Tacit Collusion by Pricing Algorithm with Rule-Based Rivals

Yongzhi Xu

Texas A&M University

January 4 2026

Background

- Market environment is complicated.
  - Demand function.
  - Potential rivals.
- Many firms are unable to solve for the optimal prices.
  - Amazon third-party sellers.
  - Gasoline stations.
- Retailers turn to rely on automated pricing software.

Pricing Algorithm

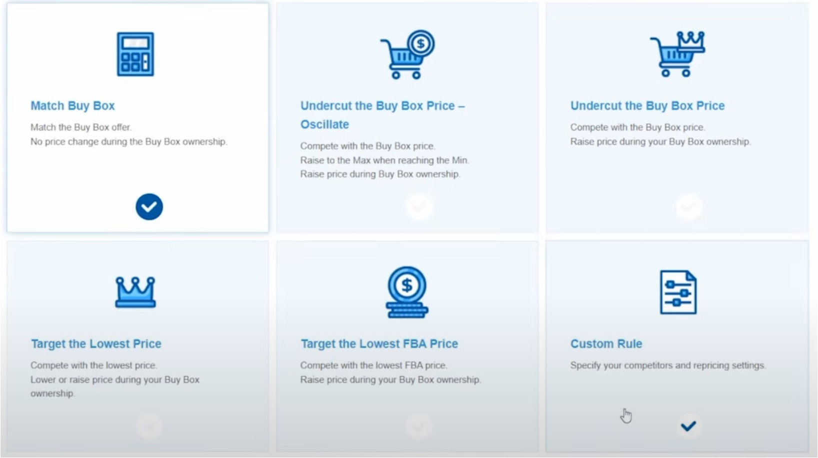
- Rule-based strategy: predetermined. (*Rule*)
  - Pricing undercutting.
  - Price trigger.
  - Tit-for-tat.

Pricing Algorithm

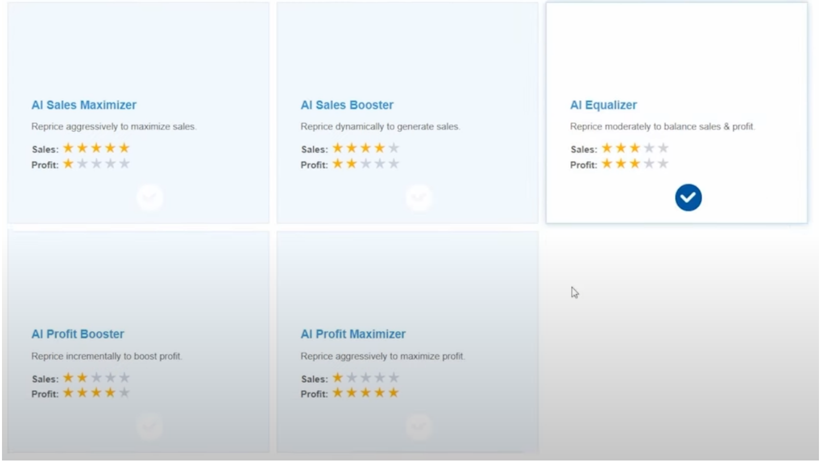
- Rule-based strategy: predetermined. (*Rule*)
  - Pricing undercutting.
  - Price trigger.
  - Tit-for-tat.

- Learning-based algorithm: updating. (*Algorithm*)
  - Reinforcement learning.

Motivating Example: Rule-Based Repricing (Amazon)



Motivating Example: AI-Powered Repricing (Amazon)



Motivation

- Learning-based algorithms may lead to supracompetitive prices (price level higher than Nash equilibrium price).
  - Evidence from simulation and empirical study.
  - Outcomes observed with simultaneous adoption.
- In practice, firms rarely adopt learning algorithms at the same time without communication.
  - Firms are unlikely to identify whether rivals are using learning-based algorithms or rule-based strategies.
  - Algorithms that begin learning with historical data can effectively "start earlier" than rivals.
- When a firm is the first to adopt an algorithm, it competes against a rule-based rival.

Sequential Adoption

- Algorithms learn from rivals' response.
- Simultaneous adoption hides the price dynamics.
  - Prior work observes only the high prices, but cannot identifying how prices change as each firm adopts.
- The transition from rules to algorithms is unexplored.
  - Particularly, can algorithms coordinate with existing rule-based rivals?
- Sequential adoption offers new implications for regulation.

Preview of Main Findings

- We develop theoretical model that shows an increase in price when the first firm adopts a reinforcement learning algorithm.
- The firm maintaining a rule-based strategy can "free ride" and benefits more from the other firm's adoption.
- Simulation results are consistent with the theoretical predictions across several widely used rule-based strategies.
- Firms charge supracompetitive prices when two firms both adopt an algorithm.

Contributions

- Provides the first investigation of sequential algorithm adoption, showing that algorithms lead not only to high prices, but also to price increases when adoption is non-simultaneous.
- Examines how reinforcement learning algorithms compete against rule-based rivals across a broad class of strategies
- Lays the groundwork for research and policy on the assessment and development of effective regulatory policies, by analyzing how sequential adoption and mixed competition affect market pricing.

Outlines

Introduction

Economic Model

Simulation Results

Conclusion

Rule-based Strategy

In a complicated environment, firms are competing with simple rules before time 0:

$$p_{i,t} = g_i(p_{j,t-1})$$

Examples:

- Price undercut: The price is rival's price minus one cent, i.e., $g(p) = p - 0.01$.
- Price trigger: Choose monopoly price if my rival does, and Bertrand price otherwise, i.e., $g(p) = p^M$ if $p = p^M$, $g(p) = p^N$ if $p < p^M$.

Assumptions

Let p^N be the static Nash equilibrium price, and p^M be the static monopoly price.

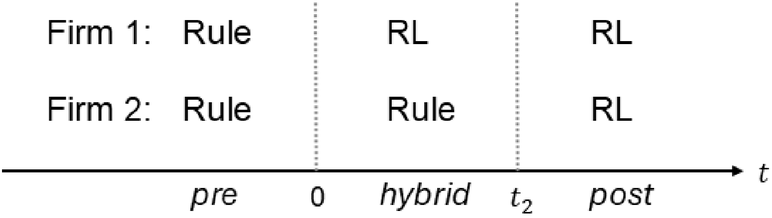
- $p, g_i(p) \in \mathcal{P} = \left[p^N, p^M\right]$

For price $p \in \mathcal{P}$

- $g_i(p) \leq p$.
- $g_i(p)$ is continuous and weakly increasing w.r.t. p.

Note that we are not assuming symmetric rules such that $g_1 = g_2$.

Timing



- pre stage: Firms are competing with simple rules $p_{i,t} = g_i(p_{j,t-1})$.
- hybrid stage: Firm 1 uses a reinforcement learning (RL) algorithm to compete with firm 2 (still using rule g_2).
- post stage: Both firms are using RL algorithm.

Baseline Model

- Infinitely repeated simultaneous game.
- 2 symmetric firms.
- In each period t, firm i earns a profit

$$\pi_i(p_{i,t}, p_{j,t}) = (p_{i,t} - c) * q_{i,t}(p_{i,t}, p_{j,t}),$$

- constant marginal cost $c = 1$.
- Firm 1 (algorithm) is trying to maximize discounted payoff

$$\sum_{t=1}^{\infty} \delta^t \pi_{i,t}(p_{i,t}, p_{j,t})$$

- Firm i can observe π_i, p_i, p_j but no other information.

Steady State in Hybrid Stage

Proposition

Let p^* be the static optimal price of firm 1 given firm 2's strategy g_2, such that

$$p^* := \arg\max_p \pi_1(p, g_2(p))$$

Then there always exists a $\delta_0 \in [0, 1)$ such that $\forall \delta \in (\delta_0, 1)$, p^* is the unique fixed point of g^*, i.e., $p_1^* = p^*$ is the unique steady state.

Price Increase

To summarize, we can show that

1. An RL algorithm will always converge to policy g^*.
2. p^* is the unique steady state under policy g^*.
3. $p^* \geq p^0, g_2(p^*) \geq g_2(p^0)$, where p^0 is the steady-state price of the pre period.

Combined all above, we can conclude that the RL algorithm will increase the price in the hybrid stage.

This result could be extended to markets with asymmetric firms and multiple firms.

Adoption Decision

Table: Payoff matrix of the adoption game.

		Firm 2	
		Rule	Algorithm
Firm 1	Rule	(π_0, π_0)	(π_2^*, π_1^*)
	Algorithm	(π_1^*, π_2^*)	$(\bar{\pi}, \bar{\pi})$

- A pure-strategy NE $(Algorithm, Algorithm)$ if $\bar{\pi} \geq \pi_2^*$.
- A mixed-strategy NE such that both firms adopt an algorithm with probability $(\pi_1^* - \pi_0)/(\pi_1^* + \pi_2^* - \pi_0 - \bar{\pi})$.

Outlines

Introduction

Economic Model

Simulation Results

Conclusion

Q-learning: pseudo-code

Initialize $Q_i^0(s, a)$

$t = 1, s^1 = \text{random}$

record actions $Opt_i(s)$

while $t < 10^8$

$$a_i^t = \begin{cases} \arg\max_a Q_i^t(s^t, a) & \text{with prob. } 1 - \varepsilon^t \\ \text{random} & \text{with prob. } \varepsilon^t = e^{-\beta t} \end{cases}$$

if $a_i^t == Opt_i(s^t)$ for 10^5 continuous periods

break

else

$Opt_i(s^t) = a_i^t$

end if

$$Q_i^{t+1}(s^t, a_i^t) = (1 - \alpha) Q_i^t(s^t, a_i^t) + \alpha \left[\pi_i^t + \delta \max_{a \in A} Q_i^t(s^{t+1}, a_i^t) \right]$$

end while

Parametrization

- constant marginal costs $c_i = 1$
- product/firm $i = 1, 2$, product 0 is the outside good
- period $t = 1, 2, \ldots$
- vertical differentiation $\gamma_i = 2, \gamma_0 = 0$
- horizontal differentiation $\mu = 1/4$
- logit demand

$$q_{i,t} = \frac{e^{\frac{\gamma_i - p_{i,t}}{\mu}}}{\sum_{j=1}^{n} e^{\frac{\gamma_i - p_{i,t}}{\mu}} + e^{\frac{\gamma_0}{\mu}}}$$

- reward $\pi_{i,t} = (p_{i,t} - c_i)q_{i,t}$

Initialization

- State $s^{t+1} = (a_1^t, a_2^t)$.
- $\mathbf{Q}_0$ is set at the discounted payoff that would accrue to player i if opponents randomized uniformly:

$$Q_{i,0}(s, a_i) = \frac{\sum_{a_{-i} \in A^{n-1}} \pi_i (a_i, a_{-i})}{(1-\delta)|A|^{n-1}}.$$

- Learning rate $\alpha = 0.05$.
- Greedy index $\beta = 1 \times 10^{-6}$.
- 1000 session for each setup.
- Price set $p \in \{p^1, \dots, p^{15}\}$, where p^2 is the static Bertrand price and p^{14} is the static monopoly price.

Measures

- To have a clearer view of collusion level, we use the price grid $1 \sim 15$ instead of the absolute prices:

$$p^1, p^N, p^3, \ldots, p^{13}, p^M, p^{15}$$

- To quantify the profits, we use a normalized measure:

$$\Delta = \frac{\bar{\pi} - \pi^N}{\pi^M - \pi^N}$$

Rule-based strategies

- Myopic

$$p_{i,t}(p_{-i,t-1}) = \arg\max \pi_{it}(p_{it}, p_{-i,t-1})$$

- Undercut

$$p_{i,t}(p_{-i,t-1}) = \begin{cases} p_{-i,t-1} - \Delta p & \text{if } p_{-i,t-1} > p^N \\ p^N & \text{o.w.} \end{cases}$$

- Trigger

$$p_{i,t}(p_{-i,t-1}) = \begin{cases} p^M & \text{if } p_{-i,t-1} = p^M \\ p^N & \text{o.w.} \end{cases}$$

- Ceiling

$$p_{i,t}(p_{-i,t-1}) = \begin{cases} p_{-i,t-1} & \text{if } p_{-i,t-1} \leq p^C \\ p^C & \text{o.w.} \end{cases}$$

where p^N is the Bertrand Nash equilibrium price and p^M is the monopoly price.

Hybrid Stage: Converged results

Table: Converged results in hybrid stage

Rule	Highest Pre s-s	Predicted Hybrid s-s	Observed Probability	Δ
Myopic	(p^2, p^2)	(p^8, p^5)	100%	$(0.18, 0.85)$
Undercut	(p^2, p^2)	(p^{14}, p^{13})	100%	(0.84,1.15)
Trigger	(p^{14}, p^{14})	(p^{14}, p^{14})	100%	(1.00,1.00)
Ceiling	(p^7, p^7)	(p^7, p^7)	100%	(0.61,0.61)

Notes: The observed probability is the probability that prices converge to the predicted steady state in simulations.

Hybrid Stage: Myopic Rule

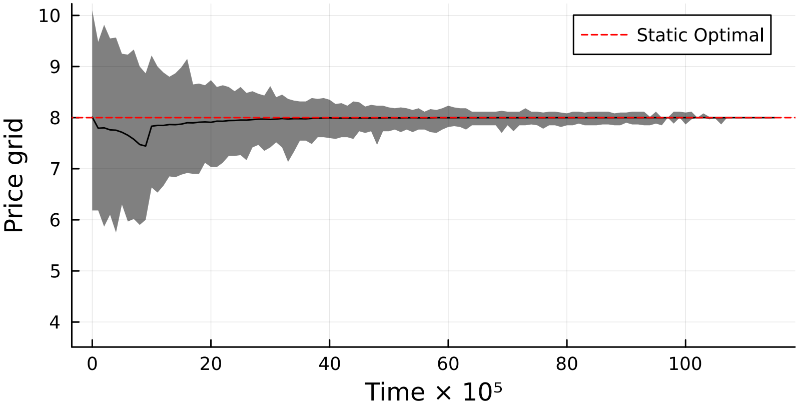


Figure: Myopic: Firm 1

Other Rules

Hybrid Stage: Myopic Rule

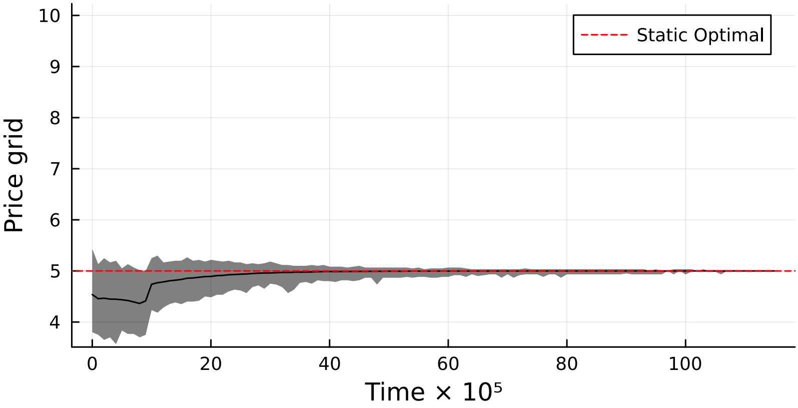


Figure: Myopic: Firm 2

Other Rules

Post Stage: Myopic Rule

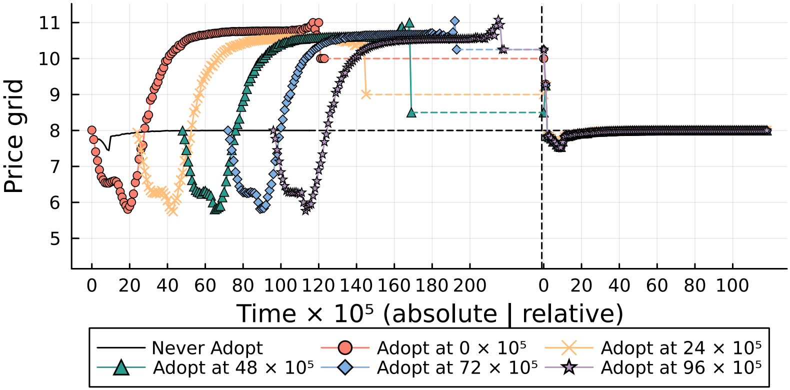


Figure: Myopic: Firm 1

Other Rules

Post Stage: Myopic Rule

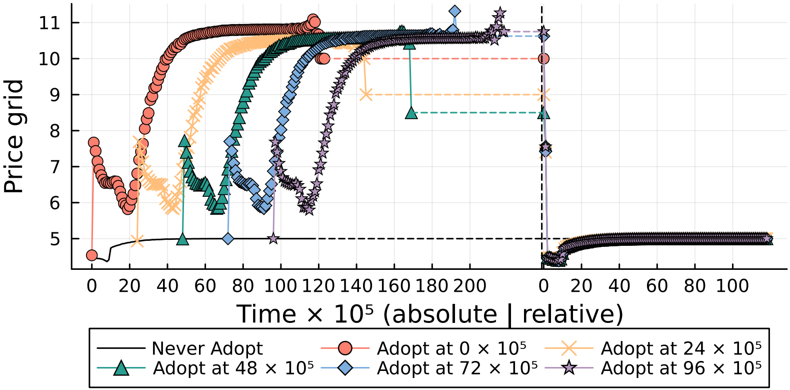
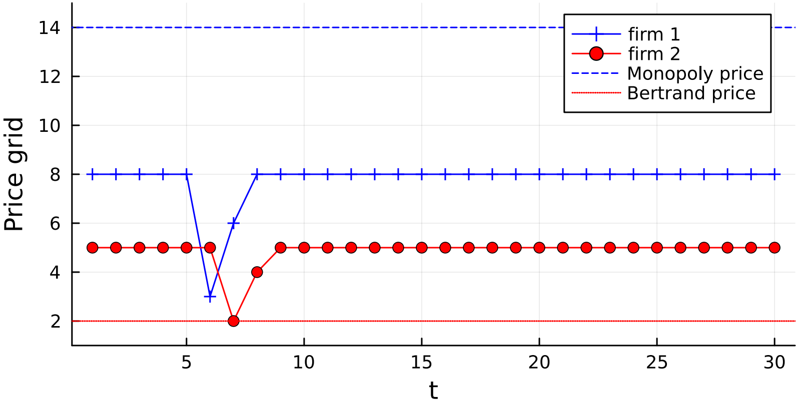


Figure: Myopic: Firm 2

Other Rules

Deviation: Myopic Rule



Other Rules

Outlines

Introduction

Economic Model

Simulation Results

Conclusion

Conclusion

- We develop theoretical model that shows an increase in price when the first firm adopts a reinforcement learning algorithm.
- The firm maintaining a rule-based strategy can "free ride" and benefits more from the other firm's adoption.
- Simulation results are consistent with the theoretical predictions across several widely used rule-based strategies.
- Firms charge supracompetitive prices when two firms both adopt an algorithm.

Thank you!

Reference I

Hybrid Stage: Undercut Rule

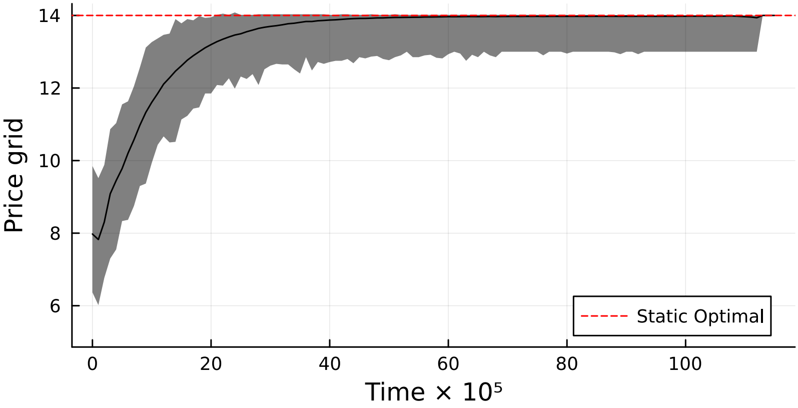


Figure: Undercut: Firm 1

Back

Hybrid Stage: Undercut Rule

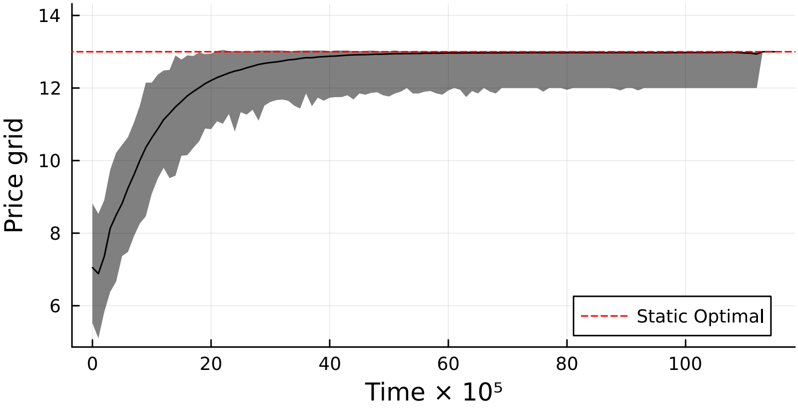


Figure: Undercut: Firm 2

Back

Hybrid Stage: Trigger Rule

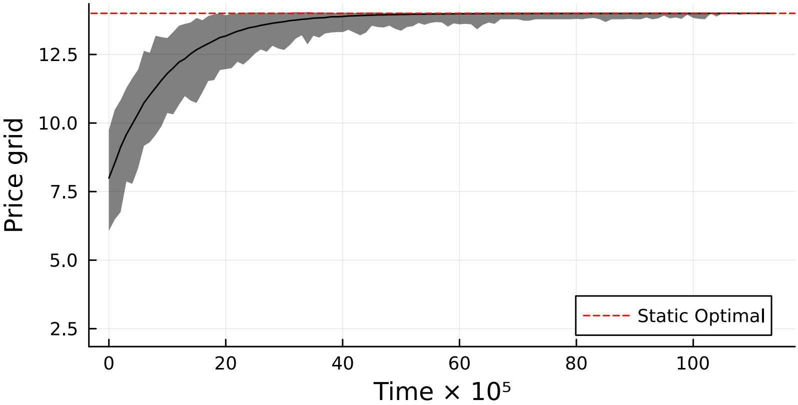


Figure: Trigger: Firm 1

Back

Hybrid Stage: Trigger Rule

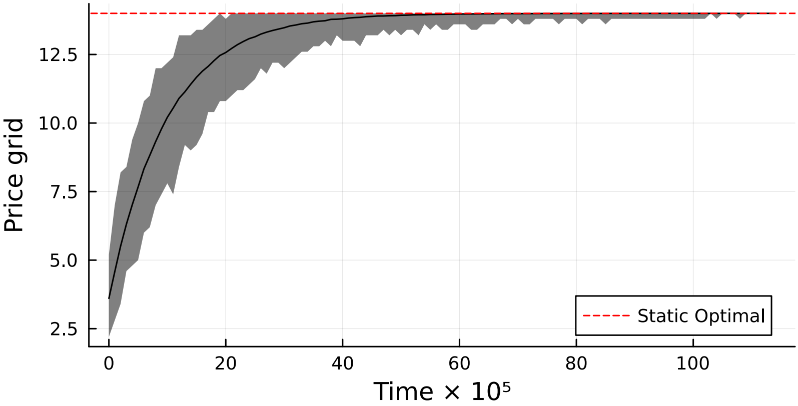


Figure: Trigger: Firm 2

Back

Hybrid Stage: Ceiling Rule

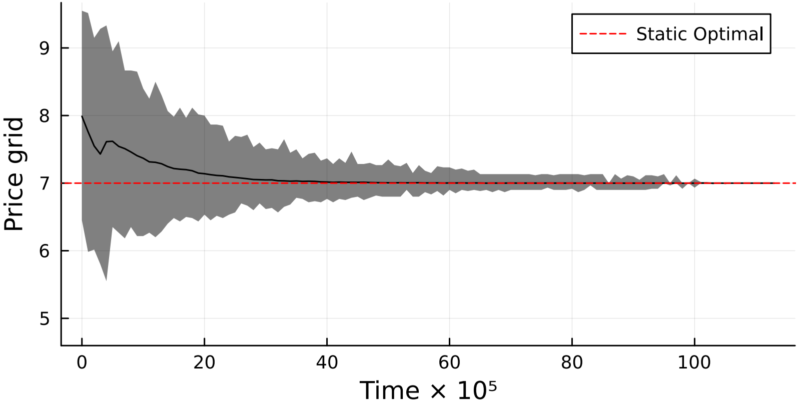


Figure: Ceiling: Firm 1

Back

Hybrid Stage: Ceiling Rule

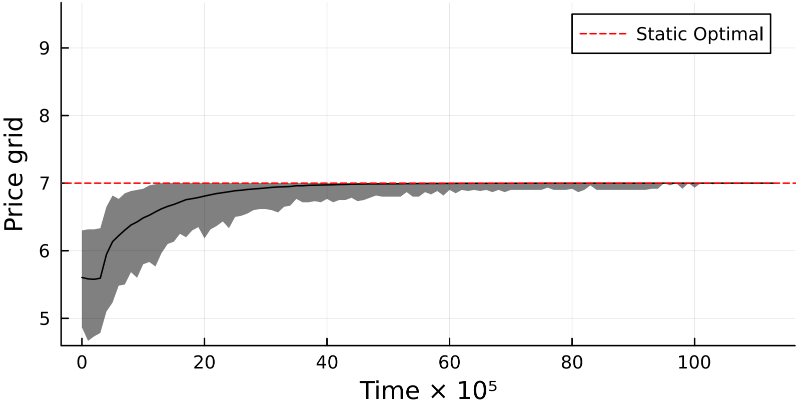


Figure: Ceiling: Firm 2

Back

Hybrid Stage: Undercut Rule

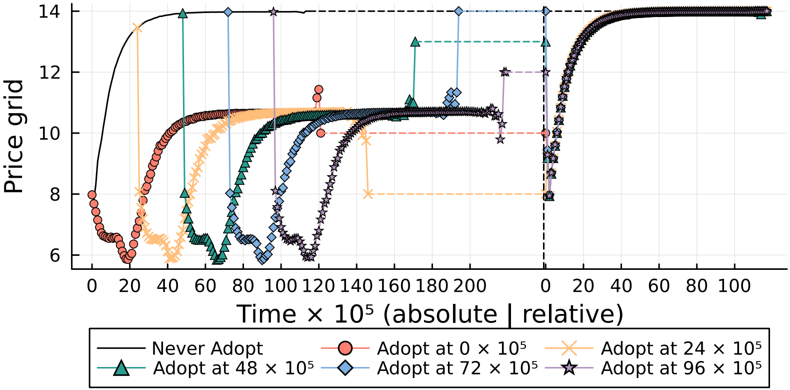


Figure: Undercut: Firm 1

Back

Hybrid Stage: Undercut Rule

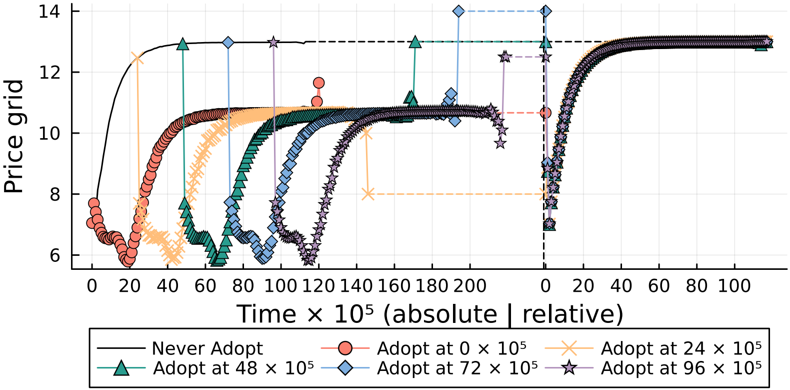


Figure: Undercut: Firm 2

Back

Hybrid Stage: Trigger Rule

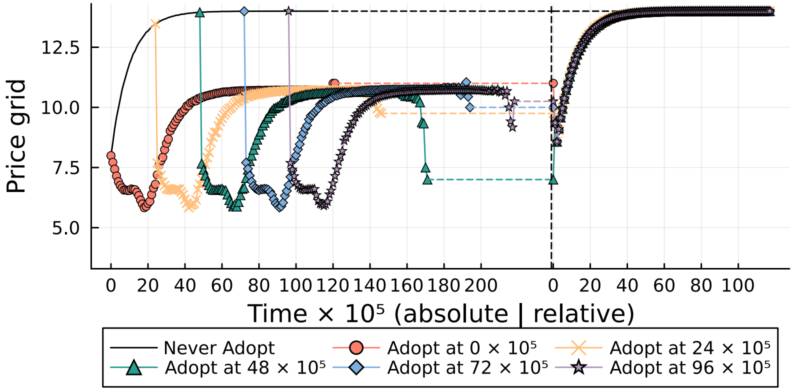


Figure: Trigger: Firm 1

Back

Hybrid Stage: Trigger Rule

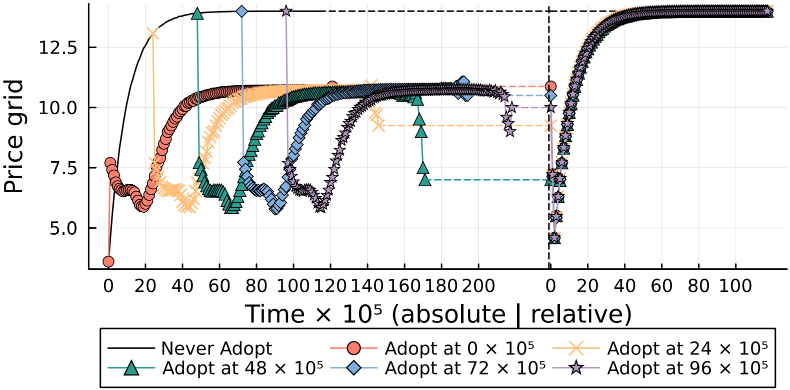


Figure: Trigger: Firm 2

Back

Hybrid Stage: Ceiling Rule

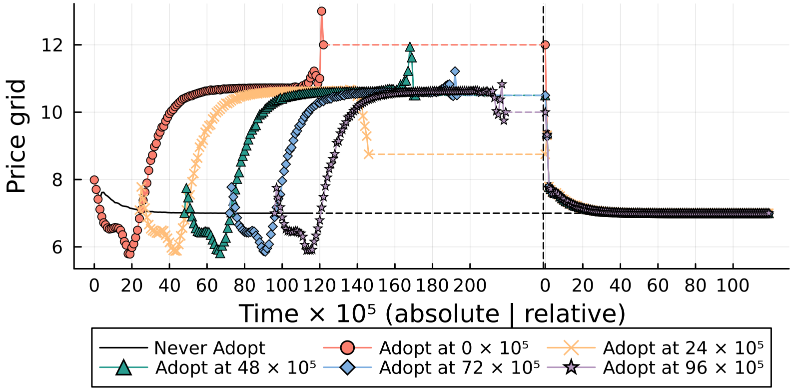


Figure: Ceiling: Firm 1

Back

Hybrid Stage: Ceiling Rule

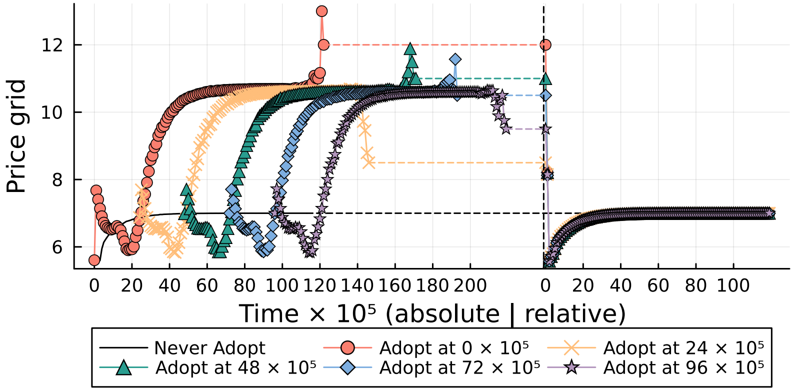
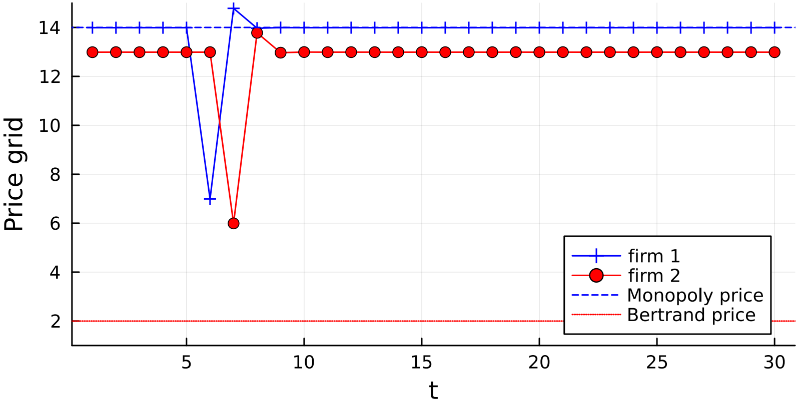


Figure: Ceiling: Firm 2

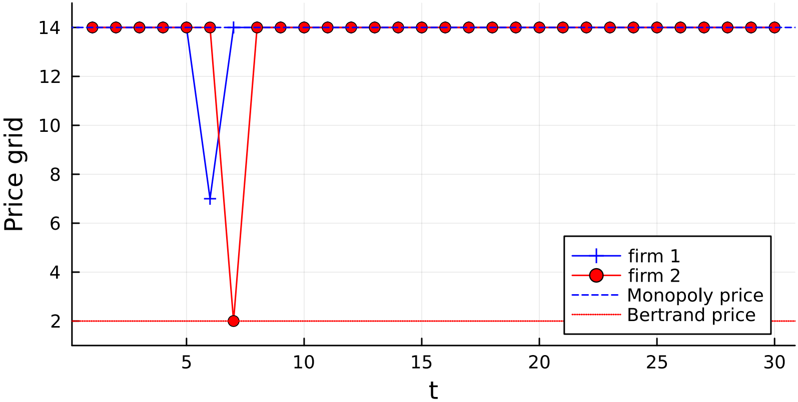
Back

Deviation: Undercut Rule



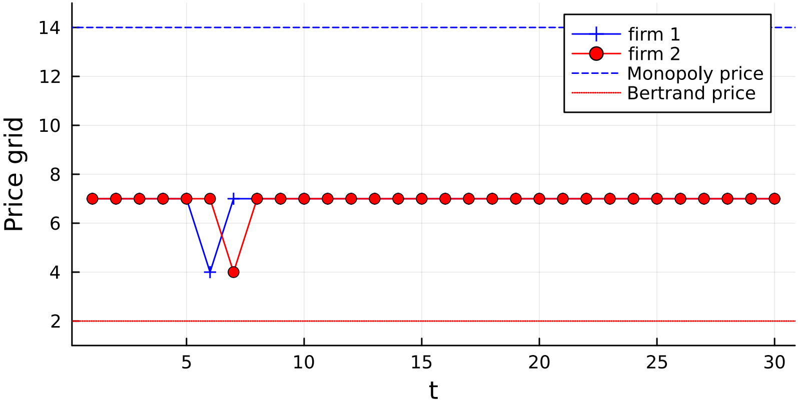
Back

Deviation: Trigger Rule



Back

Deviation: Ceiling Rule



Back