.$$

This yields the pricing rule

$$\sigma(p_b) = \frac{(2d+1)(3d+1)}{(\alpha+5)d(d+1)-1}p_b. \tag{A-9}$$

The slow firm chooses price p_b. The slow firm's profit function is given by $\tilde{\pi}_b(p_b) = (1-\alpha)\pi_b(p_b, p_a^\dagger) + \alpha\pi_b(p_b, \sigma(p_b))$. We solve for it explicitly:

$$\tilde{\pi}_b(p_b) = \frac{p_b(d(\alpha-5(1-\alpha)d-5)-1)(2\alpha d(d(2dp_b+p_b+2)+2)-2d(d+1)((2d+3)p_b-10)-p_b+4)}{4(2d((\alpha-1)d-2)-1)((\alpha+5)d(d+1)+1)}$$

For this function, it is the case that $\frac{\partial \tilde{\pi}_b(p_b)}{\partial p_b} > 0$ when $p_b > p_b^\dagger$ and $\frac{\partial \tilde{\pi}_b(p_b)}{\partial p_b} < 0$ when $p_b < p_b^\dagger$. Thus, gradient learning converges to the optimum and will yield the target prices. □

A.2 Extension to Oligopoly

While we primarily consider the case where an algorithmic firm faces a single slower rival, it is straightforward to extend the results to a more general setting in which a single algorithmic firm faces multiple slower rivals that all have the same pricing frequency. Thus, the speed advantage α characterizes the reaction time that the faster firm has relative to each of the slower firms.

We consider an n firm oligopoly with one algorithmic firm and $n-1$ slower rivals. The algorithmic firm's price is given by p_a while the vector of prices for slower rivals is given by $\mathbf{r} = (r_1, r_2, \ldots, r_{n-1})$. Generalizing the model in Section 4, firm a chooses the target price for itself, $p_a^\dagger$, and rivals, $\mathbf{r}^\dagger = (r_1^\dagger, r_2^\dagger, \ldots, r_{n-1}^\dagger)$, that maximize its discounted profits, subject to the algorithm technology and the incentive compatibility constraint for the rival firms:

$$\max_{(p_a^\dagger, \mathbf{r}^\dagger)|\mathbf{r}_t} (1-\alpha)\pi_a(\rho, \mathbf{r}_t) + \alpha\pi_a(\sigma(\mathbf{r}_t), \mathbf{r}_t) + \frac{\beta\gamma}{1-\beta\gamma}\pi_a(p_a^\dagger, \mathbf{r}^\dagger) \tag{A-10}$$

$$\text{s.t.} \quad (i) \quad \rho = p_a^\dagger \tag{A-11}$$

$$(ii) \quad \sigma(\mathbf{r}) = \begin{cases} p_a^\dagger & \text{if } \mathbf{r} = \mathbf{r}^\dagger \\ R_a(\mathbf{r}) & \text{if } \mathbf{r} \neq \mathbf{r}^\dagger \end{cases} \tag{A-12}$$

$$(iii) \quad \pi_j(p_a^\dagger, \mathbf{r}^\dagger) \geq (1-\alpha)\pi_j(\rho, \hat{r}_j, \mathbf{r}_{-j}^\dagger) + \alpha\pi_j(\sigma(\hat{r}_j, \mathbf{r}_{-j}^\dagger), \hat{r}_j, \mathbf{r}_{-j}^\dagger) \quad \forall \hat{r}_j, \forall j \tag{A-13}$$

After a fraction $1-\alpha$ of period t elapses, the algorithmic firm observes the vector of prices $\mathbf{r_t}$ and maintains price $p_{at}^\dagger$ if all prices are equal to the target prices for rivals. Otherwise, the algorithmic firm best responds to rivals, setting price $R_a(\mathbf{r})$.

We examine the case of linear demand that generalizes the duopoly demand given by equation (15). Demand for the algorithmic firm is given by

$$D_a(p_a, \mathbf{r}) = 1 - \left(\frac{1}{4} + \frac{d}{2}\right) p_a + \sum_{j=1}^{n-1} \frac{d}{2(n-1)} r_j \tag{A-14}$$

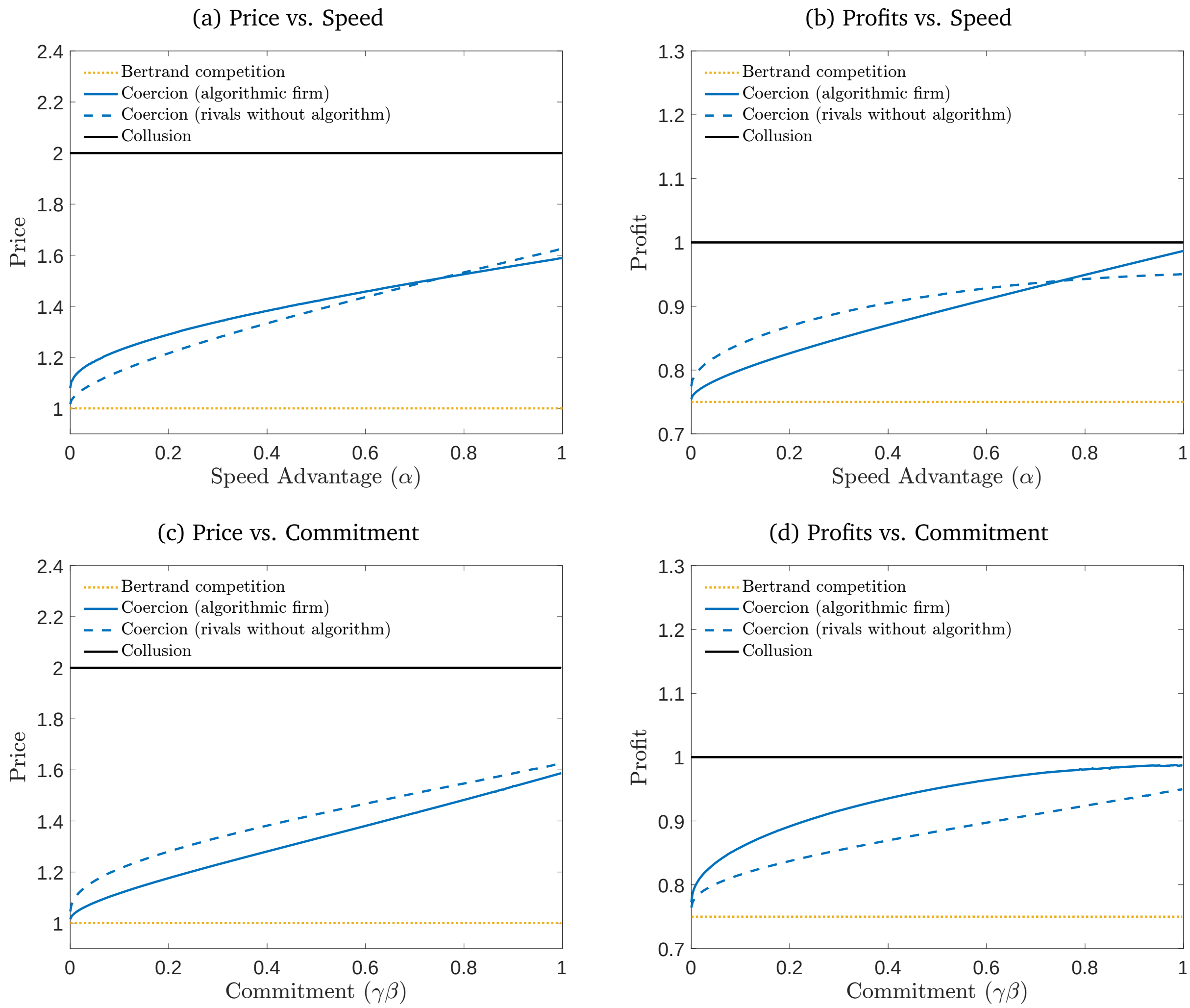
Demand is symmetric. Therefore, demand for rival j is given by

$$D_j(p_a, \mathbf{r}) = 1 - \left(\frac{1}{4} + \frac{d}{2}\right) r_j + \frac{d}{2(n-1)} p_a + \sum_{k \neq j} \frac{d}{2(n-1)} r_k \tag{A-15}$$

When $d = 1$, the Bertrand price and joint profit maximization prices are the same as for the duopoly linear demand given by equation (15).

We simulate equilibrium outcomes for the $n = 3$ case with $d = 1$. Appendix Figure A-1 shows equilibrium prices (panel (a)) and profits (panel (b)) with full commitment and different values for the speed advantage. The chart is qualitatively similar to the case of two firms shown in Figure 3. As in our previous analysis, prices are increasing in the speed advantage of firm a.

Figure A-1: Prices and Profits in Coercive Equilibrium with Three Firms

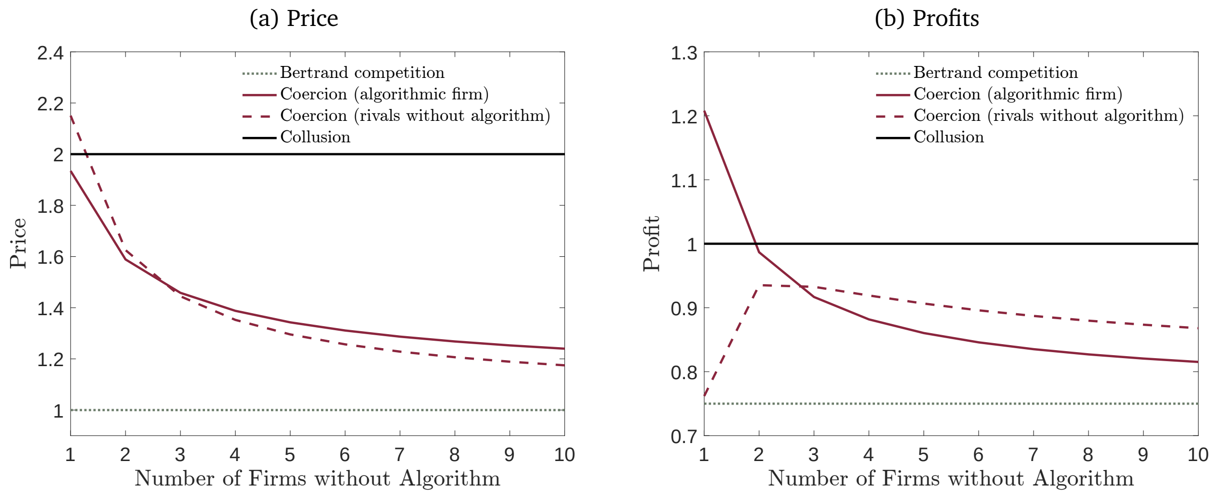


Notes: Panels (a) and (b) show equilibrium prices and profits for different values of speed advantage α under coercion with full commitment ($\beta\gamma \to 1$). Panels (c) and (d) show equilibrium prices and profits under coercion for different values of commitment, $\beta\gamma$, when the faster firm has maximum speed ($\alpha = 1$). Outcomes under Bertrand competition with simultaneous pricing and joint profit maximization are displayed for comparison. Assumes $d = 1$ under linear demand given by equation (A-14).

Profits of firm a are also increasing in the speed advantage. However, the ability of firm a to coerce rivals into setting higher prices is somewhat muted compared to the two-firm case due to the fact that the incentive compatibility constraints are more binding in the three-firm case.

Panels (c) and (d) of Appendix Figure A-1 show equilibrium prices and profits in the case with maximum speed advantage and different values for the degree of commitment. Again, the results are similar to Figure 3. Prices and profits are increasing in the degree of commitment; however, prices under full commitment are lower than in Figure 3 given the additional constraints on firm a. In the two-firm case, the incentive compatibility constraint for the firm

Figure A-2: Prices and Profits in Maximum Coercive Equilibrium under Oligopoly



Notes: Panel (a) shows prices and panel (b) shows profits under Bertrand competition with simultaneous pricing, the coercive equilibrium with maximum speed ($\alpha = 1$) and maximum commitment ($\beta\gamma \to 1$), and joint profit maximization. Assumes $d = 1$ under linear demand given by equation (A-14).

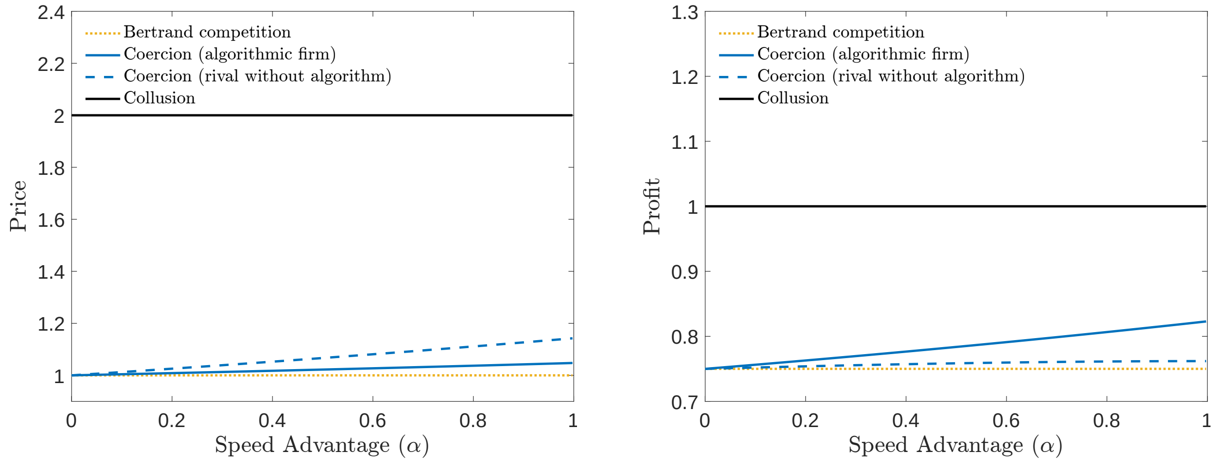
without the pricing algorithm is determined by $(1-\alpha)\pi_b(\hat{p}_b, \rho) + \alpha\pi_b(\hat{p}_b, \sigma(\hat{p}_b))$, which does not depend on the degree of commitment. However, in the three-firm case, the equivalent expression, $\pi_b(p_a^\dagger, r_b^\dagger, r_c^\dagger) \geq (1-\alpha)\pi_b(\rho, \hat{r}_b, r_c^\dagger) + \alpha\pi_b(\sigma(\hat{r}_b, r_c^\dagger), \hat{r}_b, r_c^\dagger)$, does depend on the degree of commitment since commitment affects the other slow firm's price, $r_c^\dagger$.

Finally, we examine the maximal coercion equilibrium by the number of firms in Appendix Figure A-2. Given the demand system, the Bertrand price and profits under joint profit maximization are constant as the number of firms increases. Under the coercive equilibrium, prices are decreasing as the algorithmic firm faces additional rivals without an algorithm. The algorithmic firm's profits are also decreasing in the number of rivals. However, the effect of additional rivals without an algorithm is relatively modest. Prices and profits are still substantially higher relative to the Bertrand equilibrium even with several rivals. With 10 rival firms, market prices are on average 18 percent higher than the Bertrand equilibrium.

A.3 Additional Tables and Figures

Figure A-3: Prices and Profits in Coercive Equilibrium: No Commitment

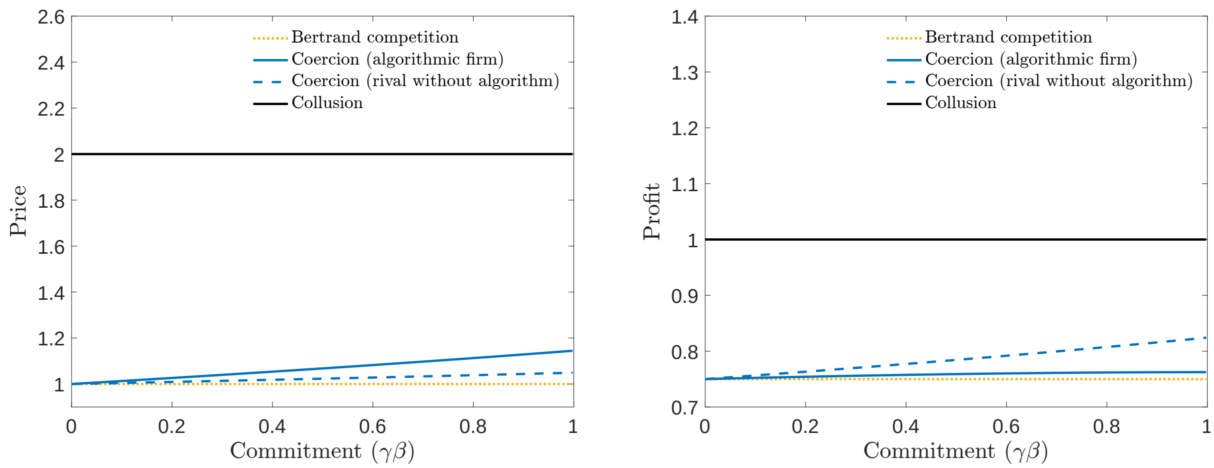
(a) Price (b) Profits



Notes: Panel (a) shows prices and panel (b) shows profits under Bertrand competition with simultaneous pricing, the coercive equilibrium with no commitment ($\beta\gamma = 0$) and speed advantage α, and joint profit maximization. Assumes $d = 1$ under linear demand given by equation (15).

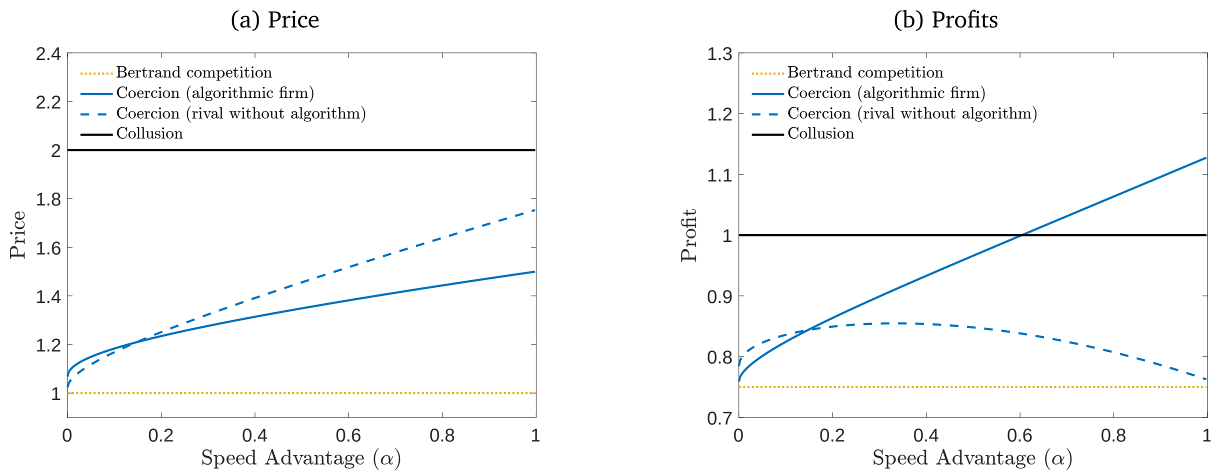
Figure A-4: Prices and Profits in Coercive Equilibrium: No Speed Advantage

(a) Price (b) Profits



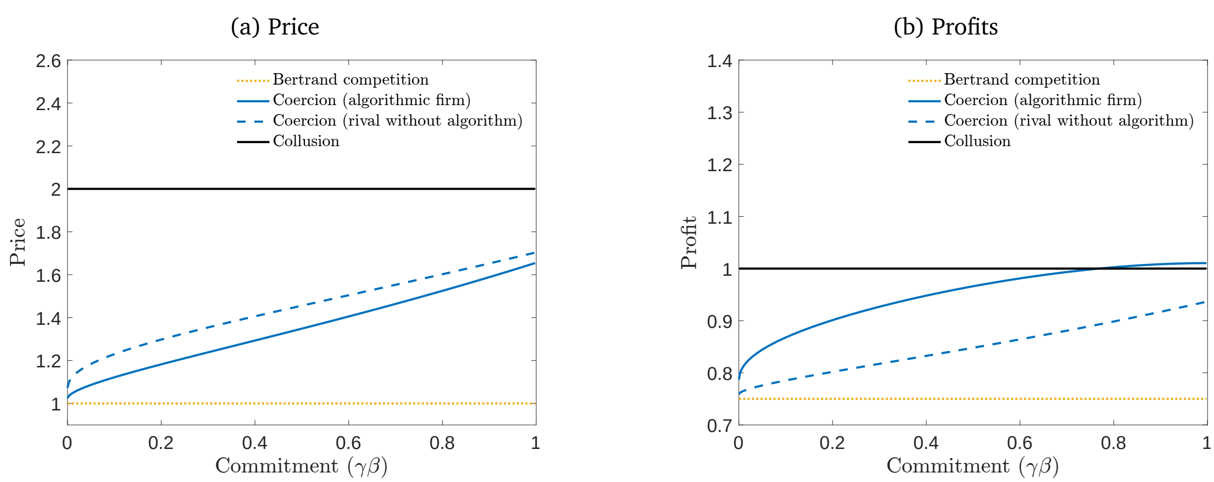
Notes: Panel (a) shows prices and panel (b) shows profits under Bertrand competition with simultaneous pricing, the coercive equilibrium with $\alpha = 0$ and commitment $\beta\gamma$, and joint profit maximization. Assumes $d = 1$ under linear demand given by equation (15).

Figure A-5: Prices and Profits in Coercive Equilibrium: Partial Commitment



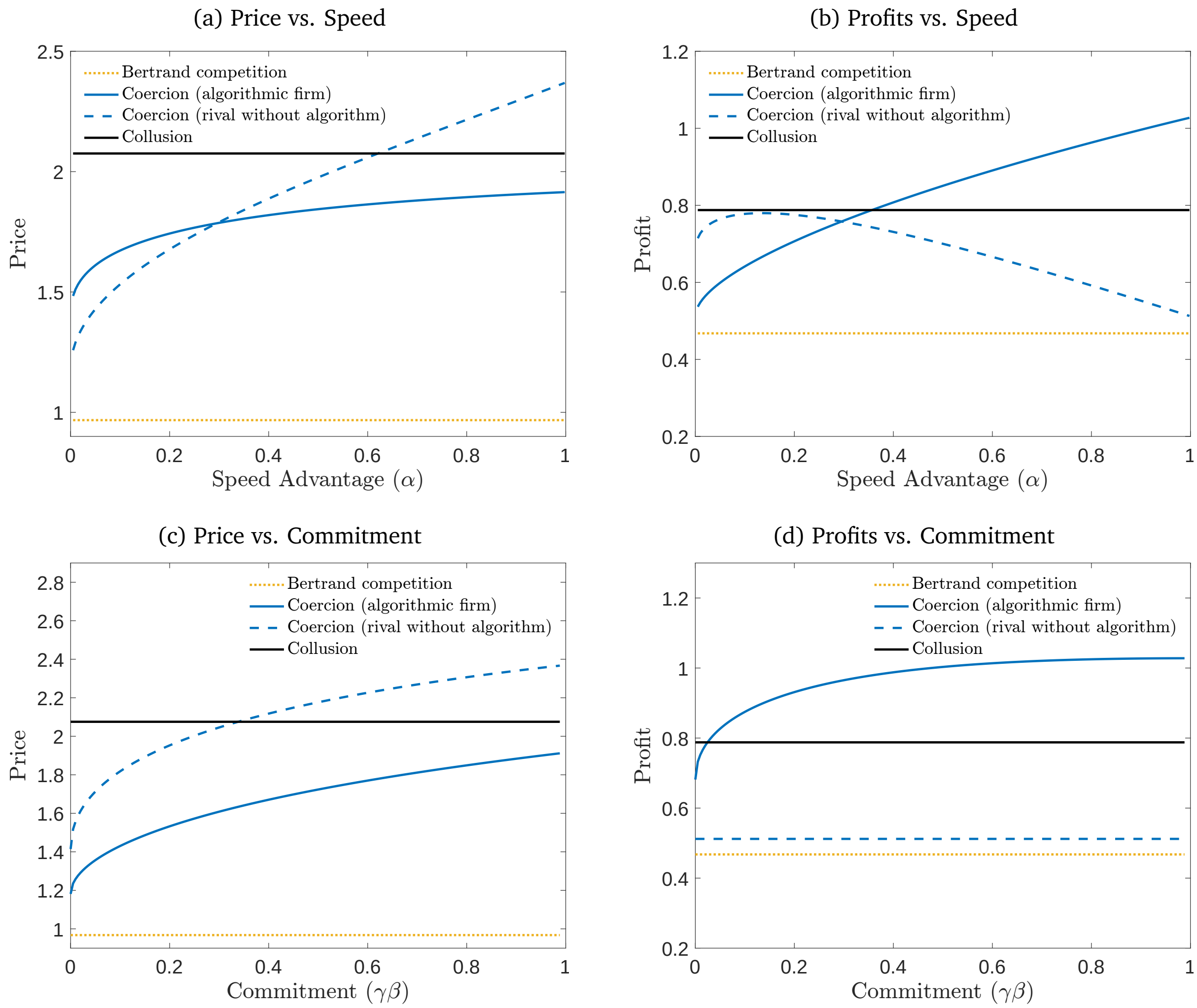
Notes: Panel (a) shows prices and panel (b) shows profits under Bertrand competition with simultaneous pricing, the coercive equilibrium with partial commitment ($\beta\gamma = 0.5$) and speed advantage α, and joint profit maximization. Assumes $d = 1$ under linear demand given by equation (15).

Figure A-6: Prices and Profits in Coercive Equilibrium: Intermediate Speed Advantage



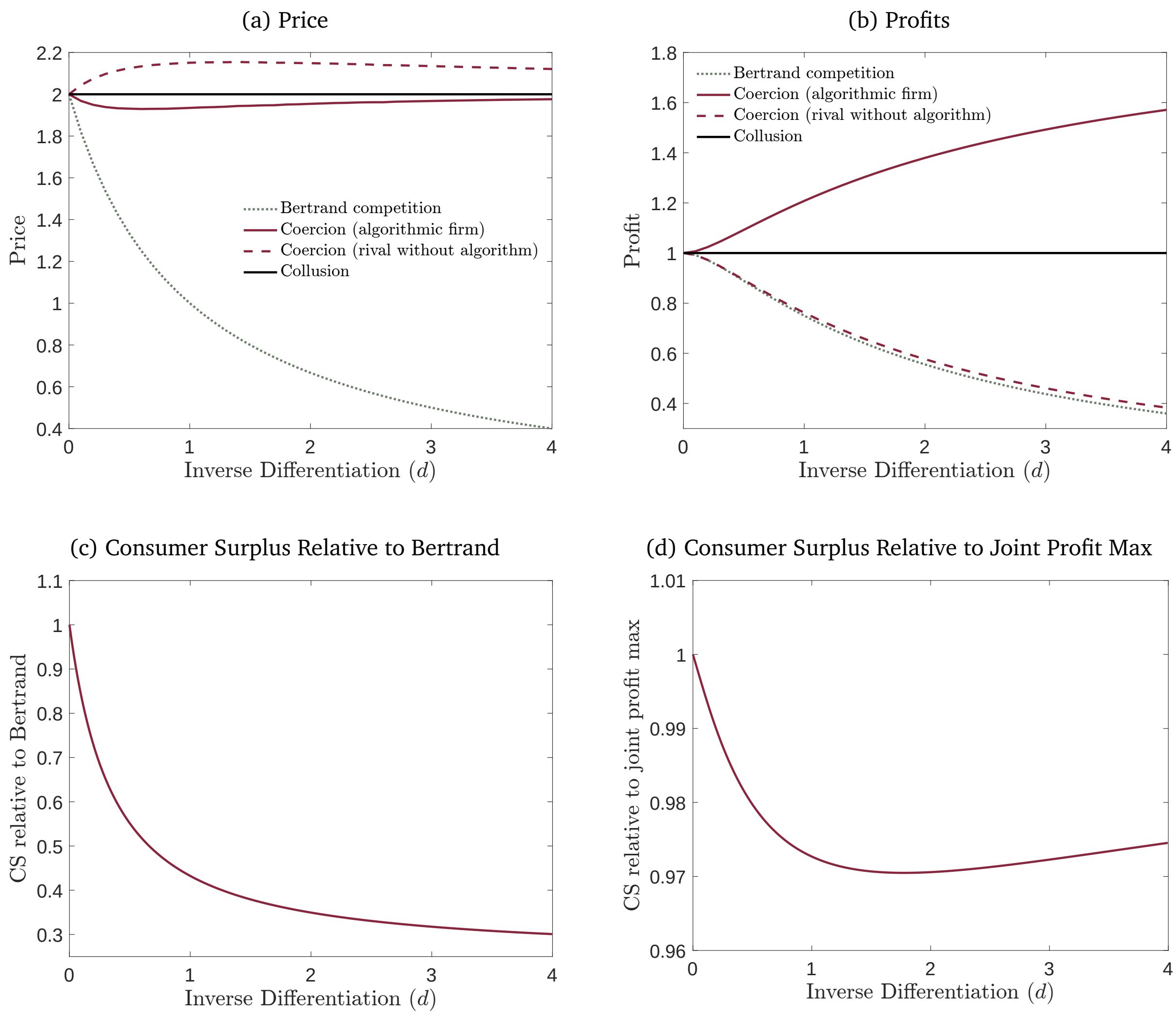
Notes: Panel (a) shows prices and panel (b) shows profits under Bertrand competition with simultaneous pricing, the coercive equilibrium with $\alpha = 0.5$ and commitment $\beta\gamma$, and joint profit maximization. Assumes $d = 1$ under linear demand given by equation (15).

Figure A-7: Prices and Profits in Coercive Equilibrium: Logit Demand



Notes: Panels (a) and (b) show equilibrium prices and profits for different values of the speed advantage α under the coercive equilibrium with full commitment ($\beta\gamma \to 1$). Panels (c) and (d) show equilibrium prices and profits under coercion for different values of commitment, $\beta\gamma$, when the faster firm has maximum speed ($\alpha = 1$). Outcomes under Bertrand competition with simultaneous pricing and joint profit maximization are displayed for comparison. Assumes logit demand given by $D_i(p_i, p_{-i}) = \exp(-\eta p_i)/[\zeta + \exp(-\eta p_i) + \exp(-\eta p_{-i})]$ where $\eta = 2$ and $\zeta = 0.01$.

Figure A-8: Prices and Profits in Coercive Equilibrium, by Product Differentiation



Notes: Panel (a) shows prices and panel (b) shows profits under Bertrand competition with simultaneous pricing, coercion, and joint profit maximization. Panel (c) shows consumer surplus under the coercion equilibrium relative to Bertrand and panel (d) shows consumer surplus under the coercion equilibrium relative to joint profit maximization. Considers the maximal coercion case with $\alpha = 1$ and $\beta\gamma \to 1$. Figures show equilibrium for different differentiation parameters d using linear demand given by equation (15).

Table A-1: Periods to Convergence for Coercive Equilibrium with Linear Pricing Rule

(a) Convergence Criterion of 1%

λ	Mean	Min	10th Percentile	25th Percentile	50th Percentile	75th Percentile	90th Percentile	Max
0.2	27.8	3	16	23	30	35	36	37
0.4	13.1	3	8	11	14	16	17	17
0.6	7.7	3	6	7	8	9	9	9
0.8	6.7	3	5	5	8	8	9	9
1	7.1	3	4	7	7	8	8	10
1.2	8.3	3	6	7	9	10	10	10

(b) Convergence Criterion of 0.1%

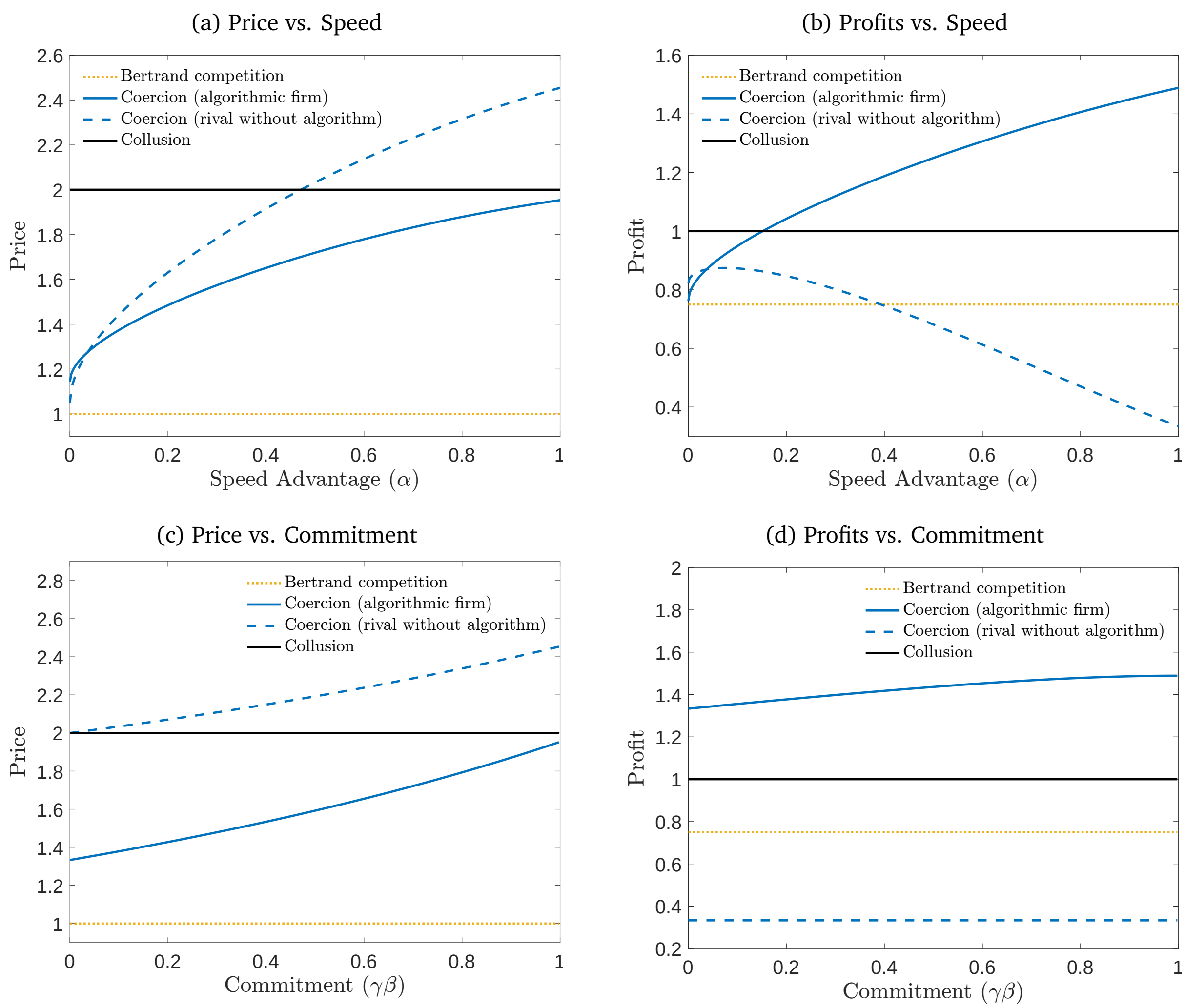
λ	Mean	Min	10th Percentile	25th Percentile	50th Percentile	75th Percentile	90th Percentile	Max
0.2	45.2	3	33	40	48	52	54	55
0.4	20.2	3	15	18	21	23	24	24
0.6	10.6	3	9	10	11	12	12	12
0.8	9.4	6	8	9	10	10	10	10
1	10.6	6	8	10	11	11	12	14
1.2	12.4	7	10	12	13	13	15	15

(c) Convergence Criterion of 0.01%

λ	Mean	Min	10th Percentile	25th Percentile	50th Percentile	75th Percentile	90th Percentile	Max
0.2	62.6	3	51	58	65	70	71	72
0.4	27.4	4	23	25	28	30	31	31
0.6	13.4	6	12	13	14	14	15	15
0.8	12.9	8	10	13	14	14	14	14
1	14.3	9	12	14	15	15	15	17
1.2	16.8	11	13	16	16	19	19	19

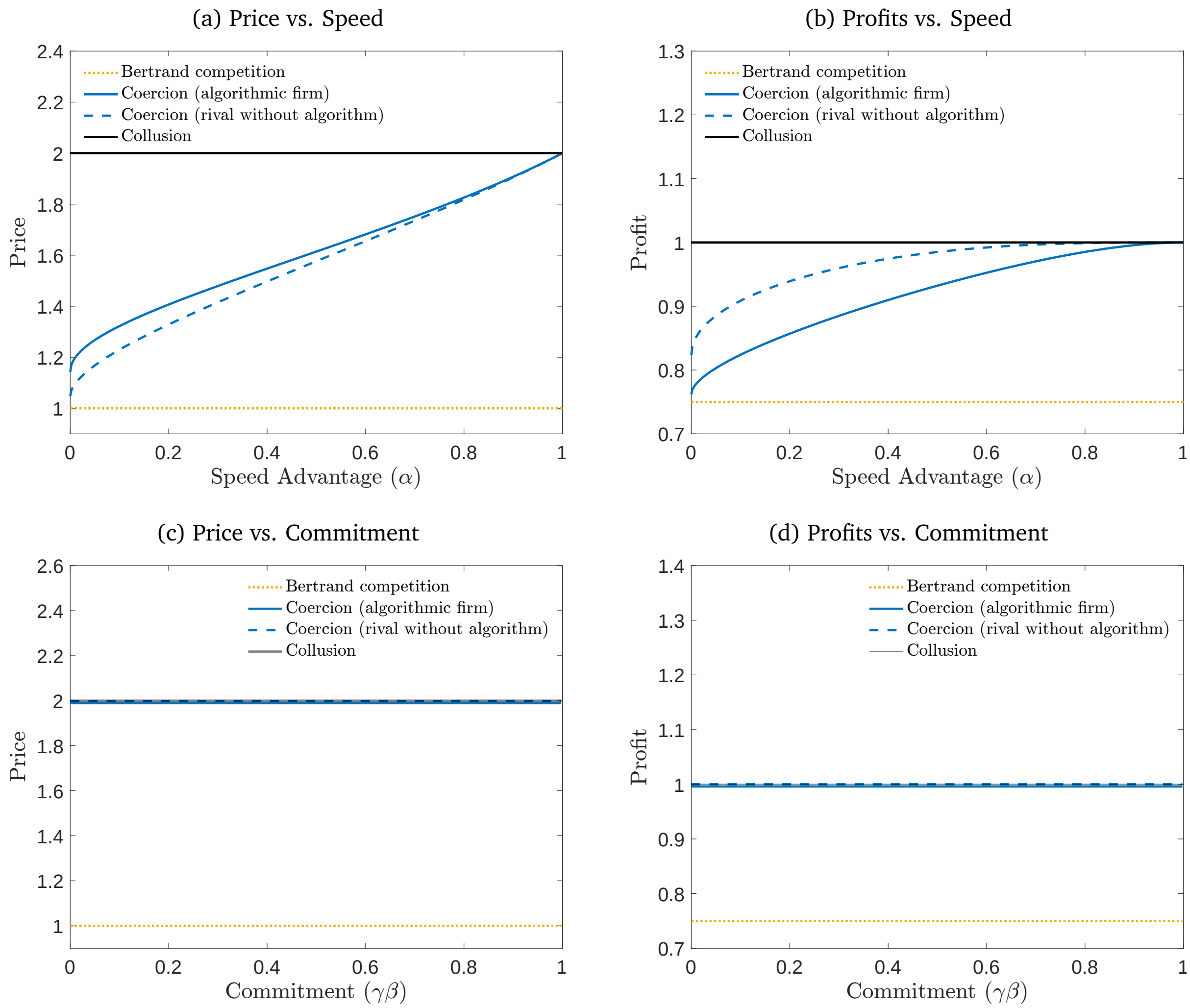
Notes: Table shows statistics of the number of periods until convergence when the algorithmic firm uses a linear pricing rule and the rival is assumed to use gradient learning. Rows indicate values of the price adjustment step size, λ. Convergence is defined as being within 1% of the equilibrium price for panel (a), within 0.1% of the equilibrium price for panel (b), and within 0.01% of the equilibrium price for panel (c). For each value of λ, we simulate price paths from 1,000 values of initial starting prices from a grid between 0 and 2 and report the mean, min, max, and percentiles. Simulation assumes algorithmic firm has maximum pricing speed ($\alpha = 1$) and uses the linear pricing rule described in Section 5.2. Assumes $d = 1$ under linear demand given by equation (15).

Figure A-9: Prices and Profits in Coercive Equilibrium: Harsh Punishment



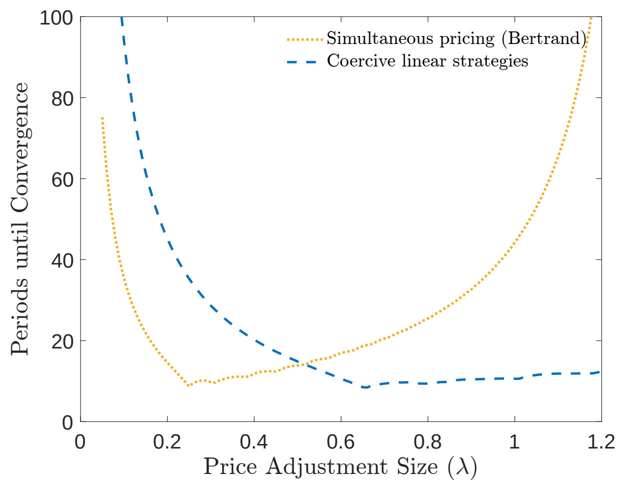
Notes: Panels (a) and (b) show equilibrium prices and profits for different values of pricing speed α under coercion with full commitment ($\beta\gamma \to 1$). Panels (c) and (d) show equilibrium prices and profits under coercion for different values of commitment, $\beta\gamma$, when the faster firm has maximum speed ($\alpha = 1$). Outcomes under Bertrand competition with simultaneous pricing and joint profit maximization are displayed for comparison. Unlike Figures 3 and 4, we assume firm a uses a punishment price of 0 rather than the static best response. Assumes $d = 1$ under linear demand given by equation (15).

Figure A-10: Prices and Profits in Coercive Equilibrium: Price Matching Punishment



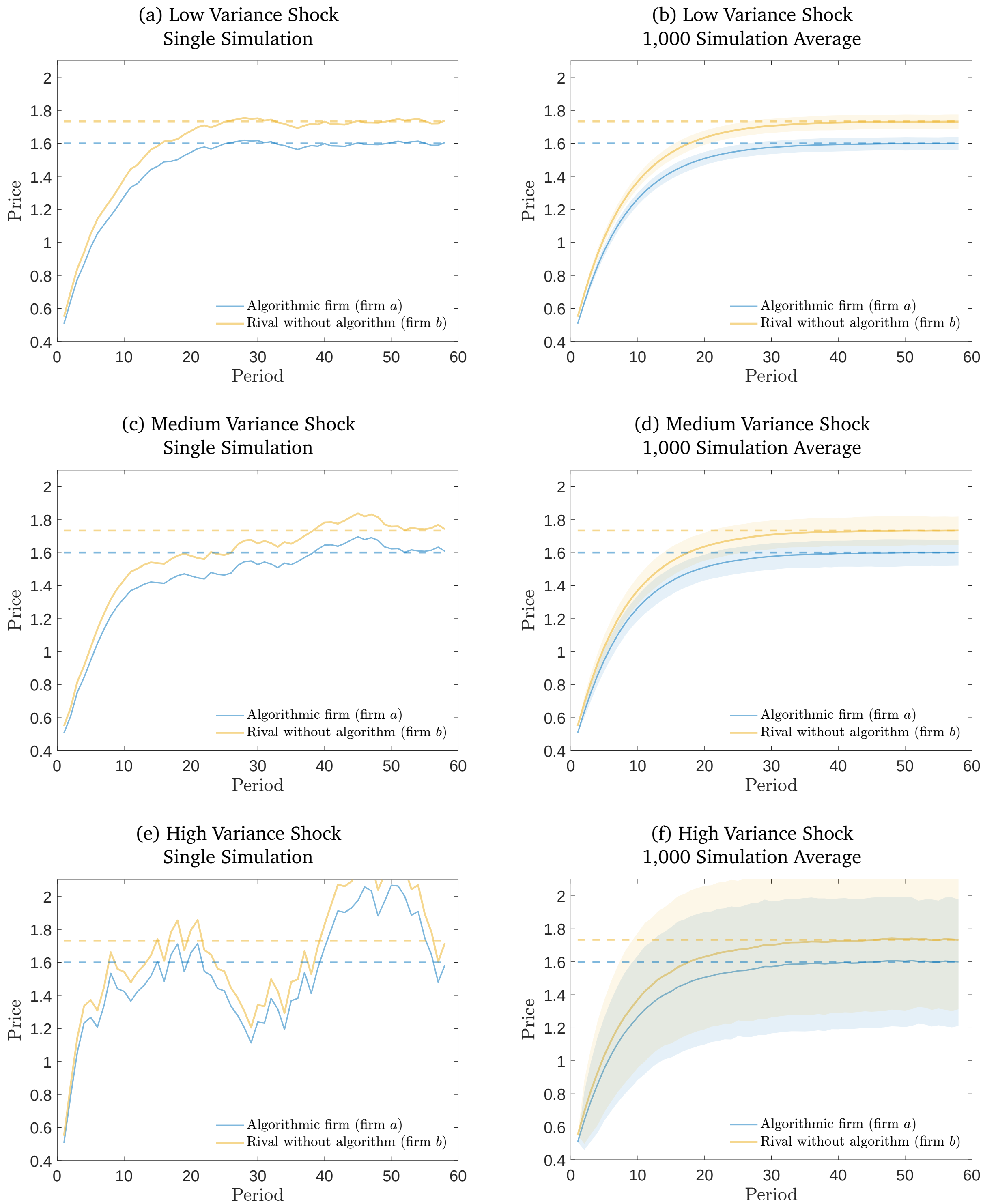
Notes: Panels (a) and (b) show equilibrium prices and profits for different values of speed advantage α under coercion with full commitment ($\beta\gamma \to 1$). Panels (c) and (d) show equilibrium prices and profits under coercion for different values of commitment, $\beta\gamma$, when the faster firm has maximum speed ($\alpha = 1$). Outcomes under Bertrand competition with simultaneous pricing and joint profit maximization are displayed for comparison. Unlike Figures 3 and 4, we assume firm a uses a punishment function that matches rival's price rather than the static best response. Assumes $d = 1$ under linear demand given by equation (15).

Figure A-11: Convergence Speed with Gradient Learning by Price Step Size



Notes: Figure shows the mean number of periods until convergence when a naive firm uses gradient learning with price adjustment step size λ. Convergence is defined as being within 0.1% of the equilibrium price. For each value of λ, we simulate price paths from 1,000 values of initial starting prices from a grid between 0 and 2 and then average the number of periods until convergence. The blue lines show a simultaneous pricing game that results in Bertrand prices. The orange lines show the game in which the algorithmic firm has maximum pricing speed ($\alpha = 1$) and uses the linear pricing rule described in Section 5.2. Assumes $d = 1$ under linear demand given by equation (15).

Figure A-12: Convergence of Simulated Learning with Demand Shocks



Notes: Charts show simulated price paths when firm b uses gradient learning with uniformly distributed demand shocks with mean 1. Top panel shows shock with standard deviation 0.05, middle panel shows shock with standard deviation 0.1, and bottom panel shows shock with standard deviation 0.5. Demand shock affects the constant in linear demand given by equation (15). Left charts show a single simulation while right charts show the average of 1,000 simulations with error bars showing two standard deviations from the mean. Dashed lines show equilibrium prices derived in the proof for Proposition 5. The figure considers the maximal coercion case with $\alpha = 1$ and $\beta\gamma \to 1$.