PHANTOM

Pricing heuristics against non-human transaction orchestration

Daniel Rösel
IE University ▪ Supervisor: Alberto Martín Izquierdo

velocitatem.github.io/PHANTOM

Roadmap: one argument in six stages (15 min)

Intro 2m → Platform 4m → Signal 4m → DR-RL 4m → Results 1m → Close

Main research question

How can dynamic pricing preserve margin integrity when transactions are increasingly mediated by non-human agents?

Agentic recon creates direct financial pressure on pricing power

query and purchase are decoupled

Recon session
samples multiple quotes

→

Clean execution session
buys using the best found quote

The platform sees behavior proxy $\hat{q}$, while true demand response $d(p \mid \theta)$ stays latent.

$\mathrm{COI}(\pi) = \mathbb{E}[P] - \underline{p}$
pricing power KPI

$-9{,}014$
revenue units
per +0.1 contamination

$\lim_{N\to\infty} \mathrm{COI} = 0$
theorem-level pressure

Implication: if quote discovery and purchase split, standard session-based pricing overestimates willingness to pay.

The thesis answers one chain: mechanism → signal → control

1. **Mechanism (SQ2):** independent reconnaissance pushes realizable price toward the order-statistics floor.

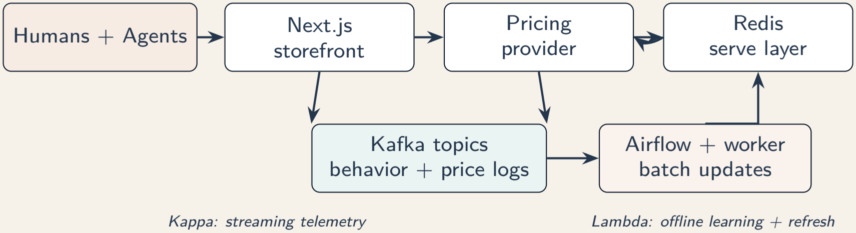
The thesis answers one chain: mechanism → signal → control

1. **Mechanism (SQ2):** independent reconnaissance pushes realizable price toward the order-statistics floor.
2. **Signal (SQ1):** human and agent sessions are behaviorally separable from trajectories alone.

The thesis answers one chain: mechanism $\rightarrow$ signal $\rightarrow$ control

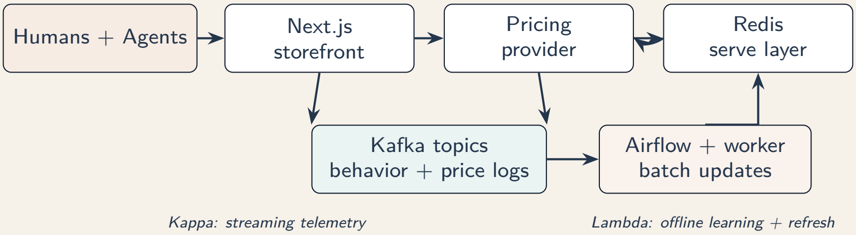
1. **Mechanism (SQ2):** independent reconnaissance pushes realizable price toward the order-statistics floor.
2. **Signal (SQ1):** human and agent sessions are behaviorally separable from trajectories alone.
3. **Control (SQ3):** the session score feeds a robust pricing learner under contamination uncertainty.

Stage 1: We built a dual-loop platform to observe behavior and price exposure together



- Every quote has a matching behavioral context in the log stream.

Stage 1: We built a dual-loop platform to observe behavior and price exposure together



- Every quote has a matching behavioral context in the log stream.
- The same architecture supports reproducible stress tests before any live deployment.

Dataset card: compact, labeled, and experiment-ready

WhoClickedIt dataset card

`huggingface.co/datasets/velocitatem/whoclickedit`

Rows 3874 **Cols** 42 **Sessions** 36

Human rows 798 **Agent rows** 3076

Flat schema and explicit actor labels simplify session-aware train/test splits.

Kafka provenance is retained for reproducibility and downstream analysis.

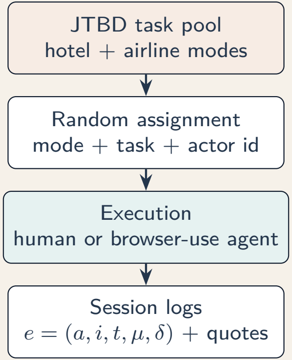
13 H / 16 A
labeled trajectories in thesis cohort

45% / 55%
human/agent trajectory split

2 streams
interaction + price-log records

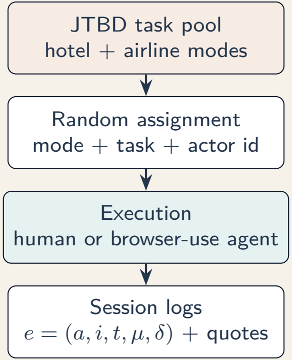
Use in practice: this card gives immediate cohort context before any modeling step.

Experimental design controls goals, not navigation paths



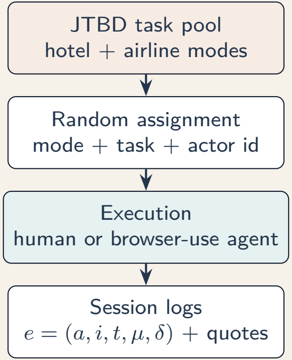
- Agents run with **browser-use** and a model-swappable LLM router (default `gpt-5-mini`).

Experimental design controls goals, not navigation paths



- Agents run with **browser-use** and a model-swappable LLM router (default `gpt-5-mini`).
- Tasks are defined by outcomes, not scripted clicks, to preserve behavioral variety.

Experimental design controls goals, not navigation paths



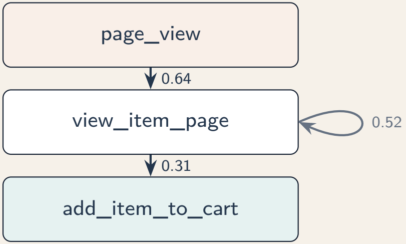
- Agents run with **browser-use** and a model-swappable LLM router (default `gpt-5-mini`).
- Tasks are defined by outcomes, not scripted clicks, to preserve behavioral variety.
- Current release is stronger on hotel flows than airline flows.

Stage 2: A behavior kernel is a compact signature of navigation dynamics

Definition

$$\hat{P}(s' \mid s) = \frac{N(s, s')}{\sum_k N(s, k)}$$

- Build one kernel per session, then prototypes for human and agent cohorts.



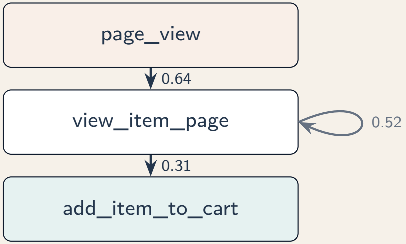
Kernel rows encode "what usually comes next."

Stage 2: A behavior kernel is a compact signature of navigation dynamics

Definition

$$\hat{P}(s' \mid s) = \frac{N(s, s')}{\sum_k N(s, k)}$$

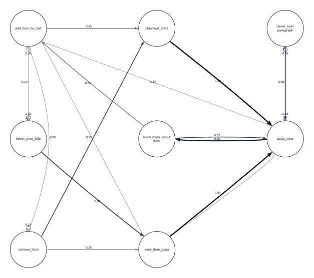
- Build one kernel per session, then prototypes for human and agent cohorts.
- Compare each incoming session to both prototypes with KL divergence.



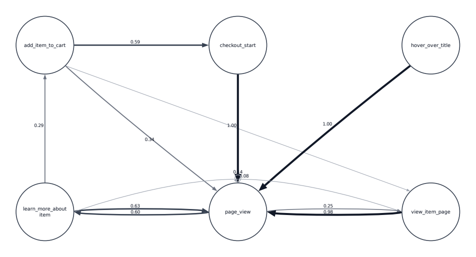
Kernel rows encode “what usually comes next.”

Human and agent kernels are separable in the controlled cohort

Human transition structure



Agent transition structure



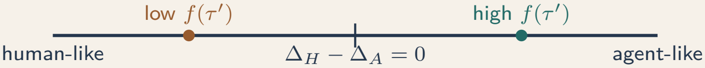
-3.35
mean gap (human)

+1.65
mean gap (agent)

$p < 0.001$
Mann-Whitney rank test

Two divergence scores become one continuous control signal

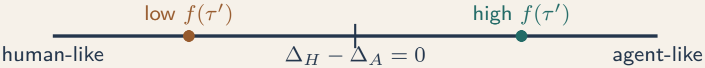
$$f(\tau') = P(A \mid \tau') = \sigma\left(\frac{\Delta_H - \Delta_A}{T}\right)$$



- Continuous scoring is used to steer contamination-aware pricing.

Two divergence scores become one continuous control signal

$$f(\tau') = P(A \mid \tau') = \sigma\left(\frac{\Delta_H - \Delta_A}{T}\right)$$



- Continuous scoring is used to steer contamination-aware pricing.
- The design target is guidance, not a hard user-level ban decision.

Stage 3: DR-RL trains against plausible contamination shifts, not one fixed world

Ideal robust object

$$\mathcal{U}_\epsilon(\hat{P}_N) = \{Q : W_p(Q, \hat{P}_N) \le \epsilon\}$$

robust against distribution shift around the empirical demand law

Engine approximation used in experiments

$$\mathcal{A}_{\epsilon_\alpha}(\alpha_0) = \{\alpha : |\alpha - \alpha_0| \le \epsilon_\alpha\}$$

small grid over α $\rightarrow$ inner worst-case candidate

Practical boundary

In code we solve a local robust loop around α_0, not the full continuous Wasserstein adversary.

Reward composition penalizes leakage while guarding user experience

$$r_t = \underline{R(p_t, \hat{Q}_t)} - \underline{\lambda\, f(\tau_t')\, c_{\text{info}}} - \underline{\eta_{\text{ux}}\, UX(\tau_t', p_t)}$$

Revenue term
keeps market objective explicit

Leakage term
scales with agent-likelihood score

UX term
discourages unstable pricing behavior

- Baseline experiments use a query-tax leakage surrogate for tractability.

Reward composition penalizes leakage while guarding user experience

$$r_t = \underline{R(p_t, \hat{Q}_t)} - \underline{\lambda\, f(\tau_t')\, c_{\text{info}}} - \underline{\eta_{\text{ux}}\, UX(\tau_t', p_t)}$$

Revenue term keeps market objective explicit	**Leakage term** scales with agent-likelihood score	**UX term** discourages unstable pricing behavior

- Baseline experiments use a query-tax leakage surrogate for tractability.
- Supra-competitive anchor penalties are tracked as an additional safety rail.

Computationally, wide sweeps are feasible only with aggressive optimization

$$4 \times 4 \times 3 \times 2 \times 2 = \mathbf{192}$$

algorithms × contamination × robustness × COI penalty × action grid

160 PFLOPS
peak aggregate TPU budget

~180 days
net compute logged in full study

Hot-path rewrite impact

Mode	Before	After
Baseline step/s	26.0	220.0
Robust step/s	7.2	136.0

- pandas lookup bottlenecks replaced with array/JAX-style loops.

Computationally, wide sweeps are feasible only with aggressive optimization

$$4 \times 4 \times 3 \times 2 \times 2 = \mathbf{192}$$

algorithms × contamination × robustness × COI penalty × action grid

160 PFLOPS
peak aggregate TPU budget

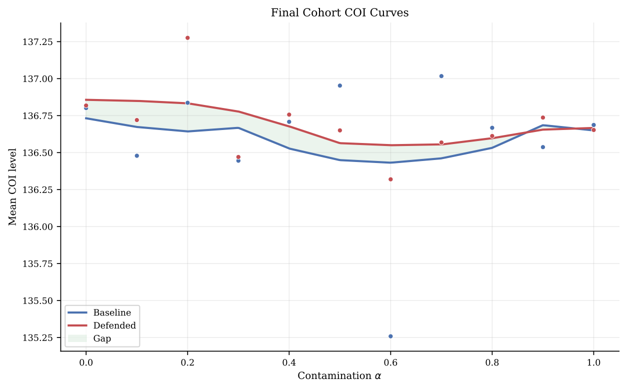
~180 days
net compute logged in full study

Hot-path rewrite impact

Mode	Before	After
Baseline step/s	26.0	220.0
Robust step/s	7.2	136.0

- pandas lookup bottlenecks replaced with array/JAX-style loops.
- Throughput gains (8.5×, 19×) made broad sweeps practical.

Results: contamination hurts revenue; defended policies recover COI



-90,140
baseline contamination slope

~3%
short-run revenue cost of defense

Regime-dependent
COI gains strongest at harder settings

Yes, with boundaries: we can defend margin integrity under agentic orchestration

SQ1 Distinguishability	SQ2 Theoretical impact	SQ3 Mitigation
kernels are separable $p < 0.001$	COI erosion mechanism proved in baseline limit	robust control shifts COI/revenue/UX trade-off

Boundary conditions

Evidence is from a controlled platform and a small labeled cohort; this is mechanism validation, not full production external validity.

What this implies for real pricing systems

- **Financially:** untreated reconnaissance behaves like an information leak and can compress sustainable margins.

What this implies for real pricing systems

- **Financially:** untreated reconnaissance behaves like an information leak and can compress sustainable margins.
- **Operationally:** behavior-only session scoring can be wired into pricing without relying on device fingerprinting.

What this implies for real pricing systems

- **Financially:** untreated reconnaissance behaves like an information leak and can compress sustainable margins.
- **Operationally:** behavior-only session scoring can be wired into pricing without relying on device fingerprinting.
- **Strategically:** robust pricing should be calibrated by regime; there is no single penalty that wins everywhere.

What this implies for real pricing systems

- **Financially:** untreated reconnaissance behaves like an information leak and can compress sustainable margins.
- **Operationally:** behavior-only session scoring can be wired into pricing without relying on device fingerprinting.
- **Strategically:** robust pricing should be calibrated by regime; there is no single penalty that wins everywhere.
- **Before deployment:** larger human baselines, governance review, and legal safeguards are mandatory.

Thank you

Questions and discussion

Appendix follows: COI theorem derivation, reward composition, and sample-size notes.

Appendix roadmap

A. Objects

Notation, COI, proxies

B. Mechanism

Order stats, kernels, KL

C. Control

Simulator, robust loop, factorial grid

Figures

Full charts, MDPs, extra revenue view

Appendix: core notation (quick reference, I)

$$\tau_s = (e_{s,1}, \ldots, e_{s,L_s})$$ session

$$\hat{q}_{t,i} = \sum_{s \in S_t} \sum_{k} \omega(a_{s,k})\, \mathbf{1}[i_{s,k} = i]$$ proxy

$$Q(p) = (1-\alpha)\, \mathbb{E}_{\theta \sim D_H}[d(p;\theta)] + \alpha\, \mathbb{E}_{\theta \sim D_A}[d(p;\theta)] + \epsilon_t$$ mixture

$$\mathrm{COI}(\pi) = \mathbb{E}[P] - \underline{p}$$ COI

Appendix: core notation (quick reference, II)

- $\underline{p}$: minimum viable price anchor (thesis simplification).
- α: contamination with agent traffic in the mixture.
- $\omega(a)$: hand-engineered action weights for the proxy (baseline).

Reading guide

Objects on the left are **observable**; $d(\cdot)$ and many θ remain hidden.

Appendix: COI as a reporting functional

$$\mathrm{COI}(\pi) = \mathbb{E}_{P \sim F_\pi}[P] - \underline{p}$$

Interpretation

Premium above the floor induced by policy π; used as a KPI and as the object Theorem 1 attacks under query saturation.

Appendix: demand proxy vs. latent demand

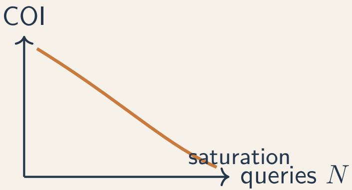
$$\hat{q}_{t,i} = \sum_{s \in S_t} \sum_{k=1}^{L_s} \omega(a_{s,k}) \, \mathbf{1}[i_{s,k} = i]$$

Key distinction

$\hat{q}$ is an operational sensor from logs; true demand $d(p;\theta)$ stays latent. Pricing reacts to $\hat{q}$, so agent-shaped behavior poisons the signal.

Appendix: independent draws and order statistics (intuition)

- Independent price draws $\{P_i\}_{i=1}^{N}$ from fixed offer law.
- Purchase-side minimum behaves like $P_{(1)}$: mass shifts left as N grows.
- Expected premium vs. $\underline{p}$ compresses: COI pressure.



Appendix: Theorem 1 scope (what is and is not claimed)

Inside the baseline proof

Non-collusive sessions, independent draws, fixed offer distribution across queries.

Outside (handled elsewhere)

Collusion, pooled recon, sequential repricing that breaks iid structure: evidence moves to the simulator.

Appendix: empirical transition kernel (MLE)

$$\hat{P}(s' \mid s) = \frac{N(s, s')}{\sum_k N(s, k)}$$

Use

Human and agent centroids $\bar{T}_H, \bar{T}_A$ for divergence-to-prototype scores.

Appendix: KL to prototypes (shared support)

$$\Delta_H = D_{\mathrm{KL}}(\hat{T}' \,\|\, \bar{T}_H), \qquad \Delta_A = D_{\mathrm{KL}}(\hat{T}' \,\|\, \bar{T}_A)$$

Asymmetric choice

KL measures deviation from the **human** reference; symmetric JS/Wasserstein on behavior was not the design target.

Appendix: softmax to sigmoid (algebra)

Let $z_A = -\Delta_A/T$, $z_H = -\Delta_H/T$. Then

$$P(A \mid \tau) = \frac{e^{z_A}}{e^{z_A} + e^{z_H}} = \frac{1}{1 + e^{z_H - z_A}} = \sigma\big(z_A - z_H\big)$$
$$= \sigma\left(\frac{\Delta_H - \Delta_A}{T}\right).$$

Takeaway

Two-class softmax over (z_A, z_H) is exactly one sigmoid on the gap $(\Delta_H - \Delta_A)$.

Appendix: contamination generator $\mathcal{G}(\alpha)$

$$\mathcal{G}(\alpha): \text{ inject synthetic agent trajectories until mixture reaches target } \alpha$$

Role in the lab

Supplies controlled stress tests for the pricing learner; not a claim of production-faithful agents.

Appendix: Wasserstein ambiguity (ideal object)

$$\mathcal{U}_\epsilon(\hat{P}_N) = \left\{ Q : \; W_p(Q, \hat{P}_N) \le \epsilon \right\}$$

What the code implements instead

A **local** grid over α near α_0 with radius ϵ_α: tractable inner worst case, not a full ball solver.

Appendix: per-step reward sketch

$$r = R(p, d) - \lambda\, \mathrm{COI}_{\mathrm{leak}}(p, \tau') - \eta\, \mathrm{UX}(\tau', p) - \text{(supra-competitive excess)}$$

- Query-tax style $\mathrm{COI}_{\mathrm{leak}}$: minimal nonzero surrogate to expose the control channel.
- UX and anchor penalties prevent trivial solutions (flat but exploitative prices).

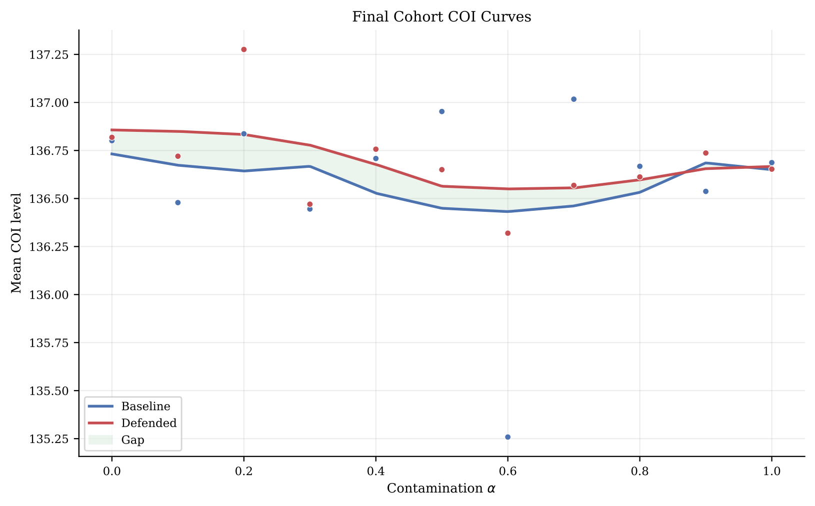
Appendix: factorial design (192 cells)

Axis	Levels	Count
RL algorithm	PPO, A2C, DQN, Q-table	4
Contamination α	4 representative values in $[0.1, 0.6]$	4
Robustness radius ϵ_α	3	3
COI penalty λ_{coi}	2	2
Action granularity	2	2
Total		$4 \times 4 \times 3 \times 2 \times 2 = \mathbf{192}$

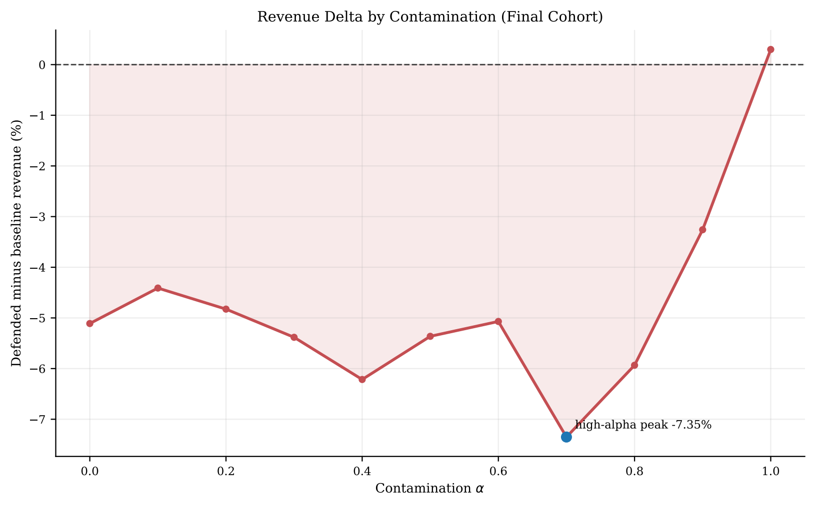
Appendix: engineering note (pandas $\rightarrow$ JAX)

- Hot path was label-indexed transition lookups; profiling showed pandas overhead dominated.
- Integer-indexed arrays + JAX inner loop: large step/s throughput (thesis numbers; environment dependent).
- Kronecker expansion of product-conditioned kernels: research simulator cost, scales with catalog.

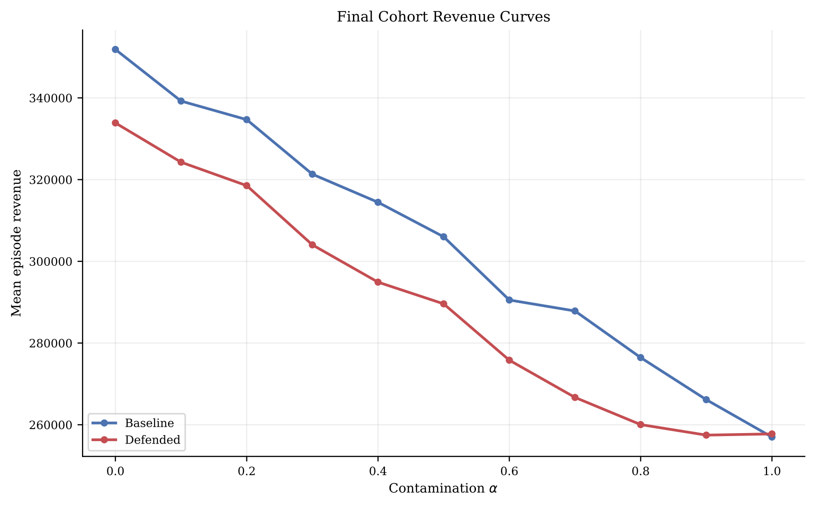
Appendix figure: COI by α (full)



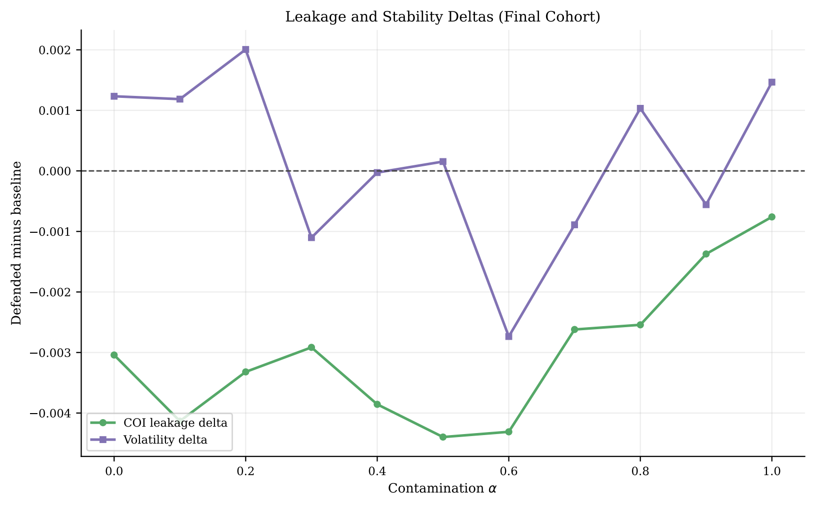
Appendix figure: revenue deltas (full)



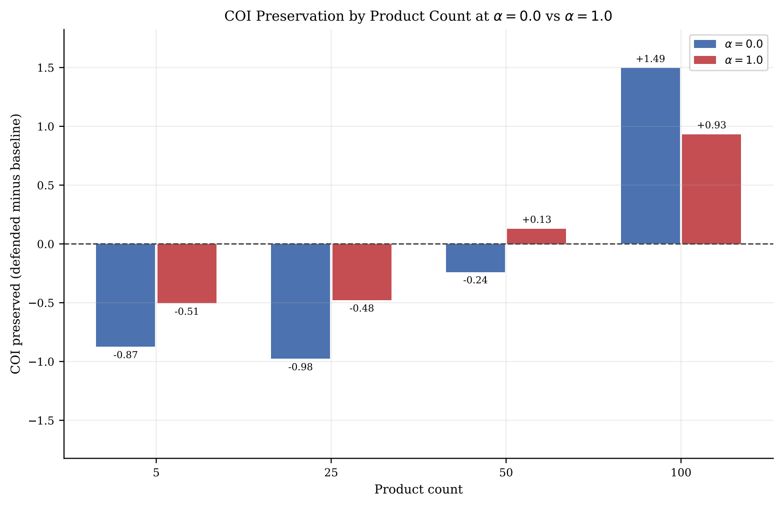
Appendix figure: revenue by α (full)



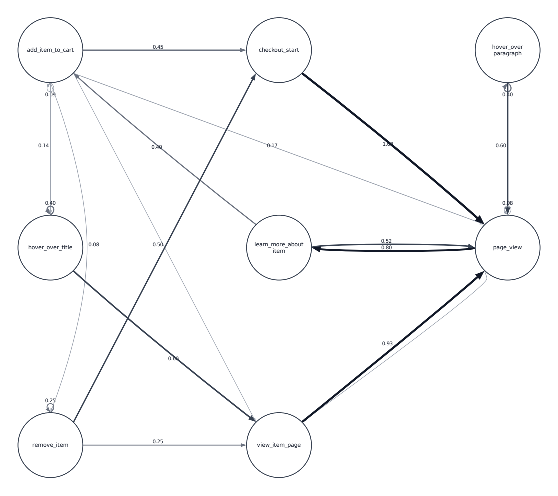
Appendix figure: risk / stability deltas (full)



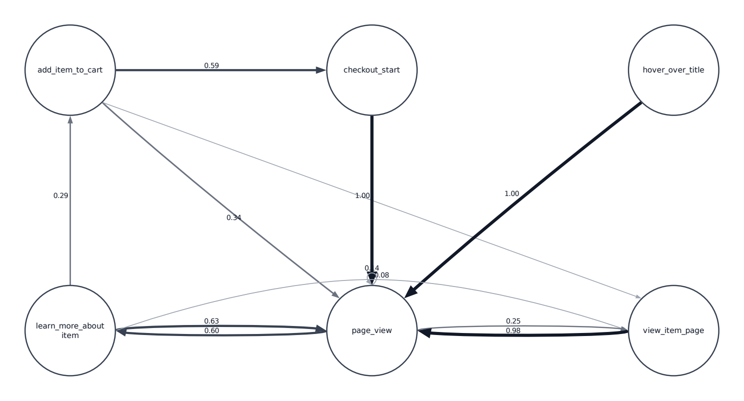
Appendix figure: COI preservation grid (full)



Appendix figure: human MDP (full)



Appendix figure: agent MDP (full)



Appendix: threat model map

Claim boundary

Residual contamination after security controls is the motivating scenario.

Appendix: evaluation checklist (robustness culture)

1. Session-aware labels: avoid splitting rows inside a trajectory if that inflates scores.
2. Document how prototypes $\bar{T}_H, \bar{T}_A$ were fit (full cohort vs. held-out); state explicitly in writing.
3. Report temperature T as calibration, not as a tuned hyperparameter unless a sweep is shown.
4. Separate **architecture** claims from **coverage** claims (hotel vs. airline balance at release).

Appendix: sim-to-real gap (explicit)

- Kernels and generators reflect a **small labeled cohort** and a **browser-use style** agent class.
- RL policies are trained in a **surrogate** market with engineered rewards and discretized prices.
- Deployment would require legal review, fairness testing, and refreshed baselines at scale.

Appendix: leakage surrogate (query-tax form)

$$\mathrm{COI}_{\mathrm{leak}}(p, \tau') \approx f(\tau') \cdot c_{\mathrm{info}}$$

Reading

$f(\tau')$ is the weak agent score; c_{info} is a minimal constant leakage proxy to expose the control channel. Revelation-style $-\log \pi(p \mid \tau')$ is the natural upgrade.

Appendix: robust pricing template (symbolic)

$$\max_{\pi} \min_{Q \in \mathcal{U}_{\epsilon}(\hat{P}_N)} \mathbb{E}_{d \sim Q}\left[R(p, d) - \lambda\, \mathrm{COI}_{\mathrm{leak}} - \eta\, \mathrm{UX}\right]$$

Code-level substitute

Inner min over a **finite grid** of $\alpha_k \in [\alpha_0 \pm \epsilon_\alpha]$ around the nominal generator mix, not a continuous adversary over all Q in the ball.

Appendix: Stackelberg timing (words)

- Leader: platform sets price vector given current state and policy.
- Follower: demand proxy updates from simulated trajectories drawn from $\mathcal{G}(\alpha)$ and kernels $(\hat{T}_H, \hat{T}_A)$.
- **Limbo** buffer stores alternating moves for a clean game history; relaxing strict alternation is listed future work.

Appendix: three layers of evidence

Theorem 1 Formal COI erosion under independence and fixed-offer assumptions.

Simulator Dynamic, adaptive pricing and contamination sweeps (different status).

Implementation Local-α robust training; spirit of DRO without claiming a full numerical Wasserstein solver.

Appendix: composite strip (five plots, small multiples)

Same PDFs as the main talk, shrunk to scan the full panel at once.