MADS-CPS: A Machine-Checkable Admissibility Contract for AI Scientists in Autonomous Laboratories

Anonymous Authors[1]

Abstract

AI scientists are moving from assistive tools toward systems that can generate hypotheses, orchestrate tools, interpret intermediate evidence, and in some domains close the loop with experimental execution. Existing evaluations increasingly measure capability, but capability alone does not tell us when one concrete AI scientist run should count as trustworthy enough to inspect, replay, or authorize. In autonomous laboratories, that distinction matters because planning, sensing, tool use, coordination, and irreversible actions are coupled at the level of a single run. We introduce *MADS-CPS*, a machine-checkable run-level admissibility contract that specifies a declared assurance envelope, a conformance checker over required artifacts and replay status, verification modes for restricted auditability, and fail-closed point-of-no-return release gating. We instantiate the framework in a robot-centric autonomous-laboratory profile and evaluate it through an eight-case conformance challenge corpus, a verification-mode admissibility matrix, an independent replay-link experiment, and a controller-matrix study spanning baseline and stressed regimes. Across these studies, MADS-CPS achieves perfect checker agreement on injected faults, strong replay match rate 1.00 in the reported E3 settings, raw controller invariance in baseline settings, normalized-interface invariance in baseline and moderate settings, and interpretable controller divergence under harder coordination stress. These results suggest that run-level admissibility can remain machine-checkable even when productivity and controller behavior separate under stress.

1. Introduction

AI scientists now span a broad spectrum, from systems that assist with literature synthesis or hypothesis suggestion to agents that can plan experiments, invoke tools, interpret observations, and decide whether a scientific workflow should proceed (Lu et al., 2024; Yamada et al., 2025; Gottweis et al., 2025). In chemistry and autonomous-laboratory settings, this spectrum already extends into tool orchestration and experimental execution (Boiko et al., 2023; MacLeod et al., 2022; Canty et al., 2025; Coley et al., 2019). As these systems move closer to closed-loop scientific action, the central question shifts. It is no longer only whether a model can produce useful hypotheses or competent plans. It is whether one concrete *run* of an AI scientist workflow is admissible for audit, replay, and authoritative release.

That question is especially sharp in autonomous laboratories. A single run may couple planning, tool invocation, sensing, coordination, and irreversible decisions. Failures are then often systemic rather than model-local. A planner can be scientifically capable yet still produce a run that is difficult to inspect, impossible to replay, or unsafe to authorize. Conversely, a workflow may appear locally sensible while remaining globally weak as an externally checkable scientific record.

Current progress in AI scientists makes this gap harder to ignore. Recent systems increasingly demonstrate the ability to generate ideas, use external tools, coordinate agents, and interact with laboratory or domain infrastructure (Gottweis et al., 2025; Boiko et al., 2023). At the same time, emerging benchmark efforts increasingly measure practical scientific capabilities rather than only textbook reasoning (Laurent et al., 2024; Panigrahi et al., 2026). This is important progress, but it still leaves a distinct trust problem unresolved: when should a *particular run* of an AI scientist count as sufficiently inspectable and replayable to support release-relevant decisions?

This paper addresses that gap. We introduce *MADS-CPS*, a machine-checkable minimum assurance bar for AI scientist workflows in cyber-physical scientific settings. The framework is intentionally narrow. It does not claim certification, replace a domain safety case, or prescribe a controller architecture. Instead, it defines a run-level admissibility contract: the minimum evidence obligations required before one workflow run may count as auditable, replayable, and eligible for authoritative release under a declared assurance envelope.

The central design principle is separation of *capability* from

admissibility. The controller may be centralized, decentralized, rule-based, learning-based, or hybrid. MADS-CPS is deliberately agnostic to that internal choice. What matters is whether the workflow emits an externally checkable evidence package inside a declared envelope. Conformance is therefore a property of interface-level artifacts and validation predicates rather than a property of planner identity.

The paper makes four contributions. First, it defines a run-level admissibility contract for AI scientist workflows in autonomous laboratories in terms of a declared envelope, required artifacts, admissibility predicates, conformance tiers, verification modes, and point-of-no-return release semantics. Second, it formalizes controller-agnostic conformance at the assurance layer: the checker operates on emitted evidence rather than controller internals. Third, it introduces a restricted-auditability model that preserves structural integrity under redaction while making explicit which predicates are lost when payload content is hidden. Fourth, it provides a reproducible thin-slice instantiation in a robot-centric autonomous laboratory and evaluates the framework across conformance, audit-mode, replay, and controller-matrix studies that distinguish invariance in easier regimes from meaningful divergence under harder coordination stress.

The contribution is intentionally narrow but strategically placed. MADS-CPS does not attempt to solve scientific capability, laboratory safety, and governance in one step. It specifies the run-level evidence contract that makes those larger questions externally inspectable. That is the right abstraction for a reusable trust substrate: strong enough to gate release and support replay, yet agnostic to the internal controller family.

2. Problem Setting and Scope

We study AI scientist workflows that emit four artifact classes for each run:

(i) a structured execution trace,

(ii) an evaluation report,

(iii) an evidence bundle, and

(iv) a release manifest.

The core question is the following:

> What is the minimum set of machine-checkable evidence obligations under which a third party can determine whether one concrete autonomous-lab run is admissible for audit, replay, and release without inspecting controller internals?

The scope is intentionally constrained. We consider workflows that operate under a declared assurance envelope and emit the required artifacts above. We focus on machine-checkable predicates over those artifacts: presence, schema validity, replay status, integrity, and point-of-no-return (PONR) coverage. We also consider restricted auditability, where traces may be redacted while preserving structural information.

We do *not* claim certification or regulatory compliance; we do not claim bit-identical hardware replay; we do not prescribe a particular planner or coordination architecture; and we do not claim that public disclosure of all payloads is always possible. The reference organism is a robot-centric autonomous laboratory workflow, chosen because it is rich enough to involve coordination, tool use, and irreversible decisions while still supporting reproducible, run-level evaluation.

3. Related Work and Positioning

MADS-CPS sits at the intersection of AI scientist systems, AI-for-science benchmarking, AI risk management, assurance-case structure, and regulated data integrity.

AI scientists and autonomous laboratories. Recent systems increasingly span literature-grounded ideation, agentic planning, tool orchestration, and autonomous experimentation (Lu et al., 2024; Yamada et al., 2025; Gottweis et al., 2025; Boiko et al., 2023). In parallel, self-driving laboratories and robotic scientific platforms have shown how closed-loop optimization and physical execution can accelerate materials and chemical discovery (MacLeod et al., 2022; Canty et al., 2025; Coley et al., 2019). This literature is essential because it clarifies what AI scientist systems can perceive, decide, and execute. MADS-CPS is complementary. It does not benchmark discovery performance or scientific novelty. It specifies what evidence must exist so that one concrete AI scientist run can be externally audited, replayed, and release-gated.

Benchmarks for scientific agents. The evaluation landscape is also maturing. LAB-Bench measures practical biology-research capabilities such as literature search, protocol planning, and data analysis rather than textbook-style science questions (Laurent et al., 2024). HeurekaBench argues that many existing scientific benchmarks still do not capture end-to-end co-scientist workflows that generate open-ended hypotheses (Panigrahi et al., 2026). These efforts are valuable because they move evaluation closer to real scientific work. Their primary question, however, remains capability: can the system perform useful scientific tasks? MADS-CPS asks an orthogonal question: when should one concrete run count as admissible for audit, replay,

and release?

Risk management and governance. Frameworks such as NIST AI RMF and related generative-AI governance profiles provide useful lifecycle structure for identifying, managing, and communicating AI risk (Tabassi, 2023; Autio et al., 2024). They are important for organizational process and governance posture. Their level of abstraction, however, is different from the one pursued here. MADS-CPS is narrower and more operational: it asks what a third party can verify mechanically about one particular run, given a declared assurance envelope and a fixed artifact family.

Assurance cases and data integrity. Goal-based standards and structured assurance-case formalisms elevate evidence and argumentation as first-class objects (UL Standards & Engagement, 2023; Object Management Group, 2023; Kelly & Weaver, 2004). Scientific and laboratory environments also impose demanding expectations around record integrity, traceability, and auditability (OECD, 2021; U.S. Food and Drug Administration, 1997). MADS-CPS is motivated by these traditions, but it does not claim to replace them. Its role is narrower: to define a bounded, controller-agnostic, machine-checkable evidence object whose integrity, replay status, and release admissibility can be evaluated mechanically.

The positioning is therefore precise. MADS-CPS is not a capability benchmark, not a full governance framework, and not a complete safety case. It is a minimal run-level admissibility substrate for AI scientist workflows that operate in cyber-physical scientific settings.

4. MADS-CPS Model

The central object in MADS-CPS is a *run-level admissibility contract*. A run r is evaluated relative to a declared envelope

$$\mathcal{E} = (s, v, P),$$

where s is the scenario or profile identifier, v is the verification mode, and P is the set of release-relevant point-of-no-return (PONR) obligations for that scenario. The run emits four required artifacts

$$A(r) = (T_r, R_r, B_r, M_r),$$

corresponding to a structured trace, an evaluation report, an evidence bundle, and a release manifest.

Verification modes. The mode $v \in \{\texttt{public}, \texttt{evaluator}, \texttt{regulator}\}$ specifies which predicates must be checkable and whether replay is mandatory. Public mode permits restricted auditability when payload content is redacted, while evaluator and regulator modes may require replay-capable traces.

Replay contract. We use a bounded replay notion. *L0 replay* means that decision and enforcement outcomes are recomputable from the trace and recorded state transitions. *L1 replay* extends L0 by including recorded observations. MADS-CPS does not require bit-identical hardware replay.

Definition 4.1 (Conformance checker)**.** *For a run r under envelope $\mathcal{E} = (s, v, P)$, define the checker $C(r; \mathcal{E})$ by*

$$\begin{aligned}
\mathrm{Tier1}(r) &\iff \textit{all artifacts in } A(r) \textit{ are present and schema-valid},\\
\mathrm{Tier2}(r) &\iff \mathrm{Tier1}(r) \wedge \mathrm{bundleOK}(B_r) \wedge \mathrm{replayOK}_v(r),\\
\mathrm{Tier3}(r) &\iff \mathrm{Tier2}(r) \wedge \forall p \in P,\ \mathrm{witnessed}(p, T_r).
\end{aligned}$$

Here $\mathrm{bundleOK}(B_r)$ *requires machine-checkable admissibility predicates recorded in the evidence bundle, including schema-validation status and replay status;* $\mathrm{replayOK}_v(r)$ *denotes replay success whenever replay is required by mode v; and* $\mathrm{witnessed}(p, T_r)$ *holds when the trace contains a corresponding completion event for obligation p.*

Definition 4.2 (Fail-closed release)**.** *A release-relevant transition is authorized only if the run satisfies the admissibility conditions required by its declared envelope. If those conditions cannot be established, release is denied unless an explicit override is recorded in an auditable manner.*

This formulation is intentionally interface-level. Conformance is computed over emitted evidence rather than planner internals, which allows the admissibility layer to survive controller substitution as long as the declared envelope and conformance-relevant artifact predicates are preserved.

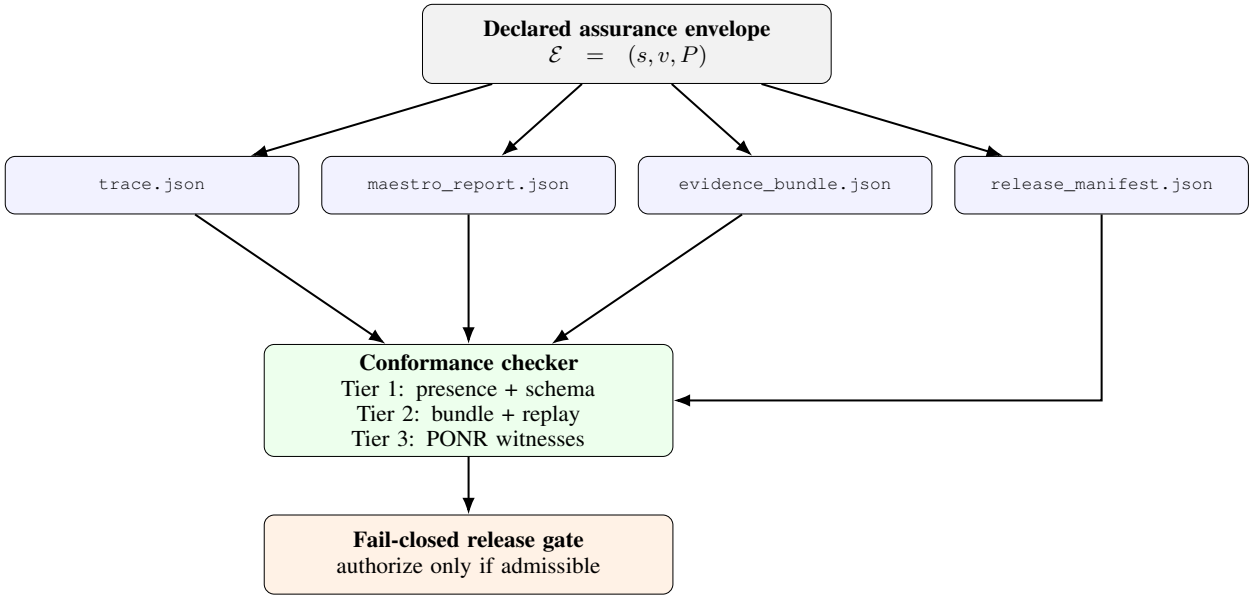

Figure 1. MADS-CPS evaluates each run relative to a declared assurance envelope and emitted artifact family. Admissibility is decided by a machine-checkable conformance checker and propagated to a fail-closed release boundary.

5. Properties of the Admissibility Layer

Proposition 5.1 (Tier monotonicity)**.** *For any run r under declared envelope $\mathcal{E}$,*

$$\mathrm{Tier3}(r) \implies \mathrm{Tier2}(r) \implies \mathrm{Tier1}(r).$$

Proposition 5.2 (Controller invariance of conformance)**.** *Fix a declared envelope $\mathcal{E}$. Let r_1 and r_2 be runs produced by possibly different controllers. If all predicates read by $C(\cdot; \mathcal{E})$ coincide on r_1 and r_2, then*

$$C(r_1; \mathcal{E}) = C(r_2; \mathcal{E}).$$

Proposition 5.3 (Redaction preservation boundary)**.** *Under structure-preserving payload redaction, schema-validity and integrity predicates may remain checkable while replay and payload-dependent PONR predicates may become unavailable.*

Proof sketches. Tier monotonicity follows directly from the nesting of the tier predicates. Controller invariance follows because the checker reads only artifact-level predicates and never branches on controller identity. The redaction boundary follows because structural fields and integrity digests can remain observable even when payload content needed for replay is withheld. Full proofs are deferred to the appendix.

6. Reference Instantiation

We instantiate MADS-CPS in a robot-centric autonomous laboratory profile. The thin-slice implementation emits `trace.json`, `maestro_report.json`, `evidence_bundle.json`, and `release_manifest.json`.

Two scenarios matter in the evaluation: `toy_lab_v0`, a simpler scenario with no PONR tasks in scope, and `lab_profile_v0`, which includes the PONR-bearing task `disposition_commit`. The former is sufficient for artifact-level and replay-level reasoning; the latter is the stronger witness for Tier 3 and release-gating semantics because it exercises PONR coverage directly.

7. Experimental Design

We evaluate MADS-CPS through four experiments that target complementary properties of the admissibility substrate.

E1: Conformance challenge corpus. We construct an eight-case corpus of positive and negative run directories with injected faults, including missing artifacts, schema-invalid traces, state-hash corruption, bundle replay mismatch, missing PONR evidence, and stale release-manifest fields. The metric is checker agreement with the intended conformance outcome.

E2: Verification-mode admissibility matrix. We evaluate which predicates remain checkable under full and redacted audit conditions. The outputs are predicate-availability judgments for schema validation, integrity, replay, and PONR coverage across verification modes.

E3: Independent replay link. An independent verifier recomputes the evaluation report from the trace in a separate process from the producer pipeline. We run two scenarios with twenty seeds per scenario and report strong replay match rate and seed-level verifier-side latency summaries.

E4: Controller matrix under baseline and stressed regimes. We evaluate two controllers, `centralized` and `rep_cps`, across three scenarios and four regimes: `baseline`, `moderate`, `stress`, and `coordination_shock`. For each controller–scenario–regime combination, we report raw conformance, normalized-interface conformance, strong replay match rate, verifier-side latency, and controller-pair diagnostics. The goal is not to claim universal controller equivalence, but to determine where evidence-layer invariance holds and where controller behavior diverges under stress.

8. Results

Taken together, the four result blocks establish decidability, mode-sensitive auditability, externally verifiable replay, and a regime-dependent controller story in which evidence-layer admissibility can remain stable even when productivity separates under stress.

Table 1. E1 conformance challenge corpus.

Case	Fault injected	Expected	Observed
`valid_toy`	none	T3 pass	T3 pass
`valid_lab`	none	T3 pass	T3 pass
`missing_artifact`	missing `maestro_report.json`	T1 fail	T1 fail
`schema_invalid`	schema-invalid trace	T1 fail	T1 fail
`hash_mismatch`	corrupted `state_hash_after`	T2 fail	T2 fail
`replay_mismatch`	evidence bundle records `replay_ok=false`	T2 fail	T2 fail
`missing_ponr`	missing PONR witness	T3 fail	T3 fail
`stale_manifest`	missing `release_id`	T1 fail	T1 fail

E1 establishes mechanical decidability. Across all eight challenge cases in Table 1, the checker agrees with the intended outcome. This is a small but load-bearing result: admissibility is not merely described but decided by an implemented procedure that catches faults at each intended layer of the evidence stack.

E2 clarifies the redaction boundary. Table 2 shows that redaction does not destroy all auditability. Schema validity and integrity remain checkable, while replay and payload-dependent PONR predicates may become unavailable. This distinguishes restricted auditability from failed auditability.

E3 makes replay externally checkable. In E3, strong replay match rate is 1.00 in both reported scenarios across twenty seeds per scenario, showing that the evaluation report can be recomputed by an independent verifier rather than trusted as producer telemetry.

Table 2. E2 verification-mode admissibility matrix.

Predicate	Full	Evaluator	Regulator	Public redacted
Schema validity	yes	yes	yes	yes
Integrity	yes	yes	yes	yes
Replay	yes	conditional	conditional	no / N.A.
PONR coverage	yes	conditional	conditional	N.A.

Table 3. E3 replay-link under an independent verifier using strong replay semantics.

Scenario	Controller	Seeds	Replay match rate	Verifier $p95$ (ms)
`lab_profile_v0`	thinslice	20	1.00	50.37 [40.48, 60.27]
`toy_lab_v0`	thinslice	20	1.00	35.37 [27.11, 43.62]

Table 4. Focused E4 controller divergence under harder coordination stress.

Controller	Raw conf.	Strong replay	Prod. succ.	Safe nonprod.	Mean tasks / mean $p95$
`centralized`	1.00	1.00	0.55	0.00	3.30 / 147.25
`rep_cps`	1.00	1.00	0.00	1.00	0.00 / 0.00

E4 separates admissibility from productivity. The controller matrix yields a more precise picture than a simple "same checker, same results" story. In baseline settings, raw controller invariance holds, and normalized-interface invariance also holds in baseline and moderate settings. Under harder coordination stress, however, behavior separates: Table 4 shows that on `rep_cps_scheduling_v0`, `centralized` remains productive while `rep_cps` becomes safe-nonproductive, even though both remain raw-conformant and strong-replay-matching. Admissibility can therefore remain stable even when productivity diverges.

9. Discussion

The results support a narrow but important claim. MADS-CPS does not certify an autonomous laboratory, establish scientific correctness, or prove that one controller family is globally safer than another. It makes run-level admissibility machine-checkable. In the present instantiation, a third party can determine whether a run is structurally complete, replay-admissible under its declared verification mode, and appropriately witnessed at release-relevant boundaries.

This matters for AI scientists because the field is moving from isolated prediction tools toward systems that increas-

ingly resemble scientific actors. Once a system can plan, coordinate, and act across a closed-loop workflow, trust can no longer be reduced to benchmark accuracy or local task success. What becomes necessary is an evidence contract for concrete runs.

The controller-matrix results sharpen that distinction. MADS-CPS does not prove controller equivalence. Instead, it separates assurance-validity from operational productivity. In baseline settings, controller substitution can preserve evidence-layer conformance. Under harder coordination stress, controllers can diverge sharply in productivity while still producing admissible, replay-consistent evidence.

The restricted-auditability result is equally important. In realistic scientific settings, full trace disclosure may be blocked by intellectual property, safety, or regulatory constraints. The relevant question is therefore what survives redaction and what does not. MADS-CPS makes that boundary explicit: structural validity and integrity can remain externally checkable even when replay and payload-dependent predicates become unavailable.

10. Limitations and Non-Goals

The implementation is a thin slice with synthetic workloads; no physical hardware is actuated. The results should therefore be interpreted as a proof of structure rather than a field-certification result. The replay contract is limited to declared fidelity, not bit-identical hardware replay. The PONR mechanism is a release-time gate rather than a live runtime safety controller. Finally, controller-independence is not claimed as a universal property across all regimes. The supported claim is narrower: evidence-layer invariance holds in baseline settings, normalized-interface invariance holds in baseline and moderate settings, and harder coordination stress can reveal meaningful controller divergence without collapsing admissibility.

11. Conclusion

We introduced MADS-CPS, a machine-checkable admissibility contract for AI scientist workflows in autonomous laboratories. The framework defines a declared assurance envelope, a conformance checker over required artifacts and replay status, verification modes for restricted auditability, and fail-closed point-of-no-return release gating. In a reproducible thin-slice instantiation, the resulting admissibility layer is mechanically decidable, externally verifiable under declared audit modes, and stable across controller substitutions in baseline settings, while harder coordination stress reveals that admissibility can remain intact even when productivity collapses.

The broader implication is straightforward. Closed-loop AI scientists need a run-level evidence contract if they are to become inspectable scientific actors rather than opaque automation stacks. Models, planners, and orchestration layers will continue to change. What should remain stable is the boundary at which a run becomes admissible for replay, audit, and release.

Impact Statement

The intended positive impact is to make such systems more auditable, replayable, and disciplined at release boundaries, especially in settings where scientific actions can have irreversible consequences. A stronger evidence contract can improve external oversight, reproducibility, and accountability. At the same time, such artifacts can create a false sense of security if they are mistaken for full certification or proof of scientific validity. We therefore frame MADS-CPS as a minimum admissibility substrate rather than a complete safety case.

Reproducibility Statement

The paper is supported by a frozen artifact set including an eight-case conformance corpus, a redaction demo and admissibility matrix, strong replay-link summaries across two scenarios and twenty seeds per scenario, and a controller-matrix study spanning three scenarios, four regimes, and two controllers with raw and normalized summaries, diagnostics, and claim-bounding exports.

References

Autio, C., Schwartz, R., Dunietz, J., Jain, S., Stanley, M., Tabassi, E., Hall, P., and Roberts, K. Artificial intelligence risk management framework: Generative artificial intelligence profile. Technical Report NIST AI 600-1, National Institute of Standards and Technology, 2024.

Boiko, D. A., MacKnight, R., Kline, B., and Gomes, G. Autonomous chemical research with large language models. *Nature*, 624:570–578, 2023. doi: 10.1038/s41586-023-06792-0.

Canty, R. B., Bennett, J. A., Brown, K. A., et al. Science acceleration and accessibility with self-driving labs. *Nature Communications*, 16:3856, 2025. doi: 10.1038/s41467-025-59231-1.

Coley, C. W., Thomas, D. A., Lummiss, J. A. M., Jaworski, J. N., Breen, C. P., Schultz, V., Hart, T., Fishman, J. S., Rogers, L., Gao, H., Hicklin, R. W., Plehiers, P. P., Byington, J., Piotti, J. S., Green, W. H., Hart, A. J., Jamison, T. F., and Jensen, K. F. A robotic platform for flow synthesis of organic compounds informed by

AI planning. *Science*, 365(6453):eaax1566, 2019. doi: 10.1126/science.aax1566.

Gottweis, J., Weng, W.-H., Daryin, A., Tu, T., Palepu, A., Sirkovic, P., Myaskovsky, A., Weissenberger, F., Rong, K., Tanno, R., Saab, K., Popovici, D., Blum, J., Zhang, F., Chou, K., Hassidim, A., Gokturk, B., Vahdat, A., Kohli, P., Matias, Y., Carroll, A., Kulkarni, K., Tomasev, N., Guan, Y., Dhillon, V., Vaishnav, E. D., Lee, B., Costa, T. R. D., Penadés, J. R., Peltz, G., Xu, Y., Pawlosky, A., Karthikesalingam, A., and Natarajan, V. Towards an AI co-scientist, 2025. URL https://arxiv.org/abs/2502.18864.

Kelly, T. and Weaver, R. The goal structuring notation — a safety argument notation. In *Proceedings of the International Conference on Dependable Systems and Networks Workshop on Assurance Cases*, 2004.

Laurent, J. M., Janizek, J. D., Ruzo, M., Hinks, M. M., Hammerling, M. J., Narayanan, S., Ponnapati, M., White, A. D., and Rodriques, S. G. LAB-Bench: Measuring capabilities of language models for biology research, 2024. URL https://arxiv.org/abs/2407.10362.

Lu, C., Lu, C., Lange, R. T., Foerster, J., Clune, J., and Ha, D. The AI scientist: Towards fully automated open-ended scientific discovery, 2024. URL https://arxiv.org/abs/2408.06292.

MacLeod, B. P., Parlane, F. G. L., Rupnow, C. C., et al. A self-driving laboratory advances the pareto front for material properties. *Nature Communications*, 13:995, 2022. doi: 10.1038/s41467-022-28580-6.

Object Management Group. Structured assurance case metamodel (SACM) specification, version 2.3. https://www.omg.org/spec/SACM/2.3, 2023.

OECD. GLP data integrity. Technical Report OECD Series on Principles of Good Laboratory Practice and Compliance Monitoring, No. 22, OECD Publishing, 2021.

Panigrahi, S. S., Videnovic, J., and Brbic, M. Heurekabench: A benchmarking framework for AI co-scientist, 2026. URL https://arxiv.org/abs/2601.01678.

Tabassi, E. Artificial intelligence risk management framework (AI RMF 1.0). Technical Report NIST AI 100-1, National Institute of Standards and Technology, 2023.

UL Standards & Engagement. UL 4600: Standard for evaluation of autonomous products, 2023. Edition 3.

U.S. Food and Drug Administration. 21 CFR part 11 — electronic records; electronic signatures. https://www.ecfr.gov/current/title-21/part-11, 1997.

Yamada, Y., Lange, R. T., Lu, C., Hu, S., Lu, C., Foerster, J., Clune, J., and Ha, D. The AI scientist-v2: Workshop-level automated scientific discovery via agentic tree search, 2025. URL https://arxiv.org/abs/2504.08066.