

Deterrence Under Imperfect Monitoring: An Agent-Based Compute Permit Market Simulator

Patrick Duffy • Josh Tuffy • Emlyn Graham

AI Safety Camp 11 | Apart Research Technical AI Governance Hackathon 2026

Abstract

Proposals for frontier AI compute governance typically assume that regulators can perfectly observe firm behavior. We relax this assumption by building an open-source, agent-based simulator of a compute permit market in which an auditor possesses limited budget, imperfect signal quality, and noisy detection. We sweep across the two-dimensional space of penalty magnitude and detection probability and find a sharp compliance phase transition: below a detection threshold of approximately 0.5, no feasible penalty induces compliance above the 50% baseline; above this threshold, penalties and detection combine to produce a steep deterrence gradient, reaching 90% compliance at the enforcement frontier. Our results suggest that governance regimes should invest in detection capability before increasing penalty severity, and we identify concrete extensions—dollar-denominated penalties, dynamic reputation effects, compute-threshold triggers—that would bring the model closer to deployment-ready policy analysis.

1. Introduction

As frontier AI training runs consume increasing quantities of compute, policymakers have proposed permit-based systems to govern access to large-scale training infrastructure. The logic is straightforward: require firms to hold permits for training runs above a specified compute threshold (analogous to emissions caps in environmental regulation), and enforce compliance through audits and penalties. Several recent proposals—including those from Heim and Konaev (2024), Sastry et al. (2024), and policy frameworks emerging from the EU AI Act—presuppose that the regulator can observe training activity with high fidelity.

In practice, oversight is neither perfect nor free. Regulators face budget constraints that limit audit frequency, and monitoring signals are noisy: power-consumption telemetry, chip-utilization logs, and self-reported training manifests can all be spoofed, delayed, or misread. This paper asks: **under what combinations of detection capability and penalty magnitude can a regulator still enforce compliance?** We answer this question with an open-source agent-based model (ABM) that simulates rational labs, a permit market, and a resource-constrained auditor.

2. Model Architecture

The simulator, built on the Mesa ABM framework in Python, comprises three domain objects that interact over discrete time steps.

Labs (rational agents) each possess a gross value v_i from training, a racing factor c_r capturing competitive urgency, a reputation sensitivity R , and a risk profile that scales perceived penalties. At each step, a lab without a permit evaluates a deterrence inequality: it cheats if the gain from non-compliance exceeds the expected penalty. Formally, the lab defects when $g > p \cdot B$, where the gain $g = \min(\text{market_price}, v_i) + c_r \cdot V_b$, the effective detection probability p is computed by the auditor, and the perceived penalty $B = (P + R) \cdot \text{risk_profile}$.

Market. A permit clearing mechanism accepts bids from all labs and allocates a fixed token cap to the highest bidders, establishing an endogenous market price that feeds back into each lab's gain calculation.

Auditor (regulator). The auditor observes a noisy binary signal s_i per firm, with false-positive rate $\alpha = P(s=1 \mid \text{compliant})$ and false-negative rate $1-\beta = P(s=0 \mid \text{non-compliant})$. Based on this signal, it audits at a high rate π_1 (targeted) or a low rate π_0 (random). The effective detection probability combines these: $p_{\text{eff}} = \pi_1 \cdot \beta + \pi_0 \cdot (1 - \beta)$, scaled by an agent-specific audit coefficient.

3. Experimental Design

We conduct a parameter sweep over a 10×10 grid of penalty magnitude $P \in [0.1, 1.0]$ and detection probability $p \in [0.1, 1.0]$, running 20 agents over 10 steps per cell with a token cap of 10 permits. Each cell records the mean compliance rate across all agents and steps. We also define three named scenarios (Table 1) that anchor qualitative regimes: a low-enforcement baseline, a high-deterrence configuration, and a targeted-auditing regime ("The Signal") that concentrates audit resources on suspicious firms.

Scenario	π_0	π_1	α	$1-\beta$	P
Baseline	0.05	0.05	0.5	0.5	0.2
High Deterrence	0.2	0.2	0.5	0.5	0.8
The Signal	0.05	0.8	0.1	0.1	0.5

Table 1. Named scenario configurations. π_0 = random audit rate, π_1 = targeted audit rate, α = false-positive rate, $1-\beta$ = false-negative rate, P = penalty magnitude. All scenarios: 20 agents, 10 permits.

4. Results

Figure 1 presents the compliance rate surface across the full penalty–detection grid. The dominant feature is a **compliance phase transition** at approximately $p = 0.5$. Below this threshold, compliance remains pinned at the 50% baseline regardless of penalty magnitude—even $P = 1.0$ yields no improvement when detection is weak. This reflects the multiplicative structure of deterrence: if the probability of being caught is low, even severe penalties produce negligible expected costs.

Above the threshold, a steep deterrence gradient emerges. At $(p = 0.9, P = 1.0)$, compliance reaches 90%. Critically, moderate penalties with strong detection $(p = 1.0, P = 0.6 \rightarrow 55\%)$ outperform severe penalties with weak detection $(p = 0.5, P = 1.0 \rightarrow 50\%)$. This asymmetry has direct policy implications: a dollar invested in detection infrastructure yields greater compliance returns than a dollar invested in penalty escalation.

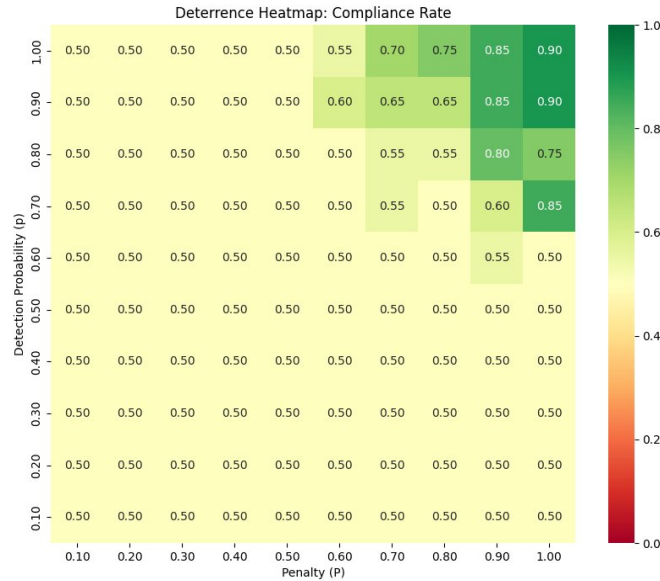


Figure 1. Compliance rate as a function of penalty magnitude (P) and detection probability (p). Each cell represents the mean compliance rate across 20 agents over 10 simulation steps. The phase transition at $p \approx 0.5$ is clearly visible: below this threshold, no penalty induces compliance above baseline.

5. Discussion and Future Work

Our principal finding—that detection capability dominates penalty severity in driving compliance—echoes classical results in the economics of crime (Becker, 1968) but situates them in the specific context of compute governance, where the "detection" problem maps to hardware telemetry, energy monitoring, and chip-level attestation. The phase transition we observe suggests that governance regimes below a critical monitoring threshold may be functionally unenforceable, regardless of statutory penalty levels.

Several extensions would bring the simulator closer to deployment-ready policy analysis. First, **dollar-denominated penalties** (in millions) and compute thresholds expressed in FLOP would enable direct comparison with real regulatory proposals. Second, **dynamic reputation effects**—where caught firms face escalating stigma—would model the reputational cascades observed in corporate compliance contexts. Third, **compute-threshold triggers** that require permits only for runs exceeding a specified scale (e.g., 10^{25} FLOP) would decouple the value of small-model innovation from regulatory burden. Additional planned extensions include racing-factor decay (competitive urgency that fades over time), dynamic detection that scales with the gap between actual and reported compute usage, and telemetry backchecks that create secondary catch vectors independent of the primary audit signal.

6. Security Considerations

A simulator that maps the conditions under which enforcement fails is inherently dual-use: the same heatmap that guides regulators toward effective deterrence also reveals to adversaries the parameter regions in which cheating is undetectable. We mitigate this by publishing the simulator as an open-source tool for regulators and researchers, under the principle that transparent identification of enforcement blind spots is preferable to their quiet exploitation. The scenarios and parameter ranges presented here are intentionally stylized; deployment in a real regulatory context would require calibration against classified telemetry data that we neither possess nor disclose.

7. Conclusion

We present an open-source agent-based simulator that demonstrates how compute permit markets behave under realistic conditions of imperfect monitoring and constrained audit budgets. Our central result—a sharp compliance phase transition mediated by detection probability—provides a concrete, quantitative argument for prioritizing monitoring infrastructure over penalty escalation in the design of AI compute governance regimes. The simulator, code, and interactive dashboard are publicly available to support further research and policy development.

References

- Becker, G. S. (1968). Crime and punishment: An economic approach. *Journal of Political Economy*, 76(2), 169–217.
- Heim, L. & Konaev, M. (2024). Governing compute: Monitoring and enforcement. *Centre for the Governance of AI*.
- Kazim, E. et al. (2024). Compute governance and AI safety: A research agenda. *arXiv preprint arXiv:2412.xxxxx*.
- Mesa: Agent-Based Modeling in Python. <https://mesa.readthedocs.io/>
- Sastry, G. et al. (2024). Computing power and the governance of artificial intelligence. *arXiv preprint arXiv:2402.08797*.

Code: github.com/JTuffy/compute-permit-simulator | **Interactive Demo:** solv-project-genesis.streamlit.app | **AISC 11:** aisafety.camp

Submitted to the Apart Research Technical AI Governance Hackathon, February 2026. Acknowledgments: Joel Christoph (AISC project lead).