Pricing Algorithms and Algorithmic Collusion

Danish Association for Competition Law, November 25, 2025

Paul Hünermund

COPENHAGEN BUSINESS SCHOOL
HANDELSHØJSKOLEN

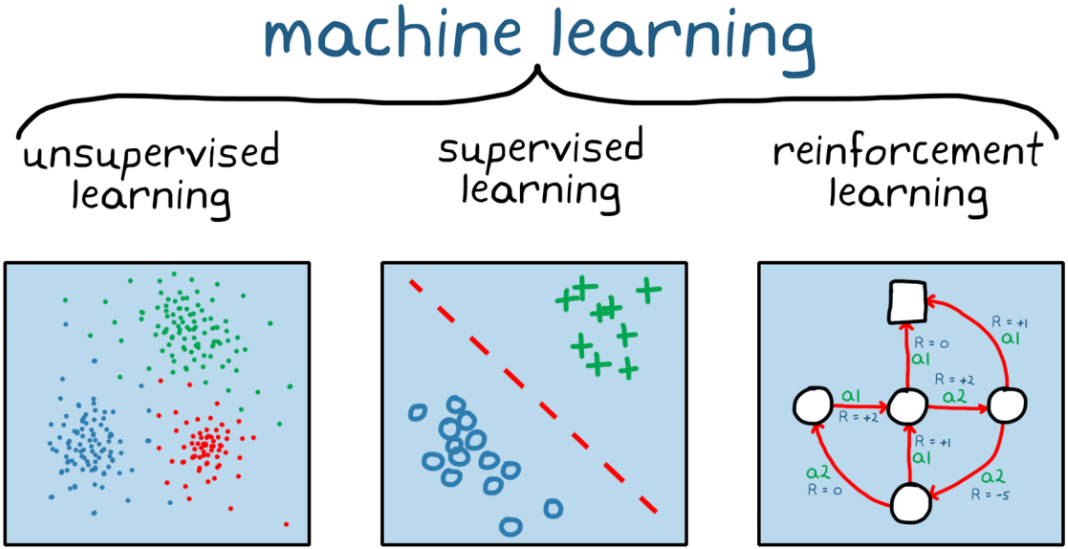
Competition in Markets Run by Algorithms

Collusion vs. Tacit Collusion

- **Collusion (hardcore cartel conduct)**
  - Rivals coordinate to keep prices high and profits up – explicitly illegal
  - Econ intuition: with n firms and marginal cost c
    - Competitive benchmark: $p = c \Rightarrow \pi = 0$
    - Cartel outcome: all charge p^m and share monopoly profits $\pi_i = \pi^m/n$
- **Tacit collusion**
  - Firms align pricing *without* an explicit agreement or info exchange
  - Often one firm raises prices and others simply "go along"
  - Also illegal, but much harder to detect and prove

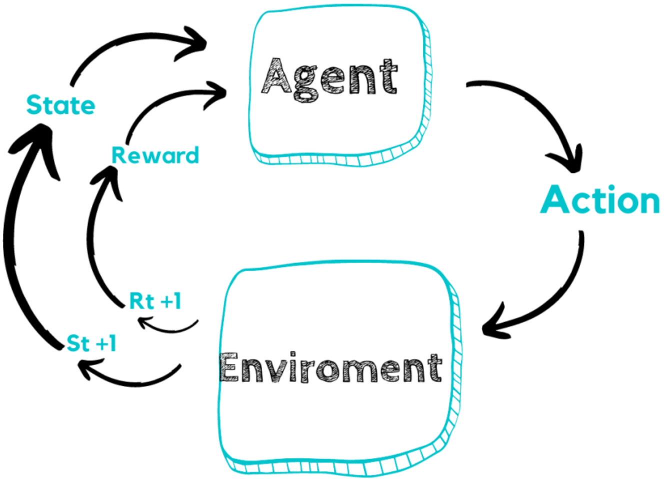
Sustaining Tacit Collusion

- Coordination is held together by threat of retaliation
- Typical **"grim trigger"** logic:
  - If everyone charges $p^m \Rightarrow$ keep cooperating
  - If someone undercuts to $p^m - \varepsilon \Rightarrow$ others punish with a price war: $p = c$ for k periods
- Short-term gain from cheating vs. long-term losses from retaliation
- Result: firms can maintain cartel-like pricing *even without talking*



Source: MathWorks

Pricing Algorithms Are Based on RL



Source: KITRUM

Multi-armed Bandit Problem

Q-learning in a Nutshell

- The agent observes a state s_t
  - E.g., the prices the other agents played in the current and past k periods
- The agent takes an action a_t
  - E.g., playing a price itself
- The agent receives a reward π_t and system moves on to new state s_{t+1}
- The goal is to maximize the expected present value of rewards

$$E\left[\sum_{t=0}^{\infty} \delta^t \pi_t\right]$$

- The agent does not know which action in which state leads to the highest reward
  - It iteratively updates beliefs about optimal actions given feedback from previous play
- Trade-off between **exploration vs. exploitation**

Updating of the Q-matrix

- Q-function represents the discounted payoff of taking action a in state s

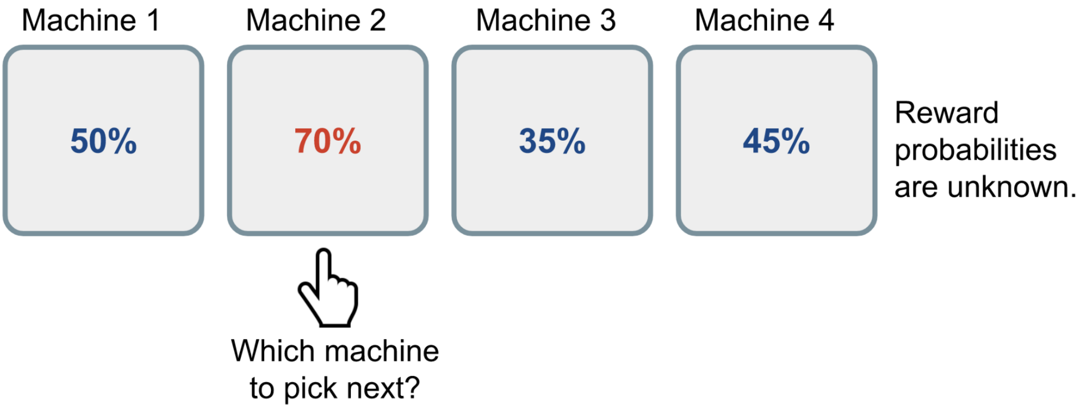
$$Q(s,a) = E(\pi|s,a) + \delta E\left[\max_{a' \in A} Q(s',a')|s,a\right]$$

- S and A are finite, we can therefore represent it by a $|S| \times |A|$ matrix
- If the agent knew the Q-function, we could calculate the optimal action for any given state
- Q-learning is an iterative procedure to estimate the Q-matrix
- We update the Q-matrix according to the following learning equation

$$Q_{t+1}(s,a) = (1-\alpha)Q_t(s,a) + \alpha\left[\pi_t + \delta \max_{a \in A} Q_t(s',a)\right]$$

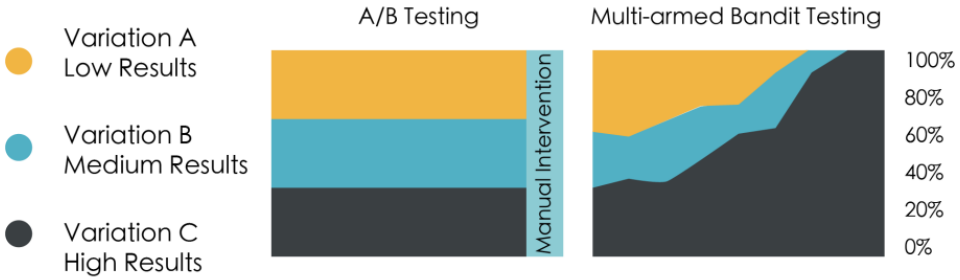
- The weight $\alpha \in [0,1]$ is called the learning rate

Exploration versus Exploitation



- ε-greedy model: choose currently optimal action (**exploitation**) with probability $1 - \varepsilon$ and randomize across all other options with probability ε (**exploration**)
  - ε will be set to a time-declining exploration rate: $\varepsilon = e^{-\beta t}$

Strategy Convergence over Time



Methodology of Calvano et al. (202, AER)

- Build autonomous pricing agents based on **tabular Q-learning**
  - Agents choose prices repeatedly and update Q-values from realized profits
  - No prior knowledge, no communication, fully unsupervised learning
- Run agents in **simulated oligopoly markets**
  - Baseline: symmetric duopoly with deterministic demand
  - Robustness: more firms, asymmetries, and stochastic environments
- Allow **very long interaction sequences** so strategies converge
- Evaluate not only price levels but **learned strategies** (reward-punishment patterns)

Simulation Results (Calvano et al., 2020)

$\Delta = 1$: perfectly collusive, $\Delta = 0$: perfectly competitive

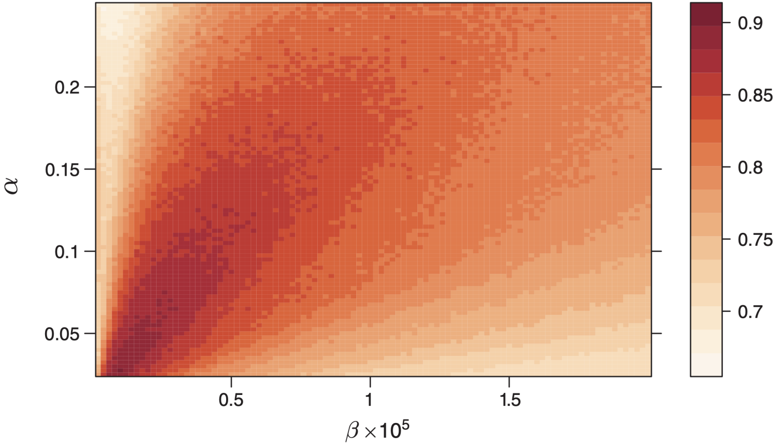


FIGURE 1. AVERAGE PROFIT GAIN Δ FOR A GRID OF VALUES OF α AND β

Simulation Results (II)

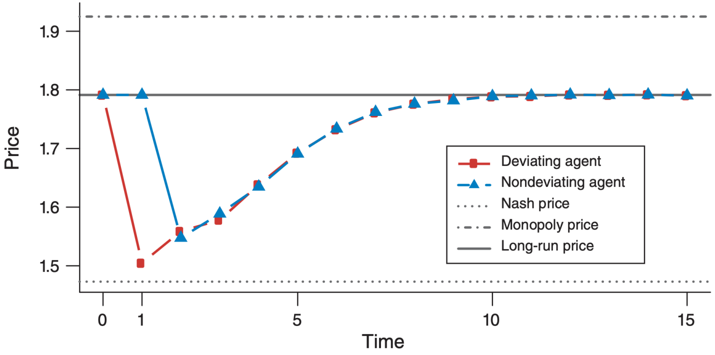


FIGURE 4

Notes: Prices charged by the two algorithms in period τ after an exogenous price cut by one of them in period $\tau = 1$. The forced cheater deviates to the static best response, and the deviation lasts for one period only. The figure plots the average prices across the 1,000 sessions. For sessions leading to a price cycle, we consider deviations starting from every point of the cycle and take the average of all of them. This counts as one observation in the calculation of the overall average.

Simulation Results (III)

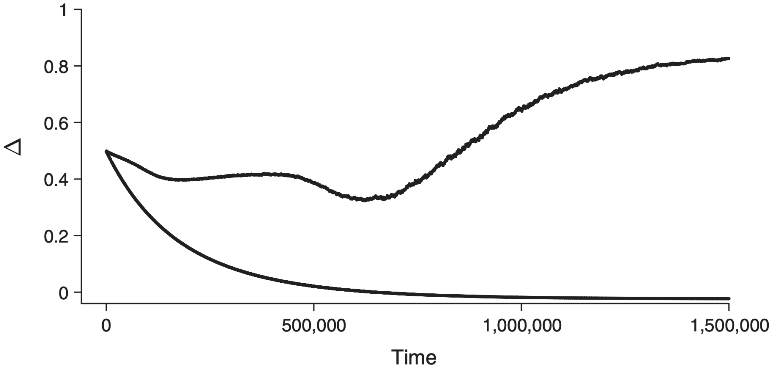


FIGURE 10

Notes: The average profit gain as a function of the number of repetitions (moving average over the last 100 repetitions). The dashed line is the profit gain that results from exogenous exploration, on the assumption that when they do not explore, the algorithms set the Bertrand-Nash price (approximated by defect).

Recent Theoretical Research on Algorithmic Collusion

- **Algorithms can collude**
  - Learning algorithms can reach *supra-competitive* prices without communication
  - (Calvano et al. 2020; Bichler, Durmann & Oberlechner 2024)
- **But collusion is not inevitable**
  - Depends on market structure and algorithm design
  - (Kühn & Tadelis 2018; Deng, Schiffer & Bichler 2025)
- **When is collusion more likely?**
  - Similar firms + similar algorithms + few competitors + ability to punish
- **When is collusion less likely?**
  - Heterogeneous algorithms and larger numbers of competitors

Empirical Evidence on Algorithmic Pricing

- **Algorithms can raise prices in practice**
  - Adoption of pricing algorithms in German fuel stations led to higher margins and higher prices *when multiple competitors adopted*
  - (Assad, Clark, Ershov & Xu 2024)
- **Growing use of pricing algorithms in e-commerce**
  - Algorithms detect and react to rivals' prices quickly; competition becomes less aggressive in some markets
  - (Hanspach, Sapi & Wieting 2024)
- **Algorithmic pricing changes competitive dynamics**
  - Fewer price wars and less undercutting observed after adoption; consistent with softer competition
  - (Musoff 2024)

Algorithmic Collusion on the Policy Agenda

Unclassified **DAF/COMP/WD(2017)12**

Organisation for Economic Co-operation and Development

14 June 2017

English - Or. English

DIRECTORATE FOR FINANCIAL AND ENTERPRISE AFFAIRS
COMPETITION COMMITTEE

Algorithms and Collusion - Note from the European Union

Key Messages from the European Commission Note

- **Algorithms can make collusion easier and more stable**
  - Real-time monitoring enables rapid detection of deviations
  - Automatic retaliation mechanisms can deter firms from lowering prices
  - Shared third-party pricing tools may create "hub-and-spoke" coordination risks
- **Tacit coordination may become more effective without explicit communication**
  - Prices can align purely through algorithmic adaptation and learning
  - Traditional evidence of an "agreement" may become harder to identify in practice
- **Legal accountability remains unchanged**
  - Firms cannot escape liability by claiming "the algorithm set the price"
  - Conduct that is illegal offline remains illegal when implemented online

Conclusions

- Pricing algorithms can learn to **systematically raise prices above competitive levels**
  - Not as low as Bertrand-Nash, but typically not as high as full monopoly
- Resulting coordination is **partial and self-enforcing**
  - Deviations are punished through temporary "price wars" followed by a gradual return to high prices
- Importantly: **No explicit agreement or communication is needed**
  - Algorithms learn by trial-and-error rather than being programmed to collude
- **Legal framework is challenged**
  - Tacit collusion is difficult to detect and prove in court
  - Authorities long assumed tacit collusion was hard for humans to sustain
  - **Algorithms change that calculus** and competition agencies are increasingly concerned

Thank you

Personal Website: p-hunermund.com

Bluesky: @p-hunermund.com

Email: phu.si@cbs.dk

COPENHAGEN BUSINESS SCHOOL
HANDELSHØJSKOLEN