Dynamic Pricing under Competition: Challenges and Opportunities

Dr. Rainer Schlosser

Keynote Session

ICORES 2023, Lisbon, Portugal

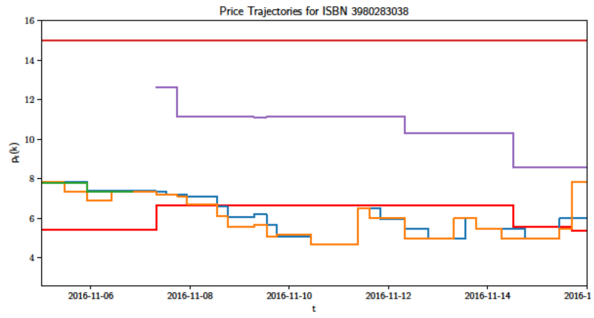
February 20, 2023

Motivation: Revenue Management & Pricing

- Revenue Management applications
  - – E-commerce is everywhere
  - – Prices are dynamic
- Challenges and opportunities!
  - – **Automation** is needed
  - – But how to do that effectively?

Prices on Amazon Marketplace

(*a used book over* 10 *days*)

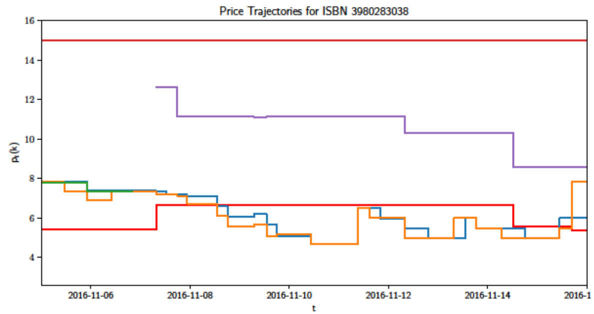


Motivation: Revenue Management & Pricing

- Revenue Management applications
  - – E-commerce is everywhere
  - – Prices are dynamic

Prices on Amazon Marketplace

(*a used book over* 10 *days*)



- Challenges and opportunities!
  - – **Automation** is needed
  - – But how to do that effectively?

- Approaches used **in practice**:
  - – Rule-based (suboptimal) & Optimal control (limited applicability)
  - – Will we see AI-based solutions (data hungry, less control) soon?

- **Vision**: Self-tuning data-driven solutions

Outline

- Topic: Dynamic Pricing & Ecommerce
- Personal Background
- I Analytical Solutions
- II Approaches in Practice
- III Self-Learning Approaches in Recommerce Markets
- IV Summary & Outlook

Academic Background

- Humboldt-University of Berlin
  - – Master in Business Administration (2010)
  - – Master in Mathematics (2010)
  - – PhD in **Operations Research** (2014) at the Institute of OR

    Thesis: Six Essays on Stochastic and Deterministic Dynamic Pricing and Advertising Models

- Field of Research
  - – Optimal Control of Markov Decision Processes (MDPs)
  - – Dynamic Pricing & Revenue Management (RM)

Group Leader at HPI

- Hasso Plattner Institute (HPI), University of Potsdam (since 2015)
  - PostDoc at the Chair (Enterprise Systems) of Prof. Plattner (cf. SAP)
  - ~12 PhDs working on Computer Science & Data Science
  - Established the group "**Data-driven Decision Support**" (3 PhDs)
  - Senior Researcher (since 2020)

- Field of Research
  - Control of MDPs, Dynamic Pricing & RM
  - Analytics, Decision Support
  - Self-tuning Algorithms, Resource Allocation Problems

Memories

- ICORES 2017 Dynamic Pricing
- ICORES 2018 Pricing with HMMs
- ICORES 2019 Strategic Consumers
- ICORES 2020 Ride-hailing Dispatch Decisions
- ICORES 2021 Pricing Competition
- ICORES 2022 Resource Allocations for Databases
- ICORES 2023 Reinforcement Learning Techniques
- ICORES 2024 :-)

Research Profile

Background

Business Adm.

Mathematics & OR

CS & Data Science

Research Focus in RM & Pricing

Analytical Solutions (Theoretical Analysis)

Numerical Solutions (Algorithms)

Application in Practice

Self-learning Solutions & Simulation

I Pricing in Theory

I Analytical Solutions: Overview

- How to set prices over time to optimally control a stochastic sales process?
- Typical model:
  - – MDP in continuous time, continuous price sets, monopoly
  - – State: remaining items; Rewards: sales profits
  - – Stylized dynamics (e.g., iso, exp, lin demand rates, Poisson-type)
- Solution approach: Dynamic programming (DP), Bellman equation
- Results: State-dependent optimal policy
  Managerial insights

I Analytical Solutions: Methodology

- Objective: Find a policy to maximize **expected discounted rewards**

  Basic example: Sell N items over the time span $[0,T]$, prices $p \geq 0$

- Approach: Consider the **value** of being in **state** $n \in \{0,...,N\}$ at **time** $t \in [0,T]$

  Use the Bellman equation to find **value function** $V_n(t)$

- Solution: 1st order optimality conditions of the Bellman equation

  Obtain a system of difference-differential equations for $V_n(t)$

  Solve for $V_n(t)$ and obtain an **optimal pricing policy** $p_n(t)$

- Insights: **Analyze** optimal prices at time t in state n (inventory left)

I Analytical Solutions: Bellman Equation

- Bellman Equation: $$\dot{V}_n(t) + \sup_{p \geq 0} \left\{ \lambda(t, p) \cdot \left(p - c - \Delta V_n(t) \right) \right\} = 0$$
- Boundary conditions: $V_n(T) = V_0(t) = 0 \qquad \forall n, t$

I Analytical Solutions: Bellman Equation

- Bellman Equation: $$\dot{V}_n(t) + \sup_{p \geq 0} \left\{ \lambda(t, p) \cdot \left(p - c - \Delta V_n(t) \right) \right\} = 0$$
- Boundary conditions: $V_n(T) = V_0(t) = 0 \qquad \forall n, t$
- Optimality conditions: $$p_n^*(t) - c - \Delta V_n(t) = \frac{\lambda(t, p_n^*(t))}{-\lambda'(t, p_n^*(t))}$$
- Diff.-DE: $$\dot{V}_n^*(t) + \lambda(t, p_n^*(t; \Delta V_n^*(t))) \cdot \left(p_n^*(t; \Delta V_n^*(t)) - c - \Delta V_n^*(t) \right) = 0$$

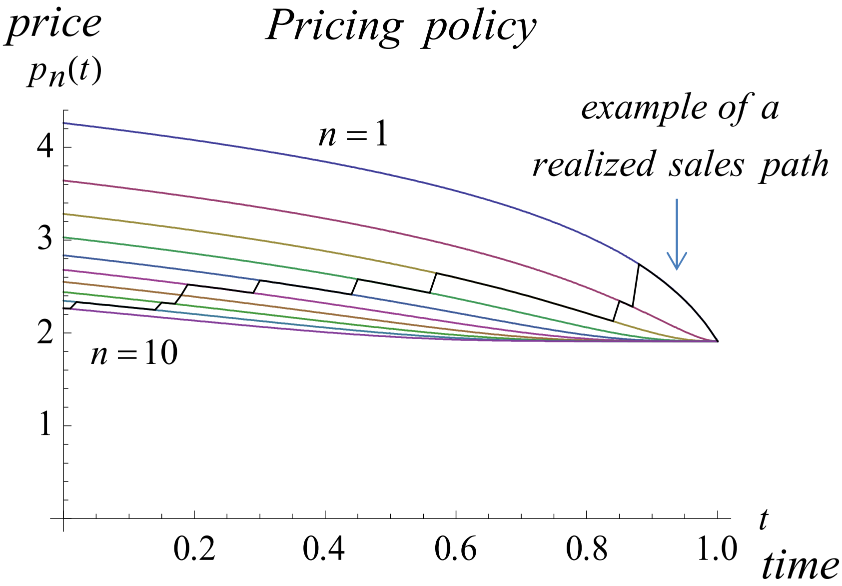
I Analytical Solutions: Bellman Equation

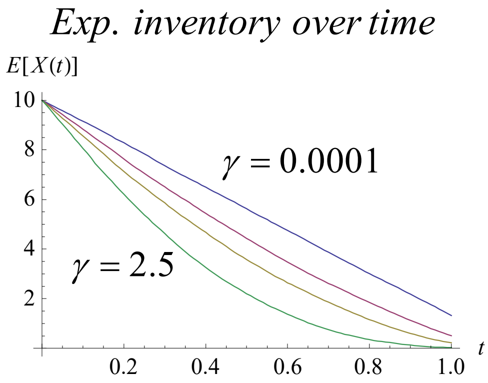
- Bellman Equation: $$\dot{V}_n(t) + \sup_{p \geq 0} \left\{ \lambda(t,p) \cdot \left(p - c - \Delta V_n(t) \right) \right\} = 0$$
- Boundary conditions: $V_n(T) = V_0(t) = 0 \qquad \forall n, t$
- Optimality conditions: $$p_n^*(t) - c - \Delta V_n(t) = \frac{\lambda(t, p_n^*(t))}{-\lambda'(t, p_n^*(t))}$$
- Diff.-DE: $$\dot{V}_n^*(t) + \lambda(t, p_n^*(t; \Delta V_n^*(t))) \cdot \left(p_n^*(t; \Delta V_n^*(t)) - c - \Delta V_n^*(t) \right) = 0$$
- Special Case: $\lambda(t,p) = a(t) \cdot e^{-\varepsilon \cdot p}$, $\dot{V}_n^*(t) + \beta(t) / d \cdot e^{-d \cdot \Delta V_n^*(t)} = 0$

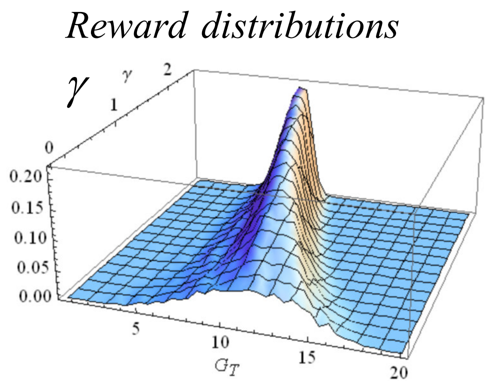
I Analytical Solutions: Bellman Equation

- Bellman Equation: $$\dot{V}_n(t) + \sup_{p \geq 0} \left\{ \lambda(t,p) \cdot \left(p - c - \Delta V_n(t) \right) \right\} = 0$$
- Boundary conditions: $V_n(T) = V_0(t) = 0 \qquad \forall n, t$
- Optimality conditions: $$p_n^*(t) - c - \Delta V_n(t) = \frac{\lambda(t, p_n^*(t))}{-\lambda'(t, p_n^*(t))}$$
- Diff.-DE: $$\dot{V}_n^*(t) + \lambda(t, p_n^*(t; \Delta V_n^*(t))) \cdot \left(p_n^*(t; \Delta V_n^*(t)) - c - \Delta V_n^*(t) \right) = 0$$
- Special Cases: $\lambda(t,p) = a(t) \cdot e^{-\varepsilon \cdot p}$, $\dot{V}_n^*(t) + \beta(t)/d \cdot e^{-d \cdot \Delta V_n^*(t)} = 0$
- Special Case Solution: $$V_n^*(t) = \frac{1}{d} \cdot \ln \left(\sum_{i=0}^{n} \left(\left(\int_t^T \beta(s) \; ds \right) \cdot (T-t) \right)^i \cdot \frac{1}{i!} \right)$$

I Analytical Solutions: Illustrations







I Analytical Solutions: Summary & Takeaways

(+) Beautiful closed-form solutions of differential equations

(+) Theoretical insights

(+) Sensitivity results

(+) Publishable

(–) Highly stylized, inflexible

(–) Limited to simple settings

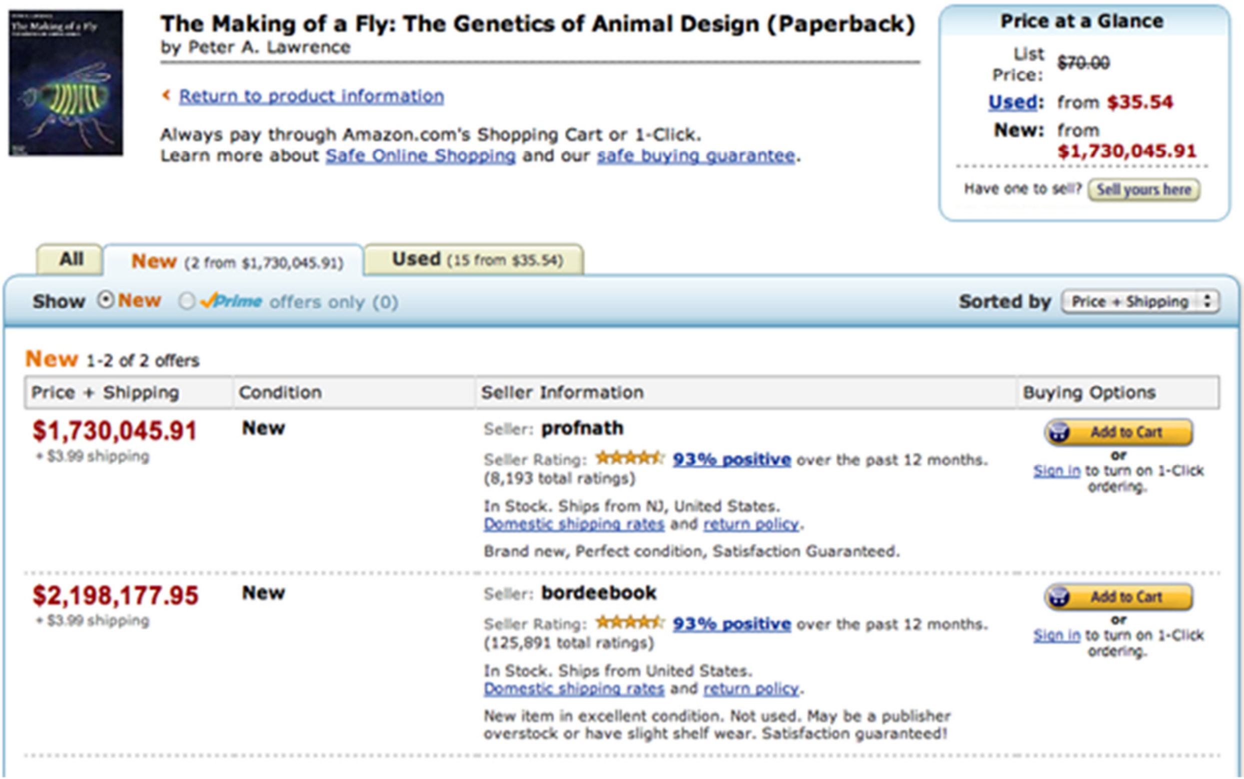
(–) **Hardly applicable in practice**

II Pricing in Practice

II Application in Practice: Online Pricing

- How to set prices in practice?
- **Project**: – Firm selling on Amazon MP
  – 100K distinct books (used)
  – ~10 updates/day/item (every 2-3h)
  – Competition
  – Multiple offer dimensions (price, quality, ratings, etc.)
- **Benchmark**: – Automated **rule-based** decisions of domain experts (Top10 seller)
  – includes: undercutting, cost-based, mark-down, . . .
- **Goal:** – Max **expected profits** & beat the firm's benchmark policy
  – Be able to balance **profitability vs. speed of sales**

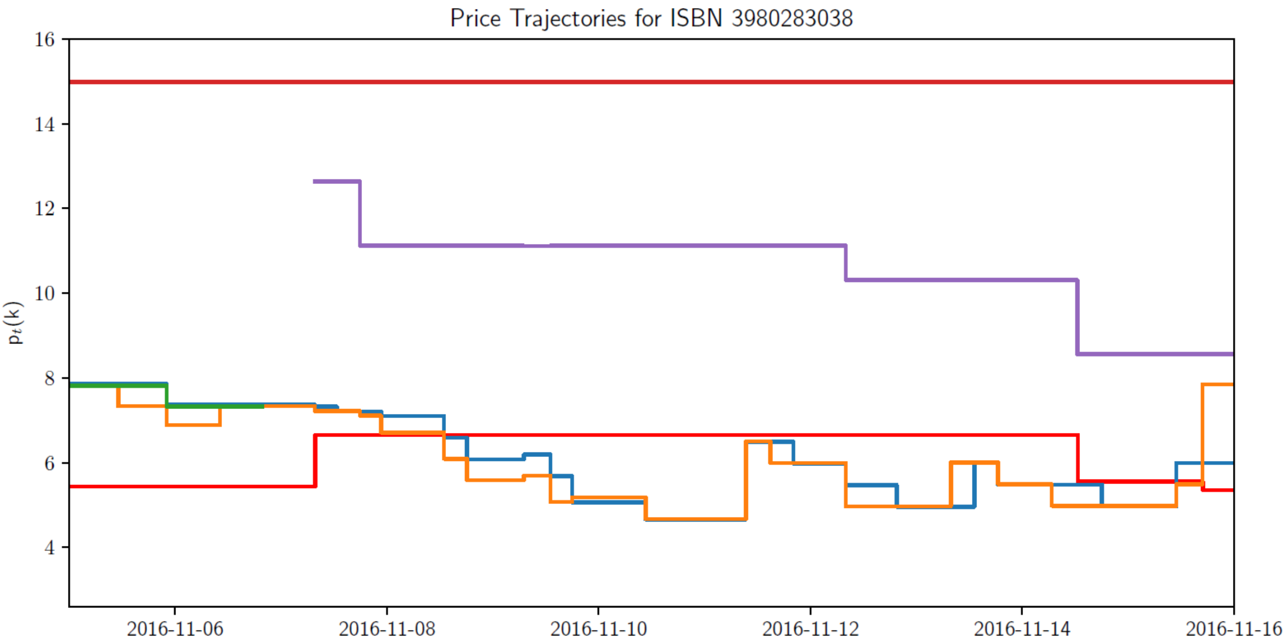
II Automated Repricing on Online Marketplaces (2011)



II Automated Repricing on Online Marketplaces (2011)

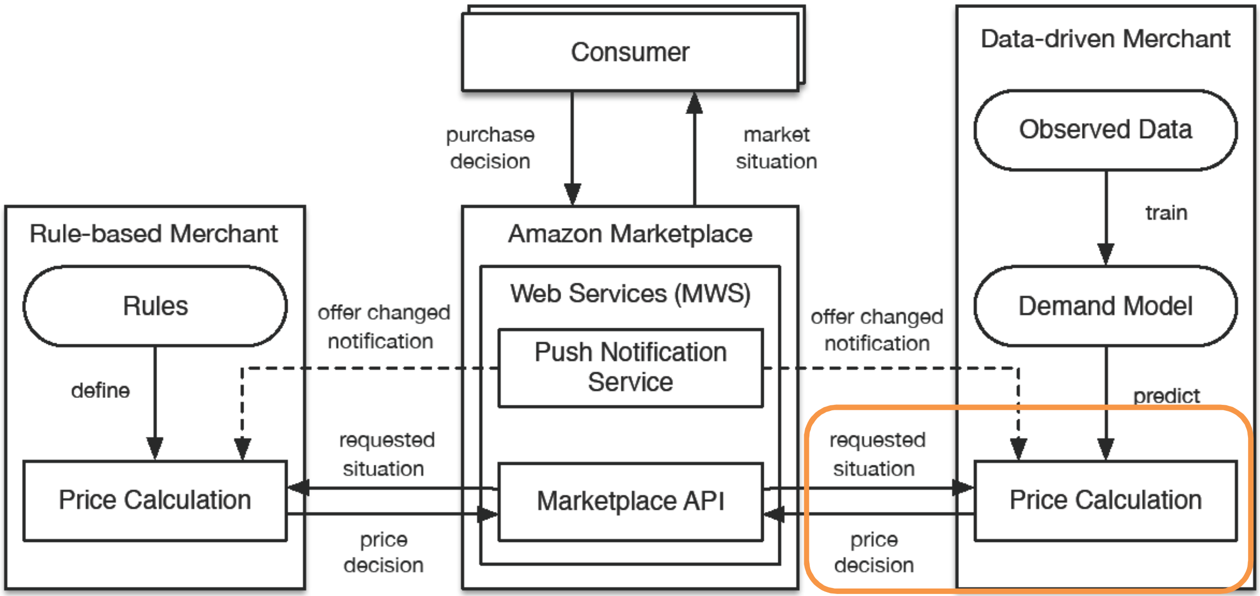
	profnath	bordeebook	profnath over previous bordeebook	bordeebook over profnath
8-Apr	$1,730,045.91	$2,198,177.95		1.27059
9-Apr	$2,194,443.04	$2,788,233.00	0.99830	1.27059
10-Apr	$2,783,493.00	$3,536,675.57	0.99830	1.27059
11-Apr	$3,530,663.65	$4,486,021.69	0.99830	1.27059
12-Apr	$4,478,395.76	$5,690,199.43	0.99830	1.27059
13-Apr	$5,680,526.66	$7,217,612.38	0.99830	1.27059

II Automated Repricing on Online Marketplaces (2016)



II Price Updates on Amazon Marketplace

- (i) request market situation, (ii) calculate price, (iii) send price update



II Project: Selling Used Books in Practice

- Our data-driven approach

  (1) Demand Estimation

  - ~10 market situations/day/item with 1-20 firms (**100 Mio obs**.)
  - 2 000 sales/month (1 year of data)
  - **Predict sales probabilities** (for time intervals & situations)

  (2) Price Optimization

  - Maximize long-term profit (**aggressiveness** via discount factor)
  - **Dynamic programming** (with relaxations)
  - Computation time for one final price adjustment: 0.001 seconds

II Estimation of Price Impacts and Optimization

- Our data-driven approach

(1) Demand Estimation

- ~10 market situations/day/item with 1-20 firms (**100 Mio obs**.)
- 2 000 sales/month (1 year of data)
- **Predict sales probabilities** (for time intervals & situations)

(2) Price Optimization

$$\max E(G_t \,|\, X_t = n, \vec{S}_t = \vec{s}_t), \qquad G_t := \sum_{s=t}^{T-1} \delta^{s-t} \cdot \Big(\big(a(X_s, \vec{S}_s) - c \big) \cdot \big(X_s - X_{s+1} \big) - l \cdot X_s \Big) \tag{1}$$

$$a(n, \vec{s}) = \arg\max_{a \in A} \left\{ \sum_{i \geq 0} \tilde{P}(i, a \,|\, \vec{s}) \cdot \Big((a - c) \cdot \min(n, i) - n \cdot l + \delta \cdot V\big((n-i)^+, \vec{s} \big) \Big) \right\} \tag{2}$$

$$V(n, \vec{s}) = \max_{a \in A} \left\{ \sum_{i > 0} \tilde{P}(i, a \,|\, \vec{s}) \cdot \begin{pmatrix} (a - c) \cdot \min(n, i) - n \cdot l \\ -z \cdot \delta \cdot V\big((n-i)^+, \vec{s} \big) \end{pmatrix} \Big/ \Big(1 - \tilde{P}(0, a \,|\, \vec{s}) \cdot z \cdot \delta \Big) \right\} \tag{3}$$

II Application in Practice: Results

Comparison: Our **data-driven** strategy vs. the seller's **rule-based** strategy

Our solution allows to balance the speed of sales vs. profitability

Strategy	#Books
Rule-Based	5,534
HPI1 (high prices)	5,206
HPI2	5,407
HPI3	5,241
HPI4 (low prices)	5,200

II Application in Practice: Results

Comparison: Our **data-driven** strategy vs. the seller's **rule-based** strategy

Our solution allows to balance the speed of sales vs. profitability

Strategy	#Books	% Sold (3 months)	
Rule-Based	5,534	42 %	100.0 %
HPI1 (high prices)	5,206	29 %	−30 %
HPI2	5,407	37 %	−12 %
HPI3	5,241	44 %	+6 %
HPI4 (low prices)	5,200	45 %	**+8 %**

II Application in Practice: Results

Comparison: Our **data-driven** strategy vs. the seller's **rule-based** strategy

Our solution allows to balance the speed of sales vs. profitability

Strategy	#Books	% Sold (3 months)		Profit per sale (EUR)	
Rule-Based	5,534	42 %	100.0 %	2.56 €	100.0 %
HPI1 (high prices)	5,206	29 %	–30 %	3.58 €	**+40 %**
HPI2	5,407	37 %	–12 %	3.03 €	+19 %
HPI3	5,241	44 %	+6 %	2.94 €	+15 %
HPI4 (low prices)	5,200	45 %	**+8 %**	2.52 €	–1 %

Result: Our strategy sold faster *and* more profitable!

II Application in Practice: Results

Comparison: Our **data-driven** strategy vs. the seller's **rule-based** strategy

Our solution allows to balance the speed of sales vs. profitability

Strategy	#Books	% Sold (3 months)		Profit per sale (EUR)		Acc. profit
Rule-Based	5,534	42 %	100.0 %	2.56 €	100.0 %	100.0 %
HPI1 (high prices)	5,206	29 %	–30 %	3.58 €	**+40 %**	–1.5 %
HPI2	5,407	37 %	–12 %	3.03 €	+19 %	+4.3 %
HPI3	5,241	44 %	+6 %	2.94 €	+15 %	**+23.1 %**
HPI4 (low prices)	5,200	45 %	**+8 %**	2.52 €	–1 %	+6.4 %

Result: Our strategy sold faster *and* more profitable!

II Pricing in Practice: Summary & Takeaways

(–) Ordinary numerical results

(–) No theoretical insights, no sensitivity results

(–) **State transitions** of the problem (MDP) have to be known/assumed

(–) Large datasets of good quality required

(–) **Dimensionality** of the MDP is limited (curse of dimensionality)

(+) Free use of estimations/predictions

(+) Data-driven DP heuristics outperform rule-based benchmarks

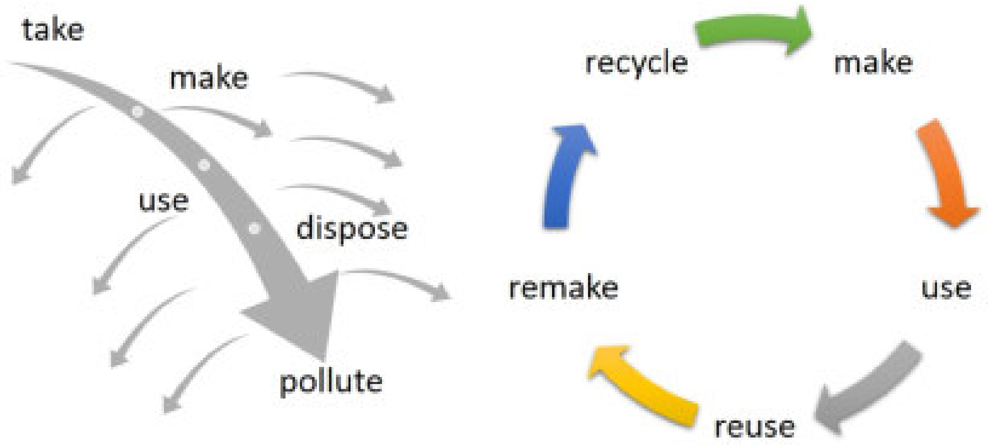
(+) Applicable in practice

III Self-Learning Approaches in More Complex Markets

(or: AI in the Circular Economy)

III Recommerce Markets: Motivation

Usecase: **Pricing & Rebuying** in the Recommerce Industry



CC 3.0 Catherine Weetman 2016

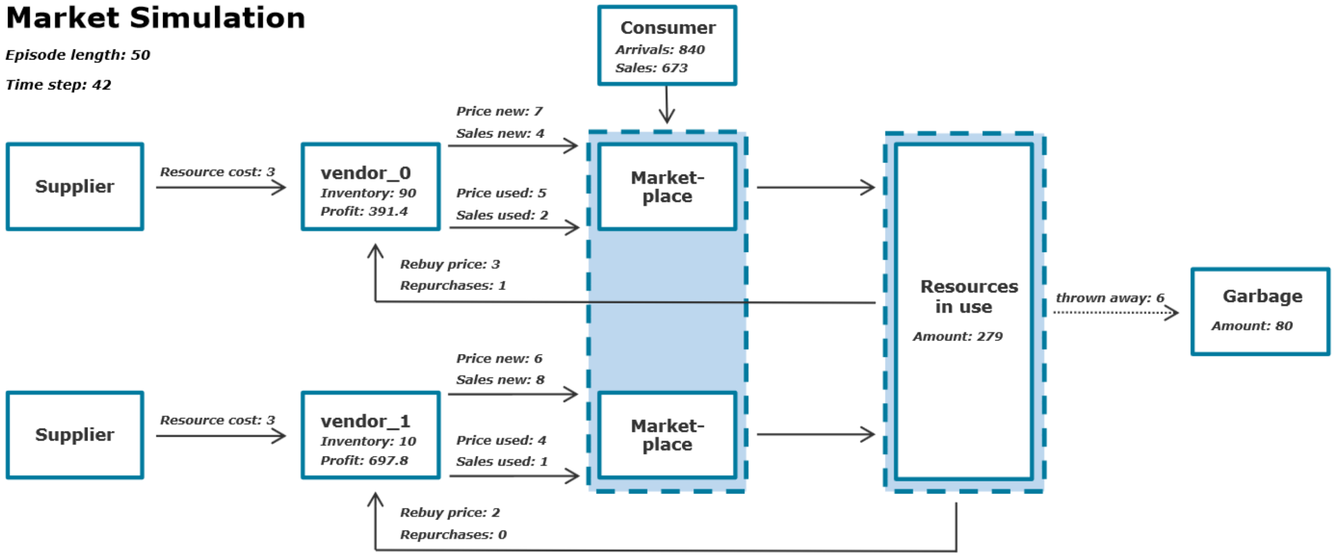
III Recommerce Markets are Growing

III Recommerce in Practice

III Recommerce Markets: Model Overview

Basis: A flexible simulation framework for **pricing agents**

Components: (i) Consumer, (ii) Firms, (iii) Marketplace, (iv) Resources in use



III Recommerce Model Description (What do we need?)

- Infinite horizon
- Discrete time (Periods)
- Duopoly competition (sequential updating of actions)

  Actions: Price new, price used, rebuy price

- Multiple consumer arrive (per period) in a certain way

  Buying behavior: Compare offers for new & used items

  Reselling behavior: Compare current rebuy prices

- Firms have individual inventory levels for used products

III MDP Formulation (Perspective of Firm 1)

- Discrete time $t = 0,1,\ldots,$ vs. periods $(t, t+1)$

- **Actions**: $p_{new}^{(1)} \in A,\ p_{used}^{(1)} \in A,\ p_{rebuy}^{(1)} \in A$ (**competitors update** within period)

- A single consumer's buying decision (cf. **Rewards**)

  Buying probabilities: $P_{no\ buy}^{(0)}(\vec{p}_{new}, \vec{p}_{used}) + \sum_{k=1,\ldots,K} P_{new}^{(k)}(\vec{p}_{new}, \vec{p}_{used}) + \sum_{k=1,\ldots,K} P_{used}^{(k)}(\vec{p}_{new}, \vec{p}_{used}) = 1$

- A single consumer's selling decision (cf. **Rewards**)

  Buying probabilities: $P_{no\ sell}^{(0)}(\vec{p}_{new}, \vec{p}_{used}, \vec{p}_{rebuy}) + \sum_{k=1,\ldots,K} P_{sell}^{(k)}(\vec{p}_{new}, \vec{p}_{used}, \vec{p}_{rebuy}) = 1$

- **Firm 1's state**: own inventory (#used), prices $\vec{p}_{new}, \vec{p}_{used}, \vec{p}_{rebuy}$

  (# resources in use, competitors' inventories)

III Objective: Max Expected Discounted Future Rewards

- Firm k's rewards from time $t = 0,1,\ldots$, on:

$$G_t^{(k)} := \sum_{i=t}^{\infty} \delta^{i-t} \cdot \left(\begin{array}{l} \underbrace{X_{new}^{(k)}(i) \cdot \left(p_{new}^{(k)}(i) - c_{virgin} \right)}_{rewards\ \ from\ sales\ new} + \underbrace{X_{used}^{(k)}(i) \cdot p_{used}^{(k)}(i)}_{rewards\ \ from\ sales\ used} \\ - \ \underbrace{N_{used}^{(k)}(i) \cdot c_{inv}}_{inventory\ \ holding\ \ costs} \ - \underbrace{X_{rebuy}^{(k)}(i) \cdot p_{rebuy}^{(k)}(i)}_{purchase\ \ costs} \end{array} \right)$$

- Objective: maximize $E\left(G_0^{(k)} \middle| \ s_0^{(k)} \right)$

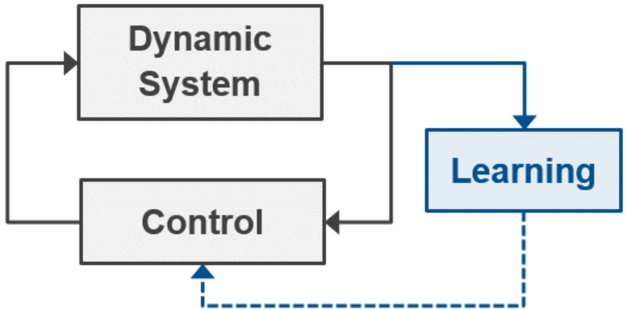
- Actions may depend on states: $s_t^{(k)} := \left(N_{used}^{(k)}(t), \vec{p}_{new}(t), \vec{p}_{used}(t), \vec{p}_{rebuy}(t) \right)$

III Solution Approach

- Dynamics known? (consumer behavior & competitors' reactions)?
- Explicitly estimate dynamics? Optimize afterwards?
- Are states observable? Number of states tractable?
- Dynamic programming methods applicable?
- Hope for analytical or closed-form solutions?
- Simplify setup?
- **Our approach: Apply & test RL techniques!**

III Self-Learning Approaches (Reinforcement Learning)

- Consider a dynamic system (MDP environment) unknown to the agent
- Observe current state
- Perform an action
- Receive a reward and the new state



- Exploration: Play different actions
- Update Value Function estimation
- Exploitation: Play in line with the Bellman Equation
- Simulate many runs/episodes
- Algorithms: QL, DQN, SAC, PPO
- Use of neural networks (to estimate V) allows for large state & action spaces

III Apply RL Algorithms (What do we need?)

- Play **actions** in the Recommerce **environment** (unknown to the agent)
  Observe realized **reward signals** and transition from old to **new state**

- Setup: Stationary, discrete time, infinite horizon
- Actions: Combinations of 3 own prices
- State space: Prices of all players + own inventory level

- Rewards: Define consumer behavior (arrival and decision)
- State transitions: Define competitors’ price response strategies

- Apply standard RL algorithms, e.g.: DQN, A2C, SAC, PPO

III Evaluation (Specific Rule-based Competitors)

$$p_{new}^{(k)}(N_{used}^{(k)}, \vec{p}_{new}, \vec{p}_{used}, \vec{p}_{rebuy}) := \max\left(\min_{i\in\{1,\dots,K\}\setminus\{k\}}\{p_{new}^{(i)}\} - h, c_{virgin} + h\right) \tag{6}$$

$$p_{used}^{(k)}(N_{used}^{(k)}, \vec{p}_{new}, \vec{p}_{used}, \vec{p}_{rebuy}) := \begin{cases} \min\limits_{i\in\{1,\dots,K\}\setminus\{k\}}\left\{p_{used}^{(i)}\right\} + h & , N_{used}^{(k)} < M/15 \\ \min\limits_{i\in\{1,\dots,K\}\setminus\{k\}}\left\{p_{used}^{(i)}\right\} - h & , N_{used}^{(k)} < M/8 \\ \min\limits_{i\in\{1,\dots,K\}\setminus\{k\}}\left\{p_{used}^{(i)}\right\} - 2h & , else \end{cases} \tag{7}$$

$$p_{rebuy}^{(k)}(N_{used}^{(k)}, \vec{p}_{new}, \vec{p}_{used}, \vec{p}_{rebuy}) := \begin{cases} \min\limits_{i\in\{1,\dots,K\}\setminus\{k\}}\left\{p_{rebuy}^{(i)}\right\} + h & , N_{used}^{(k)} < M/15 \\ \min\limits_{i\in\{1,\dots,K\}\setminus\{k\}}\left\{p_{rebuy}^{(i)}\right\} - h & , N_{used}^{(k)} < M/8 \\ \min\limits_{i\in\{1,\dots,K\}\setminus\{k\}}\left\{p_{rebuy}^{(i)}\right\} - 2h & , else \end{cases} \tag{8}$$

III Evaluation (Specific Consumer Behaviour)

To compare prices of different competitors we use the following two preference functions $u_{new}(p)$, $p \in A_{new}$, and $u_{new}(p)$, $p \in A_{used}$, defined by

$$u_{new}(p) := \frac{p_{new}^{(\max)}}{p} - e^{p - 0.8 \cdot p_{new}^{(\max)}}$$

and

$$u_{used}(p) := \frac{0.55 \cdot p_{used}^{(\max)}}{p} - e^{p - 0.5 \cdot p_{used}^{(\max)}}.$$

Based on the preference functions u_{new} and u_{used} and the fixed preference value of *one* associated to the *no buy* option, we use $\Sigma := e^1 + \sum_{i=1,...,K} e^{u_{new}(p_{new}^{(i)})} + \sum_{j=1,...,K} e^{u_{used}(p_{used}^{(j)})}$ and the softmax function to define $P_{no\ buy}^{(0)}$, $P_{new}^{(k)}$, and $P_{used}^{(k)}$, $k = 1, ..., K$, as:

$$P_{no\ buy}^{(0)}(\vec{p}_{new}, \vec{p}_{used}) := e/\Sigma \tag{9}$$

$$P_{new}^{(k)}(\vec{p}_{new}, \vec{p}_{used}) := e^{u_{new}(p_{new}^{(k)})}/\Sigma \tag{10}$$

$$P_{used}^{(k)}(\vec{p}_{new}, \vec{p}_{used}) := e^{u_{used}(p_{used}^{(k)})}/\Sigma. \tag{11}$$

III Evaluation (Model Parameters)

In the following examples and experiments, if not chosen differently, we use the parameters summarized in Table 1.

Symbol	Explanation	Default Value
K	number of competing firms	2
δ	discount factor per period	0.99
A	price sets $A_{new} = A_{used} = A_{rebuy} = A$	$[0, 10]$
$p^{(max)}$	maximum price for all three price sets A_{new}, A_{used}, A_{rebuy}	10
h	incremental price unit	1
c_{virgin}	Purchase or production price for new products	3
c_{inv}	Price per stored used product per period (step)	0.1
B	Number of customers visiting the store per step	$10/K$
M	Upper reference value for used products in stock	100
w	Proportion of owners considering resale per step	0.05
E	number of periods (steps) per episode	500

Table 1: Parameters with brief explanation and default values used for our experiments

III Experiment 1 (RL against a Rule-based Competitor)

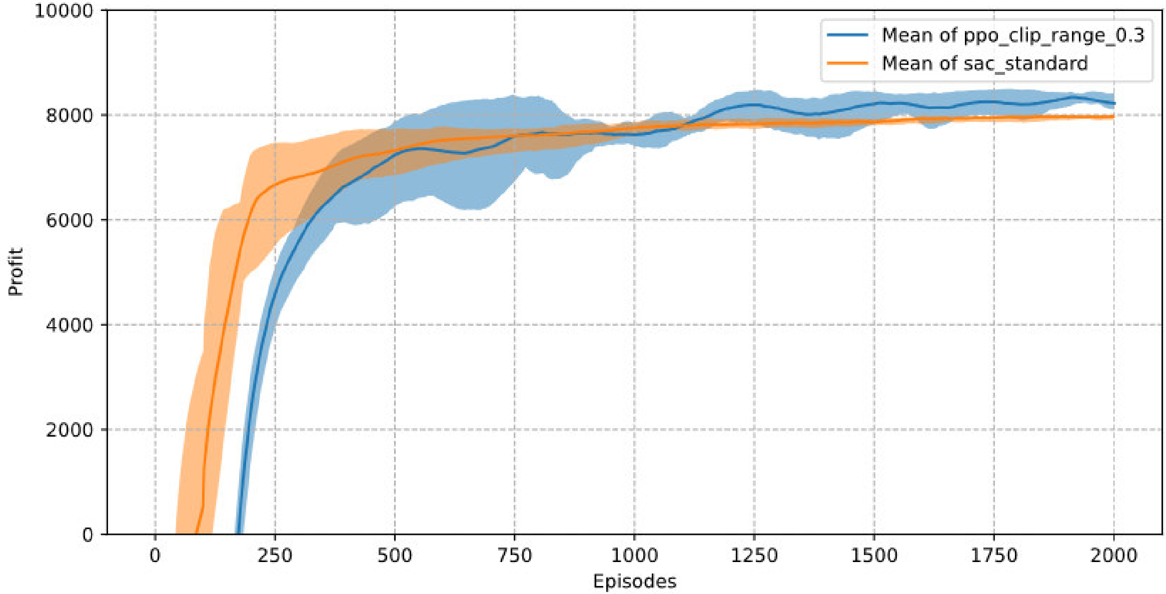


Figure 5: learning curves of four SAC runs in direct comparison with PPO

III Experiment 2 (RL against RL via Self-Play)

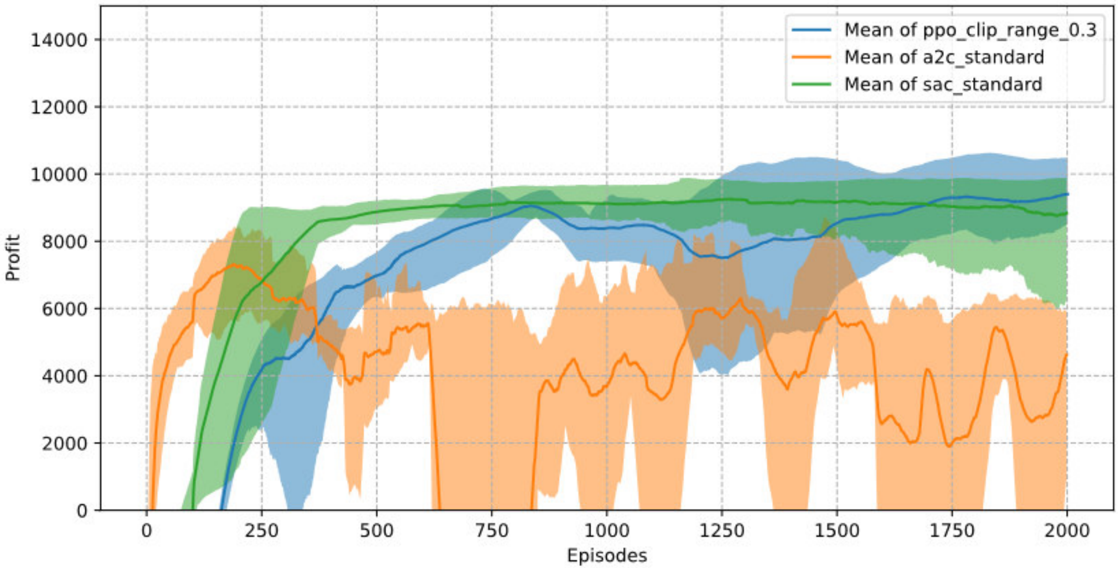


Figure 8: learning curve from four A2C, PPO, and SAC runs at Self-Play; algorithms were trained for 2000 episodes

III Experiment 3 (Ablation Study for Different State Spaces)

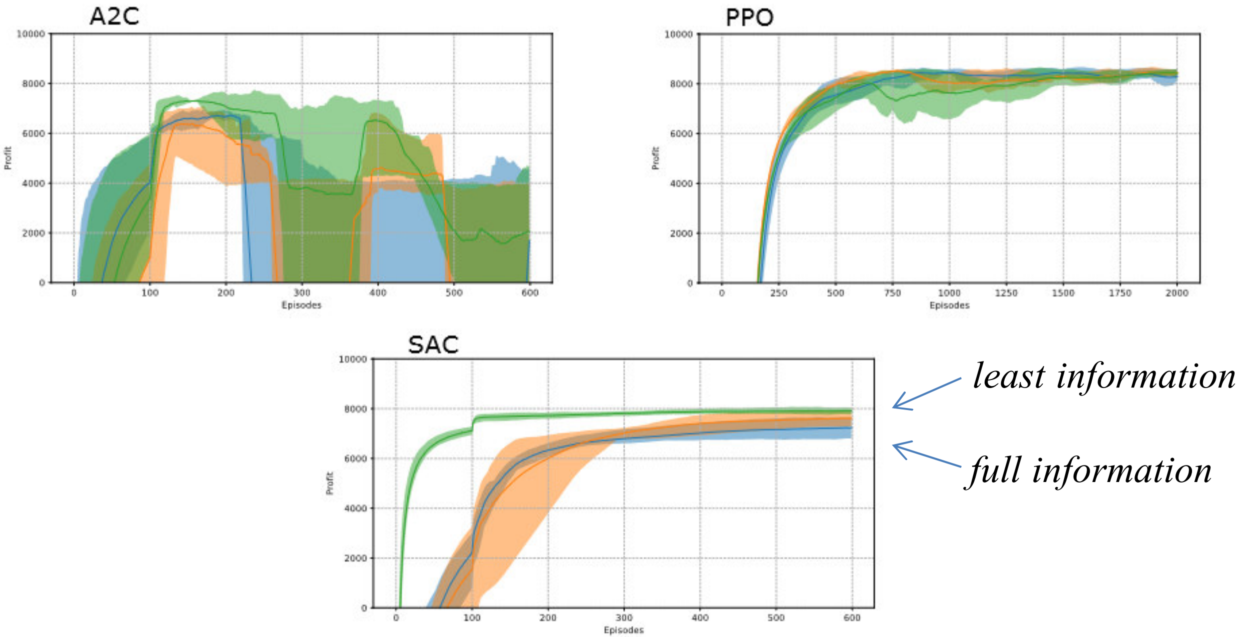


Figure 10: learning curves of A2C, PPO, and SAC with full versus partial observation; (blue) full observation, (orange) without competitor's stock level, and (green) without competitor's stock level and number of products.

III Experiment 4 (RL in an Oligopoly with 4 Rule-based Firms)

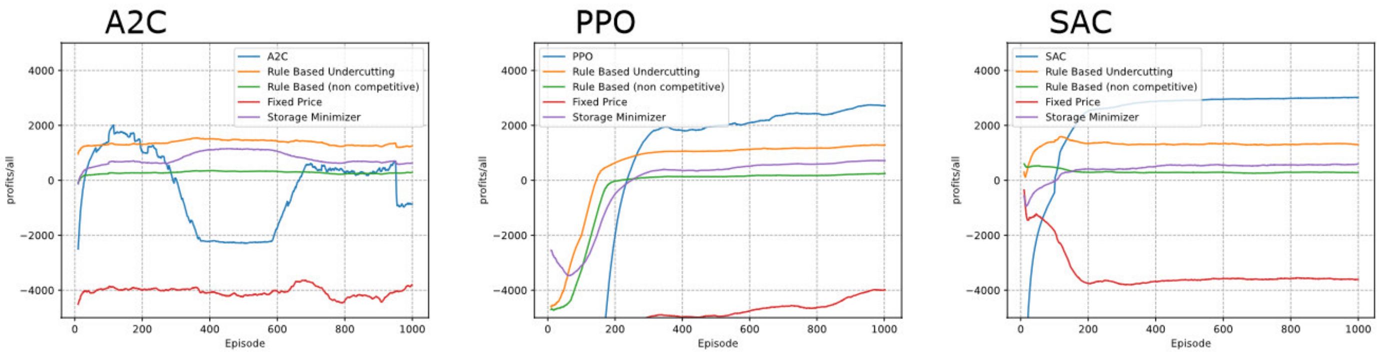


Figure 14: Profits of typical training runs of A2C, PPO, and SAC compared to their rule-based peers

III Self-Learning Pricing: Summary & Takeaways

(+) **State transitions** of the problem (MDP) **do not have to be known**

(+) **Larger** MDPs can be considered

(+) RL heuristics **outperform** rule-based benchmarks

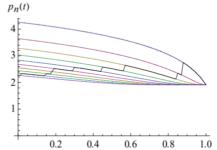
(–) Ordinary numerical results, no theoretical insights, no sensitivity results

(–) Environment has to be defined

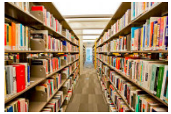
(–) Many training runs required (cf. online RL)

(+/–) Will we see the application of RL in practice? What’s next?

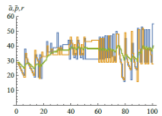
Summary & IV Future Research Directions

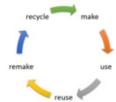


I Analytical solutions for Pricing



II Dynamic pricing applied in practice

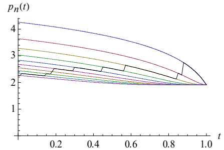




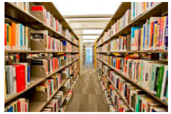
III Self-learning agents in Recommerce markets

IV
- Application in practice (fit environment from historical data?)
- Strategic consumers (reference prices, anticipate price patterns?)
- Analyze RL against RL (algorithmic collusion?)
- Trust black-box algorithms? Explainable AI? Hybrid approaches?

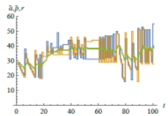
Summary & IV Future Research Directions

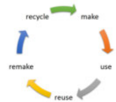


I Analytical solutions for Pricing



II Dynamic pricing applied in practice





III Self-learning agents in Recommerce markets

IV
- Application in practice (fit environment from historical data?)
- Strategic consumers (reference prices, anticipate price patterns?)
- Analyze RL against RL (algorithmic collusion?)
- Trust black-box algorithms? Explainable AI? Hybrid approaches?

Thank You!