Verification Evidence for Compute Governance

What Inspection Regimes Actually Detect

Supplementary proposal, SPAR Fall 2026

Joel Christoph and Jonas Kgomo

The gap

Compute governance proposals rest on a detection probability that nobody has estimated. Training-compute thresholds, licensing regimes and tradable permit schemes all require a regulator to observe declared compute and to catch under-reporting with some frequency. The technical literature sets out what could be built. It does not say how well any of it would work against a party that is optimising against it.

Formal treatments, including our own (SSRN 6361077), leave the detection probability as a free parameter. Referees have said so directly. That is the gap this project closes, using the only mature precedent for verifying a dual-use industrial input at scale.

Coding scheme

Each inspection instrument becomes one row. Every populated cell carries a citation and a confidence label (high, medium, low, absent). The scheme is piloted on five instruments in week three and then held fixed.

Field	Content
Instrument	Name and regime of origin
Object verified	The quantity or state the instrument is meant to establish
Detection mechanism	Physical or informational basis of detection
Documented performance	Published detection goals, measurement uncertainty, error rates
Inspection effort	Person-days, cost, or equivalent published measure
Intrusiveness	Access required, and the confidentiality objections raised
Evasion history	Documented cases where the instrument failed or was defeated
Compute analogue	Nearest counterpart in a compute governance regime
Transfer verdict	Transfers, transfers with modification, or fails, with reasons
Confidence	Strength of the evidence behind the row

Worked example

Instrument. Nuclear material accountancy with inspector verification measurement.
Object verified. That the declared inventory of fissile material at a facility matches the physical inventory, within a stated uncertainty.
Detection mechanism. Material balance closure over an accounting period, with destructive and non-destructive assay of samples.
Documented performance. Detection goals are defined against a significant quantity of 8 kg of plutonium, with a timeliness goal of one month for unirradiated direct-use material. Measurement uncertainty is published per assay method.
Evasion history. Repeated failure to detect undeclared facilities before the Additional Protocol, most visibly in Iraq before 1991.
Compute analogue. Periodic reconciliation of a developer's declared training compute against accelerator inventory and cloud provider records.

Transfer verdict. Transfers with modification, and the modification is severe. Accountancy presumes a conserved stock that must be somewhere. Compute is a flow: it is consumed, it can be bought in fragments across jurisdictions, and there is no residue to weigh afterwards. What survives is the auditing logic over declared quantities. What fails is the closure guarantee, since no compute analogue of a material balance exists. This is precisely the kind of finding the project is built to produce and the kind currently asserted without support in the compute literature.
Confidence. High on the safeguards side, low on the compute side, which is the finding.

From evidence to design

Extracted detection probabilities and effort estimates are fed into an existing compliance model with two margins of misreporting, under-declared compute and manipulable evaluation signals, and into an agent-based stress test with noisy metering, evasion and collusion. Both the model and the codebase exist and are handed to the team in week one. The reported result is a mapping from audit frequency, penalty schedule and certification structure to whether truthful reporting survives, and at what public cost.

Twelve weeks

1. Weeks 1–3. Shared orientation. Coding scheme built jointly and piloted on five instruments.
2. Weeks 4–7. Evidence extraction across nuclear safeguards and compute verification sources. Weekly written memos.
3. Weeks 8–10. Parameterisation, simulation and robustness checks.
4. Weeks 11–12. Writing. Complete draft by Demo Day.

Outputs

An openly licensed coded dataset with sourced cells and confidence labels. A paper for a governance workshop or policy journal, mentees co-authors by contribution. A short note for practitioners on which verification architectures the evidence supports.

Selected sources

- Aarne, O., Fist, T. and Withers, C. (2024). *Secure, Governable Chips.* CNAS, 8 January. cnas.org
- Baker, M. (2023). *Nuclear Arms Control Verification and Lessons for AI Treaties.* arXiv:2304.04123. arxiv.org/abs/2304.04123
- Brass, A. and Aarne, O. (2024). *Location Verification for AI Chips.* IAPS. iaps.ai
- Kulp, G. et al. (2024). *Hardware-Enabled Governance Mechanisms.* RAND WRA3056-1. rand.org
- Sastry, G. et al. (2024). *Computing Power and the Governance of Artificial Intelligence.* arXiv:2402.08797. arxiv.org/abs/2402.08797
- Shavit, Y. (2023). *What Does It Take to Catch a Chinchilla?* arXiv:2303.11341. arxiv.org/abs/2303.11341
- Wasil, A. et al. (2024). *Verification Methods for International AI Agreements.* arXiv:2408.16074. arxiv.org/abs/2408.16074