CITIUM
Companion Research Preprint

Ian Broderick
Citium · editor@citium.ovrinlabs.ca
ORCID 0009-0009-7198-0679

Structural Inevitability as a Response-Level Criterion

for AGI Evaluation

ABSTRACT

This paper develops structural inevitability as a local criterion for AGI evaluation at the problem-response level. The unit is not a system label in isolation. It is a well-formed problem under a fixed representation, a declared semantic equivalence relation, and an admissible response-class space. A response satisfies the criterion when it lands in the class the completed specification makes uniquely cheapest by a margin that remains meaningful after approximation error. The result is a criterion for response-level success, not a complete theory of AGI.

The account has three layers. The ideal layer uses exact Kolmogorov complexity as a limiting analytical object. The computable layer uses Charter Protocol to compile a raw problem into a charter-enriched specification with candidate classes, structural surrogates, verifiers, routing, and escalation. The empirical layer tests proxy margins against correctness, invariant satisfaction, false acceptance, and inadequacy behavior.

Three subsidiary results are stated under indexed assumptions: posterior concentration for an explicit observer model, invariant-constrained completion for value-laden specifications, and finite coverage for charter authoring over reachable, charter-realizable families. Every guarantee is tied to well-formedness, margin, observer model, verifier adequacy, proxy error, and charter coverage.

Keywords: AGI evaluation, structural inevitability, Charter Protocol, Kolmogorov complexity, posterior concentration, invariant-constrained completion, specification completeness

INTRODUCTION

Most AGI definitions begin with the system: its behavior across tasks, environments, benchmarks, or a universal reward distribution (Turing, 1950; Goertzel and Pennachin, 2007; Legg and Hutter, 2007; Hutter, 2005; Chollet, 2019; Morris et al., 2024). The present paper begins with a smaller unit: a problem and the response returned to it. The proposed criterion is satisfied when the completed problem specification separates one admissible semantic class from all rivals by a conditional description-length margin that remains visible under approximation.

In practice, the question is whether the problem has been specified enough to favor the correct semantic class. A chartered mathematics problem names the objects, transformations, and verification criteria. A chartered policy task names authority, temporal scope, invariants, exceptions, and escalation rules. In each case, the achievement is recovery of the determined semantic class under stated conditions.

The theory has three layers:

1. **Ideal target:** exact structural inevitability over semantic response classes.
2. **Computable approximation:** Charter Protocol as specification compilation, routing, verification, and escalation.
3. **Empirical proxy:** benchmark regimes where correctness, invariant satisfaction, and K-gap proxies can be tested.

All major claims in the paper are indexed to those layers. Exact Kolmogorov complexity supplies the limiting object; charters supply a computable approximation path; experiments test whether the approximation behaves as the theory predicts.

What this paper does not claim. Structural inevitability is not a full system-level account of artificial general intelligence. It does not supply autonomy, continual learning, open-ended exploration, transfer, value elicitation, governance, public legitimacy, or deployment safety by itself. The invariant result presupposes feasible grounded values inside the admissible set; the observer-credence result is a posterior bound for a declared observer score; the coverage result is finite and limited to reachable charter-realizable families.

DEFINITIONS

The definitions begin at the response level. The surface string is only a representative. Two answers may differ in wording, order, or implementation detail while carrying the same task-relevant solution. The object ranked by $K(\cdot \mid P)$ is the admissible semantic response class.

DEFINITION 2.1 (SEMANTIC RESPONSE CLASS)

For a problem specification P, let $Y(P)$ be the surface response space and let $\equiv_P$ be a task-relevant semantic equivalence relation supplied by the specification or charter. A semantic response class is $[R]_P \in Y(P)/\equiv_P$. This quotient ignores irrelevant paraphrase, formatting, or implementation differences.

This is the same quotienting move used in the companion verification paper (Broderick, 2026): the object of correctness is the semantic response class. Literal strings are surface representatives of that class.

DEFINITION 2.2 (STRUCTURALLY INEVITABLE COMPLETION)

Let $S(P) \subseteq Y(P)/\equiv_P$ be the admissible semantic solution classes. A class $S^\star \in S(P)$ is structurally inevitable with margin $\delta > 0$ when

$$K_U(S^\star \mid P) + \delta \leq K_U(S \mid P) \qquad \forall S \neq S^\star \in S(P),$$

relative to a fixed reference language or representation U.

DEFINITION 2.3 (PER-INSTANCE RESPONSE-LEVEL SI CRITERION)

A system satisfies the Problem-Solution Isomorphism / Structural Inevitability (PSI/SI) response criterion on a well-formed problem P when it returns $S^\star(P)$, the structurally inevitable semantic completion of P, or returns an explicit inadequacy signal when the required well-formedness, coverage, verifier adequacy, or proxy margin is unresolved.

DEFINITION 2.4 (PER-FAMILY SI COMPETENCE)

Over a domain family D with distribution μ, a system approaches response-level SI competence to the extent that it satisfies the per-instance criterion on well-formed, covered instances sampled from μ. Reliable satisfaction across a representative domain family is relevant to AGI evaluation, but it is not sufficient for a system-level AGI claim.

REMARK 2.5 (IDEALIZATION)

Exact K is uncomputable and representation-dependent up to additive constants (Solomonoff, 1964; Rissanen, 1978; Grunwald, 2007; Li and Vitanyi, 2008). All exact K statements are ideal limiting statements. Practical systems require computable surrogates such as proof length, program length, model codelength, plan length, or domain-specific structural cost.

REMARK 2.6 (OPERATIONAL MARGIN)

Let L be a declared computable proxy and let $\hat{\delta}_L(P)$ be the estimated proxy margin. A practical SI claim requires the lower confidence bound on $\hat{\delta}_L(P)$ to exceed estimated proxy error and representation tolerance, for example $\hat{\delta}_L(P) > 2\hat{\epsilon}_L(P) + \tau_U$. If proxy families disagree, the system should report unresolved adequacy rather than assert structural inevitability.

OBSERVER-RELATIVE POSTERIOR CONCENTRATION

The posterior-concentration result is deliberately narrow. Fix an observer, the evidence available to that observer, a scoring rule over a finite candidate set, and a Gibbs-form credence model. If the correct class is visibly separated from its rivals under that score, the posterior mass assigned to it increases with the visible gap. This is an epistemic calculation inside the model; psychological reliance, social warrant, and institutional legitimacy remain outside its scope.

DEFINITION 3.1 (OBSERVER POSTERIOR CREDENCE)

Let O be an observer with evidence E_O. The posterior credence of O in proposed response class S is

$$T_O(P, S; E_O) = \Pr_O[S^*(P) = S \mid P, E_O].$$

This is the observer's posterior credence that the proposed class S is the correct class. Psychological reliance and institutional legitimacy require additional evidence. When the evidence is fixed by context, write $T_O(P, S)$.

DEFINITION 3.2 (OBSERVER-VISIBLE STRUCTURAL GAP)

Let $L_O(S \mid P)$ be the description-length or structural score used by observer O over a finite candidate class $S(P)$. If S^* is the unique minimizer, define

$$\Delta_O(P) = \min_{S \neq S^*} (L_O(S \mid P) - L_O(S^* \mid P)).$$

THEOREM 3.3 (POSTERIOR CONCENTRATION UNDER VISIBLE STRUCTURAL GAP)

Assume the observer posterior in Definition 3.1 is modeled by the Gibbs distribution

$$T_O(P, S) = \pi_O(S \mid P) = \frac{2^{-L_O(S \mid P)}}{\sum_{Z \in S(P)} 2^{-L_O(Z \mid P)}}.$$

Let $N_P = |S(P)|$. If S^* is the uniquely correct class and the unique minimizer of L_O with visible gap $\Delta_O(P) > 0$, then

$$T_O(P, S^*) \geq \frac{1}{1 + (N_P - 1)2^{-\Delta_O(P)}}.$$

For fixed N_P, the posterior lower bound is increasing in $\Delta_O(P)$ and tends to 1 as $\Delta_O(P) \to \infty$.

PROOF.

Since S^* minimizes L_O, every rival S has $L_O(S \mid P) \geq L_O(S^* \mid P) + \Delta_O$. The denominator is therefore bounded above by

$$2^{-L_O(S^* \mid P)} + (N_P - 1)2^{-L_O(S^* \mid P) - \Delta_O}.$$

Dividing numerator and denominator by $2^{-L_O(S^* \mid P)}$ gives the stated bound.

COROLLARY 3.4 (IDEAL-TO-OBSERVER GAP TRANSFER)

If $|L_O(S \mid P) - K(S \mid P)| \leq \varepsilon_O$ for all $S \in S(P)$ and the PSI/SI gap is δ, then the visible gap satisfies $\Delta_O(P) \geq \delta - 2\varepsilon_O$. Observer-credence concentration requires a structural gap that remains legible relative to observer error.

REMARK 3.5 (SCOPE OF POSTERIOR-CONCENTRATION THEOREM)

The theorem is about concentration inside a declared observer model. The observer does not need access to exact Kolmogorov complexity; it needs a scoring rule that exposes a gap among the candidate classes. In practice, the retained gap depends on the observer's evidence, candidate set, score, and error profile.

INVARIANT-CONSTRAINED ADMISSIBLE COMPLETION

The invariant-constrained result is a statement about problem identity. In a value-laden problem, grounded value conditions define the admissible classes rather than arriving as an after-the-fact preference overlay. If those conditions are active in P^+, a PSI/SI optimum over the admissible set cannot violate them. A value-violating optimum indicates that the effective problem has changed.

DEFINITION 4.1 (VALUE INVARIANTS)

Let P_0 be a base task and let V be a set of semantically grounded value invariants. The value-laden problem is

$$P^+ = \langle P_0, E, V, Y_{P^+}, \equiv_{P^+} \rangle,$$

where E is the modeled environment and $\equiv_{P^+}$ is the task-relevant response equivalence relation. The admissible completion set is

$$C(P_0, V) = \{S \in Y_{P^+} / \equiv_{P^+} : \mathrm{Task}_{P_0}(S) \wedge \mathrm{Inv}_V(S)\}.$$

LEMMA 4.2 (INVARIANT-CONSTRAINED ADMISSIBLE COMPLETION)

Suppose the value invariants in V are semantically grounded, $C(P_0, V)$ is nonempty, and P^+ achieves structural inevitability over $C(P_0, V)$ with margin δ: there exists $S^\star \in C(P_0, V)$ such that

$$K(S^\star \mid P^+) + \delta \leq K(S \mid P^+) \quad \forall S \neq S^\star \in C(P_0, V).$$

Then any PSI/SI-optimal completion of P^+ satisfies V. Any value-violating response falls outside the PSI/SI completion of P^+; if optimal, it is optimal for some different effective problem $\tilde{P}$ in which one or more invariants were omitted, weakened, mistranslated, or rendered inactive by approximation or context drift.

PROOF.

By definition, PSI/SI optimization occurs over the admissible set $C(P_0, V)$. Every member of that set satisfies Inv_V. Since $S^\star$ is the unique δ-separated minimizer within the admissible set, it satisfies V. A response violating V is outside $C(P_0, V)$ and is excluded from the completion of P^+. It can be interpreted as a completion of a problem whose admissible set differs from the intended one.

REMARK 4.3 (SCOPE OF INVARIANT-CONSTRAINED COMPLETION)

At the ideal level, invariant satisfaction is absorbed into problem completion when value invariants are feasible, grounded, and active inside the specification. The remaining work is specification work: eliciting values, resolving stakeholder conflicts, maintaining invariants under deployment shift, auditing approximations, interpreting failures, and designing governance around incomplete or unstable specifications.

Failure Modes

The result loses force when the value-laden problem fails to determine a stable admissible set. Typical causes include underspecified values, unresolved stakeholder conflict, hidden externalities, distribution shift, approximate optimization, multiple optima, and representation drift. In those cases the repair question is concrete: which constraint failed to enter, failed to translate, failed to bind, or ceased to apply?

CHARTER PROTOCOL AS COMPUTABLE APPROXIMATION

Charter Protocol is the operational layer that makes well-formedness executable before a model answers. A charter records the domain boundary, normalizer, hard constraints, invariants, permitted and prohibited methods, success criteria, verifiers, routing rules, and escalation behavior. Compilation turns a raw problem into a charter-enriched specification with a candidate space, a computable structural surrogate, a verifier, and an inadequacy path.

The role of the protocol is to make the SI condition operational enough to guide execution. When the charter is adequate and the surrogate is order-consistent with the ideal ranking on valid semantic solutions, the executor can recover the PSI/SI minimizer for the covered family.

DEFINITION 5.1 (DOMAIN CHARTER)

A domain charter is an executable tuple

$$C = \langle D_C, N_C, \Gamma_C, I_C, M_C^+, M_C^-, O_C, S_C, V_C, \rho_C, E_C \rangle,$$

where D_C is a domain predicate, N_C normalizes the problem, Γ_C are hard constraints, I_C are invariants, M_C^+ and M_C^- specify permitted and prohibited methods, O_C is the output contract, S_C are success criteria, V_C are verification procedures, ρ_C are routing rules, and E_C are escalation rules.

DEFINITION 5.2 (META-CHARTER)

A meta-charter is a partial routing map

$$M : (P, L) \mapsto \{C \in L\} \cup \{\text{compose}, \text{clarify}, \text{synthesize}, \text{reject}\},$$

where L is the current charter library.

DEFINITION 5.3 (CHARTER COMPILATION)

For a charter C and problem P, compilation yields

$$\kappa(C, P) = (\tilde{P}, H_{C,P}, L_{C,P}, V_{C,P}, E_{C,P}),$$

where $\tilde{P}$ is the charter-enriched specification, $H_{C,P}$ is the induced candidate space, $L_{C,P}$ is a computable structural surrogate, $V_{C,P}$ is the compiled verifier, and $E_{C,P}$ is escalation behavior.

DEFINITION 5.4 (ORDER-CONSISTENCY)

For valid semantic solutions in $H_{C,P}$, the surrogate $L_{C,P}$ is order-consistent with the ideal ranking when, for any classes S, S' in the valid candidate space,

$$K_U(S \mid \tilde{P}) < K_U(S' \mid \tilde{P}) \implies L_{C,P}(S \mid \tilde{P}) < L_{C,P}(S' \mid \tilde{P}).$$

Operationally, a charter may claim margin-limited order-consistency only above a declared proxy-error threshold. Below that threshold, it must return unresolved adequacy rather than assert structural inevitability.

DEFINITION 5.5 (CHARTER ADEQUACY)

A charter C is adequate for P when compilation preserves task-relevant semantic classes and invariants; $V_{C,P}$ is sound on invalid candidates and passes at least one representative of each valid semantic class that can be selected or act as a margin-relevant rival under the declared surrogate threshold, unless it returns explicit verifier inadequacy; $L_{C,P}$ is order-consistent with the ideal ranking on valid candidates up to the declared margin; and the executor

$$\mathrm{Exec}_C(P) = \arg \min_{R \in H_{C,P} : V_{C,P}(R) = \mathrm{pass}} L_{C,P}(R \mid \tilde{P})$$

finds a verifier-passing candidate if one exists in the compiled candidate space or returns an explicit inadequacy witness. Silent answers are disallowed when coverage, verifier adequacy, or proxy margin is unresolved. When $L_{C,P}$ is applied to a semantic class S, it denotes the minimum surrogate cost over verifier-passing representatives of S in $H_{C,P}$; a returned representative is identified with its semantic class for class-level recovery claims. If verifier-passing proxy minimizers fall into more than one semantic class inside the declared proxy-error band, the executor returns unresolved adequacy unless those minimizers are semantically equivalent.

Worked Charter Examples

Chartered sorting function. A small code charter can make the response-level criterion concrete. The domain predicate accepts tasks asking for a Python function over a finite list of integers. The normalizer fixes the signature, input range, and output contract. The invariants require that the output be nondecreasing, preserve the input multiset, terminate within a declared bound, and avoid external service calls. Permitted methods include direct comparison and local data structures; prohibited methods include network calls, hidden global state, and changing the input type. The verifier runs property tests for ordering and multiset preservation plus edge cases for empty lists, duplicates, negative integers, and already-sorted inputs. The surrogate cost can combine AST size, proof/certificate length, and invariant violations. If the prompt asks for behavior outside the charter, such as sorting arbitrary objects by an unspecified comparator, the correct response is an inadequacy signal rather than an unverified answer.

Chartered reimbursement policy. A rule-governed policy charter has the same structure. The domain predicate accepts reimbursement questions under a named policy version and date. The normalizer extracts claimant role, purchase category, amount, approval status, exception basis, and time window. Invariants encode authority order: statute or contract overrides internal policy; explicit approval overrides default denial only when the policy permits delegation; missing receipts trigger escalation instead of silent approval. Prohibited methods include inventing missing approvals or using a later policy version. The verifier checks each decision against the normalized fields and returns pass, reject, or inadequacy. The structural surrogate prefers the shortest decision path that satisfies the authority order and all invariants. A response that approves a request by ignoring the receipt rule is not a rival completion of the same chartered problem; it is a completion of a different effective problem with a weakened invariant.

The next lemma is the local bridge from ideal SI to execution. Adequacy preserves the intended solution set; verifier soundness prevents invalid candidates from passing; order-consistency makes the computable surrogate choose the same semantic class as the ideal ranking.

LEMMA 5.6 (LOCAL PROXY-TO-IDEAL BRIDGE)

Fix a certifiably well-formed problem P and an adequate charter C. If the compiled surrogate $L_{C,P}$ is order-consistent with the ideal K ranking on valid semantic solutions, then the charter executor returns the PSI/SI minimizer on that family. If the ideal minimizer is δ-separated, the executor returns the unique δ-separated PSI/SI minimizer.

PROOF.

Adequacy preserves the correct solution set. By adequacy, the SI class and every relevant valid rival have verifier-passable representatives in the compiled candidate space, while soundness prevents invalid candidates from passing. Order-consistency makes minimization of the computable surrogate over verifier-passing valid representatives select the same semantic class as minimization of the ideal description length. If the ideal minimizer is separated by margin δ, the selected class is the unique δ-separated PSI/SI minimizer.

The coverage result is a finite coverage statement. The reachable regime is already partitioned into finitely many charter-realizable families. Covered families have verified-adequate charters. Uncovered families produce witnesses, and the authoring operator converts those witnesses into verified charters. Conservative routing prevents silent answers outside the verified library during this process.

THEOREM 5.7 (FINITE COVERAGE UNDER CHARTER AUTHORING)

Let a reachable regime R be partitioned into finitely many charter-realizable families $F_1, \ldots, F_m$. Let $I_0 \subseteq \{1, \ldots, m\}$ be the families initially covered by verified-adequate charters. Suppose every initially uncovered family eventually produces a witness problem, the authoring operator eventually synthesizes a verified-adequate charter from such a witness, and the meta-charter conservatively routes, clarifies, synthesizes, or rejects while avoiding unverified answers. Then after at most $m - |I_0|$ successful charter-authoring events, where success means producing a verified-adequate charter that covers at least one previously uncovered family, the library covers R.

PROOF SKETCH.

Because R is partitioned into finitely many families, there are $m - |I_0|$ uncovered coverage obligations after initial coverage. For any uncovered family, the assumptions guarantee that a witness is eventually produced and that the authoring operator eventually converts that witness into a verified-adequate charter. Each successful authoring event strictly reduces the number of uncovered families. After at most $m - |I_0|$ such events, all families are covered. Conservative routing prevents the system from silently treating uncovered families as covered while this process is incomplete.

REMARK 5.8 (BOUNDARY OF FINITE COVERAGE)

The theorem is intentionally bounded. It covers reachable, charter-realizable families and excludes open-ended self-referential task families, unknown empirical phenomena, uncheckable domains, and tasks outside the available tools and representations.

EXPERIMENTAL PROGRAM

The empirical question is whether charter-conditioned inference behaves like the computable approximation predicts. A successful charter should improve correctness and invariant satisfaction by favoring the intended semantic class before generation. The tests need external correctness checks, explicit invariants, measurable proxy gaps, and comparison against minimal prompting and ordinary context augmentation. Synthetic rule worlds, theorem proving, code generation with hidden tests, and rule-based policy reasoning are appropriate first tests; HorizonMath is a later-stage stress test because it targets predominantly unsolved mathematical discovery with automated verification (Wang et al., 2026). The hypotheses are:

1. charter conditioning improves correctness and invariant satisfaction;
2. estimated structural gap correlates with correctness;
3. charter advantage increases with model capability when the charter is adequate and creates a positive structural margin;
4. SI verification failures diagnose missing invariants or wrong routing.

These tests evaluate whether the computable approximation behaves as the theory predicts; exact Kolmogorov-minimality remains an ideal target.

RELATION TO EXISTING AGI AND ALIGNMENT WORK

Existing AGI definitions primarily evaluate systems across tasks, environments, or behavioral benchmarks. AIXI and universal intelligence define system-level optimality across environments (Hutter, 2005; Legg and Hutter, 2007). ARC-style work emphasizes abstraction and skill acquisition (Chollet, 2019). PSI/SI evaluates a different object: whether a response is the structurally inevitable completion of a specified problem. This response-level criterion can coexist with system-level measures, while giving a formal account of local problem completion.

Panigrahy and Sharan prove an incompatibility result under strict definitions of safety, trust, and AGI: safety means never making false claims, trust means assuming safety, and AGI means matching or exceeding human capability (Panigrahy and Sharan, 2025). PSI/SI does not refute that result. It restricts guarantees to charter-bounded domains with explicit admissible sets, verifiers, and inadequacy behavior.

The computable layer is also adjacent to scalable oversight and preference-shaping methods such as RLHF, Constitutional AI, amplification, and debate (Christiano et al., 2017; Ouyang et al., 2022; Bai et al., 2022; Christiano et al., 2018; Irving et al., 2018). Those methods shape training signals, critiques, and supervision. Charter Protocol instead asks whether the problem has been compiled into a form where admissibility, invariants, verifiers, and escalation are explicit before the response is accepted.

LIMITATIONS AND NON-CLAIMS

1. Exact Kolmogorov complexity is uncomputable; all implementation claims require proxies.
2. Small margins are representation-sensitive.
3. Observer-relative trust claims reduce here to posterior concentration under a declared observer model.
4. Invariant-constrained completion assumes fully grounded, feasible, positive-margin value-laden specifications.
5. Charter coverage is relative to reachable charter-realizable families.
6. Safety engineering remains necessary in incomplete-SI and approximate-SI regimes.
7. Empirical validation is still required; HorizonMath and related benchmarks are proxies for response-level behavior.

CONCLUSION

PSI/SI supplies a response-level criterion for AGI evaluation: a system succeeds on a well-formed problem when it returns the structurally inevitable semantic completion. Charter Protocol makes the criterion operational through domain boundaries, invariants, output contracts, compiled verifiers, routing, and escalation.

The theorem cluster identifies the conditions under which the surrounding claims become formal. Visible structural gap supports posterior concentration under an observer model. Grounded value invariants become part of admissible problem completion. Reachable charter-realizable families admit finite coverage under charter-authoring assumptions.

The practical program is to build charters that create measurable margins, test those margins with external correctness checks, and record inadequacy when well-formedness, verifier adequacy, or coverage fails.

REFERENCES

1. Yuntao Bai et al. Constitutional AI: Harmlessness from AI feedback. arXiv:2212.08073, 2022. doi:10.48550/arXiv.2212.08073.
2. Ian Broderick. Structural Inevitability: Specification-Side Source Disambiguation for Oracle-Limited Verification. Citium Verification Preprint, 2026.
3. Francois Chollet. On the measure of intelligence. arXiv:1911.01547, 2019. doi:10.48550/arXiv.1911.01547.
4. Paul F. Christiano et al. Deep reinforcement learning from human preferences. NeurIPS 30, 2017. arXiv:1706.03741. doi:10.48550/arXiv.1706.03741.
5. Paul F. Christiano, Buck Shlegeris, and Dario Amodei. Supervising strong learners by amplifying weak experts. arXiv:1810.08575, 2018. doi:10.48550/arXiv.1810.08575.
6. Ben Goertzel and Cassio Pennachin, editors. Artificial General Intelligence. Cognitive Technologies, Springer, 2007. doi:10.1007/978-3-540-68677-4.
7. Peter D. Grunwald. The Minimum Description Length Principle. MIT Press, 2007.
8. Erik Y. Wang, Sumeet Motwani, James V. Roggeveen, Eliot Hodges, Dulhan Jayalath, Charles London, Kalyan Ramakrishnan, Flaviu Cipcigan, Philip Torr, and Alessandro Abate. HorizonMath: Measuring AI progress toward mathematical discovery with automatic verification. arXiv:2603.15617, 2026. doi:10.48550/arXiv.2603.15617.
9. Marcus Hutter. Universal Artificial Intelligence: Sequential Decisions Based on Algorithmic Probability. Springer, 2005.
10. Geoffrey Irving, Paul Christiano, and Dario Amodei. AI safety via debate. arXiv:1805.00899, 2018. doi:10.48550/arXiv.1805.00899.
11. Shane Legg and Marcus Hutter. Universal intelligence: A definition of machine intelligence. Minds and Machines, 17(4):391--444, 2007. doi:10.1007/s11023-007-9079-x.
12. Ming Li and Paul Vitanyi. An Introduction to Kolmogorov Complexity and Its Applications. Third edition, Springer, 2008.
13. Meredith Ringel Morris et al. Position: Levels of AGI for operationalizing progress on the path to AGI. ICML, Proceedings of Machine Learning Research 235:36308--36321, 2024. arXiv:2311.02462. doi:10.48550/arXiv.2311.02462.
14. Long Ouyang et al. Training language models to follow instructions with human feedback. NeurIPS 35, 2022. arXiv:2203.02155. doi:10.48550/arXiv.2203.02155.
15. Rina Panigrahy and Vatsal Sharan. Limitations on safe, trusted, artificial general intelligence. arXiv:2509.21654, 2025. doi:10.48550/arXiv.2509.21654.
16. Jorma Rissanen. Modeling by shortest data description. Automatica, 14(5):465--471, 1978. doi:10.1016/0005-1098(78)90005-5.
17. Ray J. Solomonoff. A formal theory of inductive inference. Parts I and II. Information and Control, 7(1):1--22 and 7(2):224--254, 1964. doi:10.1016/S0019-9958(64)90223-2 and doi:10.1016/S0019-9958(64)90131-7.
18. Alan M. Turing. Computing machinery and intelligence. Mind, 59(236):433--460, 1950.