COUNTEREXAMPLE RETURN INDEX: ROUTING PROVER COMPUTE WITH VALIDATION-MEASURED WITNESS UTILITY

Anonymous authors
Paper under double-blind review

ABSTRACT

An executable counterexample can expose the assignment, violated literal, and dependency that break a conjecture, but extracting and minimizing that witness consumes compute that could instead train the prover or extend proof search. We introduce WITNESSBUDGET, which turns locally corrupted conjectures into checker-confirmed, attributed repair traces, and the Counterexample Return Index (CRI), which predicts their family-level return from validation-only witness yield, probe utility, timeout, cost, and compactness. In eight family-disjoint reruns of a joint three-ecosystem training protocol, all packages start from the same Llama-3.1-70B checkpoint and receive a 512 A100-hour reservation. WITNESSBUDGET reaches 41.7±0.8 on miniF2F, 34.1±0.9 on a ProofNet-derived Isabelle/HOL evaluation, and 56.8±0.6 on DafnyBench, improving over bare counterexample augmentation and witness-free active selection by 1.5–3.1 percentage points with positive paired intervals throughout. Masking the witness narrows the advantage by 0.7–1.3 points, and permutation degrades further, localizing the gain to theorem-aligned witness content. Across 43 unique test families and 344 family–rerun cells, frozen CRI predictions rank solve-rate gain with Spearman 0.77 [0.72,0.81] and identify positive end-to-end efficiency with AUROC 0.88 [0.84,0.91]. Acting on those predictions reallocates 27.2 falsifier hours to search, raises solve rate by 0.6–1.0 points, and improves end-to-end throughput by 1.6–2.2%, while random, difficulty-only, and yield-only gates trail the joint index.

1 INTRODUCTION

Lean enumeration, Nitpick countermodels, and bounded Dafny states produce different executable objects, yet each can be serialized into the same learning event: a local conjecture fails under a checker-confirmed assignment, a specific literal and dependency are responsible, and a repair restores the source theorem. This event is richer than a failed proof or an untyped hard example. It is also costly. Witness discovery, independent replay, and minimality checking occupy the same reservation as model updates and proof search, so using counterexamples effectively requires both an information-bearing trace and a pre-training allocation rule.

WITNESSBUDGET provides the trace. It binds a controlled corruption to a minimal executable witness, violated-literal attribution, a repair label, and a utility-aware admission decision. CRI provides the allocation rule. Before the main run, it combines validation-family witness yield and probe utility with timeout, extraction cost, and witness size; a frozen isotonic map then predicts solve-rate and end-to-end efficiency return for each structural family stratum. Open strata receive attributed witness traces, whereas closed strata use witness-free active selection and transfer their falsifier allocation to search. Verifier-feedback systems improve a proof after interaction, proof-data selection ranks existing training examples, and inference-time routing decides whether to continue an attempt. CRI instead estimates whether producing a new class of executable supervision will repay its cost before the joint training/search run begins.

The intervention is evaluated from a common Llama-3.1-70B initialization in eight rebuilt, family-disjoint reruns. Each method receives one 512 A100-hour reservation for a joint run over the three development ecosystems; model, prover, and falsifier shares are measured outcomes rather

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

than matched channels. B3 preserves corruption and executable witnesses but removes attribution, minimality fields, and utility admission. B4 removes witness production while retaining active selection. Additional masked, permuted, and label-only packages hold trace count and 42.8M tokens fixed, isolating the information carried by theorem-aligned witness structure.

This design yields a closed prediction-to-action result. WITNESSBUDGET improves over B3/B4 by +2.3/+2.6 pp on miniF2F, +1.5/+2.0 pp on the ProofNet-derived port, and +2.5/+3.1 pp on Dafny-Bench; all six hierarchical intervals are positive. CRI ranks held-out family gain with pooled Spearman 0.77, while its high-return tercile gains +3.1 pp compared with +0.6 pp in the low-return tercile. The frozen gate recovers 27.2 falsifier hours, improves solve rate on every benchmark, and exceeds matched random, difficulty-only, and yield-only gates. Fully charged accounting, alternate family graphs, direct content controls, and a DeepSeek-Prover-V1.5 replication preserve the decision.

The primary contribution is a prospective allocation method for executable supervision:

- **Prediction and routing:** CRI defines family-level solve and efficiency return, estimates both from validation-only signals, and converts the frozen estimate into a tested compute gate.
- **Identifiable supervision:** WITNESSBUDGET supplies a cross-logic trace whose assignment, attribution, certificate, repair, and admission fields can be removed independently.
- **Decision evidence:** a family-disjoint protocol across three formal ecosystems and two prover initializations connects content ablations and cost accounting to an allocation that improves both accuracy and throughput.

2 RELATED WORK: FEEDBACK, SELECTION, AND COMPUTE

Language-model theorem proving combines proof generation, retrieval, search, repair, and synthetic supervision. GPT-f and formal statement curricula (Polu & Sutskever, 2020; Polu et al., 2023), PACT (Han et al., 2022), HyperTree search (Lample et al., 2022), Thor (Jiang et al., 2022), Draft-Sketch-Prove (Jiang et al., 2023), LEGO-Prover (Wang et al., 2024), Baldur (First et al., 2023), and LeanDojo (Yang et al., 2023) establish the relevant generation, retrieval, and repair stacks. DeepSeek-Prover-V1.5 uses proof-assistant feedback for reinforcement learning and RMaxTS (Xin et al., 2025); Lean-STaR interleaves informal reasoning with verified steps (Lin et al., 2025); and Goedel-Prover-V2 combines statement synthesis with compiler-conditioned self-correction (Lin et al., 2026). Their feedback changes proof production or search. MPS-Prover selects existing proof data for diversity and stepwise search (Liang et al., 2025), whereas agentic cost routing chooses continue versus restart after failed attempts (Rögnvaldsson et al., 2026). CRI acts earlier and on a different resource: it predicts the net value of first producing attributed counterexamples for a family.

Executable counterexamples drive symbolic refinement in CEGAR and program synthesis (Clarke et al., 2003; Solar-Lezama et al., 2006); curriculum and data-selection methods allocate attention using schedule, difficulty, learning progress, or validation influence (Bengio et al., 2009; Graves et al., 2017; Hacohen & Weinshall, 2019; Robinson et al., 2021; Ren et al., 2018; Swayamdipta et al., 2020); and selective prediction calibrates admission decisions (Geifman & El-Yaniv, 2017; Guo et al., 2017; Lakshminarayanan et al., 2017; Minderer et al., 2021). WITNESSBUDGET+CRI differs along four jointly tested axes: it creates rather than ranks supervision, binds each example to a checker-confirmed failure explanation, charges extraction under the same reservation as training and search, and freezes a family-level routing decision before test evaluation. B3, B4, masking, and permutation separately control those axes.

Appendix Table 5 summarizes these objective and timing differences. B3 controls executable augmentation, B4 controls active selection, and masking and permutation test theorem-aligned witness content; together they isolate CRI's prospective, charged-cost allocation target.

3 RESERVATION LEDGER AND COUNTEREXAMPLE RETURN

The library $\mathcal{T} = \{(\phi_i, \pi_i)\}_{i=1}^{N}$ pairs each true theorem ϕ_i with its proof π_i in formal system $\mathcal{F}$ (e.g., Lean 4). Goals are mapped to tactic sequences by the neural prover P_θ. We assess whether

P_θ's family-held-out solve rate is higher when training uses a collection $\mathcal{C}$ of repair traces bearing witnesses from falsified conjectures, compared with training on true goals alone.

Definition 1 (Allocation instance)**.** *An allocation instance is*

$$\mathfrak{A} = (P_{\theta_0}, H, \Pi, \mathcal{P}, M, \mathcal{U}, \mathcal{L}),$$

where P_{θ_0} is the shared initial checkpoint whose weights are subsequently trained; H is the main-run reservation and hardware; Π partitions theorem families into train, validation, and test components; $\mathcal{P}$ is the set of supervision packages; M contains solve-rate and efficiency metrics; $\mathcal{U}$ is a paired hierarchical uncertainty operator; and $\mathcal{L}$ is a predeclared set of accounting lenses.

For package p, family split s, and cost lens ℓ, define its aggregate fixed-budget return as

$$R_\ell(p, s) = \left(\mathrm{Solve}_{300}(p, s),\ \frac{\text{solved held-out goals}(p, s)}{\text{charged hours}_\ell(p, s)} \right).$$

The first coordinate is *solve-rate return*; the second is *efficiency return*. CRI is evaluated on the finer family–rerun cell (f, r). If family f has n_f goals, let S_{pfr} be its checked solve percentage, $N_{pfr} = n_f S_{pfr}/100$ its solved-equivalent count, and $h_{\ell,pfr}$ its charged hours. Solver and falsifier jobs carry family tags and are charged directly; shared model-side time is allocated in proportion to serialized training tokens from each family, so $\sum_f h_{\ell,pfr}$ equals the run ledger. Define

$$\Delta^S_{fr} = S_{\mathrm{WB},fr} - \max_{q \in \{\mathrm{B3},\mathrm{B4}\}} S_{qfr}, \qquad \Delta^E_{fr} = \frac{100 N_{\mathrm{WB},fr}}{h_{\mathrm{e2e},\mathrm{WB},fr}} - \max_{q \in \{\mathrm{B3},\mathrm{B4}\}} \frac{100 N_{qfr}}{h_{\mathrm{e2e},qfr}}.$$

These are separate scalar labels; $B^\star$ below abbreviates the maximizing comparator for the coordinate being scored, with B3 breaking exact ties. Spearman correlation and MAE score Δ^S, whereas AUROC scores $\mathbf{1}[\Delta^E > 0]$. Package p dominates q in $\mathfrak{A}$ when the hierarchical 95% interval for the primary solve-rate difference is positive and the efficiency-return ordering remains positive under every lens in $\mathcal{L}$. Search-only G/h is a separate diagnostic.

Definition 2 (Corruption Operator)**.** *A corruption operator $c_k \in \mathcal{O} = \{c_1, \ldots, c_K\}$ maps a true theorem ϕ to a candidate conjecture $\tilde{\phi} = c_k(\phi)$. The checker admits $\tilde{\phi}$ exactly when the bounded falsifier returns an explicit $\mathbf{x}^*$ and the trusted evaluator verifies* $\mathrm{Eval}_{\mathcal{F}}(\tilde{\phi}; \mathbf{x}^*) = \mathrm{false}$*. We use five semantically grounded operators:* hypothesis deletion *(c_{del}),* side-condition weakening *(c_{weak}),* quantifier lifting *(c_{lift}),* invariant overgeneralization *(c_{gen}), and* lemma omission *(c_{omit}).*

Definition 3 (Minimal Counterexample)**.** *Given a candidate conjecture $\tilde{\phi}$, a bounded falsifier $\mathcal{B}$ returns either $\perp$ or a checker-confirmed witness package $\mathbf{x}^*$. A successful package contains: (i) a logic-specific falsifying object—a Lean assignment, an Isabelle/Nitpick countermodel, or a Dafny bounded execution state; (ii) violated literals with dependency attribution; and (iii) a minimality certificate under a declared witness order.*

Witnesses are ordered lexicographically by $(B, \|\mathbf{x}\|_0, \|\mathbf{x}\|_1, \mathrm{len}(\mathrm{ser}(\mathbf{x})))$, where B is the finite-domain cap, Nitpick scope vector, or Dafny input/unrolling bound; $\|\mathbf{x}\|_0$ counts nondefault bindings; $\|\mathbf{x}\|_1$ sums normalized integer magnitudes; and the final coordinate is serialized length. Starting from a confirming witness, the extractor decreases B and then the remaining coordinates. The certificate stores the reduction chain and checker outcomes for every immediate predecessor tested; a timeout during predecessor checking marks minimality as uncertified and excludes the certificate-bearing trace. "Witness size" below is the final serialized length. Structural proximity is enforced separately: $\tilde{\phi}$ preserves the source theorem's conclusion head and typed AST skeleton except at one of the declared corruption sites, so the witness describes a local failure rather than an unrelated false statement.

A *witness-bearing repair trace* is a tuple whose entries are the local false conjecture, its minimal counterexample, the repair label, and the admission decision. This trace is the intervention tested by the protocol: the corruption identifies a local failure, the assignment and violated literals expose why it fails, the certificate bounds witness complexity, the label specifies the repair class, and admission allocates fixed compute to traces with measured validation-family utility.

For a benchmark $\mathcal{D}$ with checked reference proofs, let $S(\phi)$ be the named premises used by theorem ϕ. We define $G_{\mathcal{D}} = (V, E)$ with $V = \mathcal{D}$ and $(\phi_i, \phi_j) \in E$ whenever $S(\phi_i) \cap S(\phi_j) \neq \emptyset$; each

connected component is a theorem family. Before assigning components, the same normalized statement hashes, import identifiers, and named-premise keys are computed over the union of development and evaluation sources; any cross-pool edge is kept in one component and therefore cannot cross a split. Splitting components makes direct local-premise transfer unavailable at test time. Four alternative miniF2F graph constructions in Appendix Table 17 test sensitivity to the family definition.

For the post-hoc miniF2F decomposition only, we also use twelve coarse semantic categories (e.g., ring/module and real analysis). A category pools multiple disconnected premise-graph components and may therefore have development and test members without sharing any component. Category omission removes category-matched development traces; it never removes, or trains on, a test component. These categories are not CRI strata and do not alter the family-held-out protocol.

The per-rerun chronology is fixed: construct the union graph; split development components into train and validation; run validation-only witness extraction and probe updates; fit and freeze CRI maps and the optional gate; execute the 512-hour train/search reservation; and only then score the fixed evaluation components. Validation cost is included in the end-to-end and solver-heavy lenses, not in the 512-hour main-run reservation.

CRI is computed for a structural family stratum g after the charged validation stage and before full curriculum training:

$$\mathrm{CRI}(g) = \frac{\widehat{y}_g(1-\widehat{t}_g)\max(\widehat{u}_g, 0)}{\widehat{c}_g[1+\log(1+\widehat{m}_g)]}. \tag{1}$$

Here $\widehat{y}_g$ is checker-confirmed witness yield conditional on a completed falsifier call, $\widehat{t}_g$ is timeout/unknown frequency, $\widehat{u}_g$ is admitted-minus-rejected gain on disjoint validation probe goals, $\widehat{c}_g$ is mean charged extraction time per attempt, and $\widehat{m}_g$ is median witness size. Yield and timeout use add-one smoothing; cost is floored at one millisecond; nonpositive utility sets CRI to zero. Strata require at least 20 completed validation attempts. A family is assigned to the nearest supported validation stratum using the predeclared signature of statement logic, applicable operators, AST-size quartile, and import-graph degree quartile; ties use lower AST quartile, then lexicographic signature order. Unsupported strata close and receive B4. The signature uses theorem statements and imports available before training; reference-proof premises, test witnesses, and test outcomes are excluded.

Validation labels come from matched WB, B3, and B4 micro-branches that share the same pre-branch checkpoint, probe goals, and stopping rule. Their family-level solve and efficiency labels use the scalar definitions above; branch, probe, fitting, and threshold-evaluation time is charged to the corresponding end-to-end ledger. Separate one-dimensional isotonic regressions map CRI to Δ^S and Δ^E over $B^\star$, while the routing regression maps CRI to B4-relative efficiency. Leave-one-validation-family-out fits generate calibration diagnostics; the final maps are refitted on all validation families and frozen once per rerun. The gate opens stratum g iff predicted B4-relative efficiency is positive; exact-zero predictions close. Closed strata receive B4-style witness-free active selection, and saved falsifier time is reassigned to prover search. Test-family outcomes enter only after every map and gate decision is frozen.

The triage policy optimizes the validation surrogate

$$\max_{\psi}\ \mathbb{E}_{(\mathbf{z},\mathcal{Q})\sim\mathcal{D}_{\mathrm{val}}}\left[\Delta_{\mathrm{gain}}(\theta^*(\psi);\mathbf{z},\mathcal{Q})\right] - \lambda\, C(\mathcal{C}_\psi), \quad \theta^*(\psi) = \arg\min_{\theta} \mathcal{L}_{\mathrm{train}}(\theta;\mathcal{C}_\psi), \tag{2}$$

where $\mathcal{Q}$ is a disjoint validation probe set, $\mathcal{C}_\psi$ is the curriculum selected by policy ψ, C is charged extraction and update cost, and $\mathcal{L}_{\mathrm{train}}$ combines proof-generation loss on true goals with repair-prediction loss on falsified instances. Fixed test families estimate transfer only after ψ, CRI maps, and the gate are frozen.

4 Generating Witness-Bearing Repair Traces

WitnessBudget supplies the information-bearing intervention whose return CRI estimates. It binds controlled corruption, executable witness extraction, repair attribution, and utility-aware admission into one trace; CRI operates at the family-stratum level, while triage operates among candidate traces inside an admitted stratum.

4.1 Stage 1: Controlled Corruption

We encode a true theorem ϕ in a typed abstract syntax tree (AST) that makes its hypotheses, quantifier patterns, and library links explicit; each operator $c_k \in \mathcal{O}$ then proposes candidate conjectures from this representation. The five operators comprise c_{del} for hypothesis deletion, c_{weak} for side-condition weakening, c_{lift} for quantifier lifting, c_{gen} for invariant overgeneralization, and c_{omit} for lemma omission. Respectively, these delete an antecedent hypothesis, weaken an essential side condition, change an existential quantifier to a universal one or enlarge a bounded domain, substitute an overly general version for an invariant, or remove a proof-context dependency on a lemma. A fixed balanced allocation draws at most three distinct valid sites per operator and at most 15 candidates per theorem; unused slots for one operator are not transferred to another. Site order within an operator is sampled once from the seeded temperature-0.8 proposer. This rule is fully specified, prevents a high-yield operator from occupying the candidate pool, and keeps the intervention local.

4.2 Stage 2: Bounded Falsification and Witness Extraction

Each candidate $\tilde{\phi}$ is submitted to $\mathcal{B}$ with $\tau = 60$ seconds for miniF2F and the ProofNet-derived port and $\tau = 30$ seconds for DafnyBench. Lean mutations are restricted to a reified decidable finite fragment: an enumeration wrapper extracts assignments, and `decide` independently checks the resulting closed proposition rather than producing the witness itself. The 371 official Lean 3 ProofNet statements are evaluated through an Isabelle/HOL port that preserves normalized types, binders, logical connectives, constants, and source IDs; ported statements must typecheck and pass this structural correspondence check before Nitpick (Blanchette & Nipkow, 2010) searches a bounded countermodel. Dafny candidates are compiled through Dafny's verification-condition pipeline to Boogie/Z3 under explicit input and loop-unrolling bounds; Dafny is not treated as a built-in model checker. In every ecosystem, the returned assignment, countermodel, or bounded state is replayed by an evaluator independent of the search call. The result is either a checker-confirmed minimal witness package or $\perp$ (no witness within the bound, timeout, or unknown); only confirmed packages enter the trace pool.

The common trace serializer records the logic tag, source statement ID, bound, typed bindings, violated literal and dependency slice, minimality reduction chain, independent replay outcome, and repair operator. Comparing $\tilde{\phi}$ with ϕ supplies the corruption label. This interface makes Lean assignments, Nitpick countermodels, and Dafny bounded states comparable without claiming that their search procedures are identical.

4.3 Stage 3: Cost-Aware Triage

Counterexamples vary substantially in downstream utility. The triage model T_ψ consumes a full trace $\mathbf{z}$—the corrupted-statement AST, witness metadata, attribution, repair label, and embedded solver budget b—and predicts three quantities:

$$T_\psi(\mathbf{z}, b) = \Big(P(\text{probe solve} \mid \mathbf{z}, b),\ P(\text{confirmed falsity} \mid \mathbf{z}, b),\ \mathbb{E}[\widetilde{\Delta}_{\mathrm{gain}} \mid \mathbf{z}, b] \Big), \tag{3}$$

where $\Delta_{\mathrm{gain}} \in [-1, 1]$ is the change in solve fraction on related true goals after a fixed micro-update and $\widetilde{\Delta}_{\mathrm{gain}} = (\Delta_{\mathrm{gain}} + 1)/2$. The model is a 3-layer transformer over the AST plus typed witness features, trained with

$$\mathcal{L}_{\mathrm{triage}} = \mathcal{L}_{\mathrm{BCE}}^{\mathrm{probe}} + \mathcal{L}_{\mathrm{BCE}}^{\mathrm{falsify}} + \lambda_{\mathrm{cal}} \mathcal{L}_{\mathrm{Brier}}^{\mathrm{gain}} + \lambda_{\mathrm{ece}} \mathcal{L}_{\mathrm{ECE}}, \tag{4}$$

with all weights fixed on validation families. For the continuous target, Brier is mean squared error on $\widetilde{\Delta}_{\mathrm{gain}}$ and ECE bins predicted normalized gain, then compares each bin's mean prediction with mean realized normalized gain. Each candidate receives matched admit and reject micro-updates from the same pre-branch checkpoint and is evaluated on a disjoint, fixed validation-family probe set $\mathcal{Q}_0$; their difference defines the pre-run utility used by CRI. Round-indexed probe labels update T_ψ during training but never revise frozen CRI scores. The admission threshold $\epsilon = 0.01$ is fixed from the pilot before the eight reported seeds and is not selected from their calibration curves. Appendix Table 21 compares admitted and rejected traces directly.

4.4 Degeneracy Monitoring and Balanced Allocation

The corruption pool can collapse to shallow perturbations such as trivially flipped inequalities. The per-operator quota above is the preventive intervention; its ablation pools all valid sites, follows the same seeded proposer order, and takes the first 15 irrespective of operator. We separately monitor shallow-perturbation rate, trivial-refutation rate, and repair triviality. The predeclared $\delta = 0.4$ is a diagnostic alarm rather than an optimizer switch: crossing it flags the run for inspection, while training remains identical across seeds. Appendix Fig. 3 reports the monitored trajectories.

Before falsified conjectures enter prover training, the prover receives a 3-epoch warm-start using true theorem–proof pairs; the triage model's warm-start uses a pilot set pairing falsified conjectures with observed gain labels. A fixed mix of true and falsification examples is maintained throughout the main loop's 8 curriculum rounds. Each method–rerun pair is one joint multi-ecosystem run, not a per-benchmark run or an average of three ledgers. Its 512 A100-hour reservation charges model-side training/inference, prover-solver calls, and falsifier time across the union of the three development libraries; the resulting checkpoint is scored separately on all three test pools. Appendix Table 7 records main-loop occupancy, and Table 16 additionally charges validation branches under end-to-end and solver-heavy lenses.

5 Experiments

5.1 Benchmarks, controls, and protocol

We evaluate miniF2F (Zheng et al., 2022) in Lean 4 (244 official validation and 244 test problems), the 371-statement ProofNet collection (Azerbayev et al., 2023) through the Isabelle/HOL port specified above, and DafnyBench (Loughridge et al., 2025) (782 evaluation tasks) through bounded Dafny/Boogie/Z3 verification conditions. "ProofNet" below always denotes this port, not the official Lean 3 implementation. The miniF2F validation pool and benchmark-excluded Isabelle/HOL and Dafny libraries supply development families. For each rerun, the union graph is built before an 80/20 component split into train and validation; any development component connected to a fixed evaluation component is withheld from training. The complete evaluation pools—244, 371, and 782 tasks—define the reported denominators and are used only for held-out scoring. MATH (Hendrycks et al., 2021) supplies a fixed 5,000-answer non-executable specificity probe.

All primary methods start from the same Llama-3.1-70B (Grattafiori et al., 2024) checkpoint and run on an identical 8×NVIDIA A100 80GB node with CUDA 12.4. A method–seed cell is one joint run over all three development libraries with a 512 A100-hour main-run reservation (64 node-hours), not three separate budgets. Per-channel allocations are outcomes of the method. The primary metric is checked solve rate at $b = 300$s. Main-table goals per solver-hour (G/h) uses wall-clock proof-search service time and is a search-only diagnostic; Appendix Table 7 reports aggregate A100 occupancy. Efficiency return instead divides solved-equivalents by every charged hour under the named lens: miniF2F WitnessBudget, for example, gives $101.7/538.8 \times 100 = 18.9$ solved goals per 100 end-to-end hours.

Proof search is held fixed across packages: theorem serialization, the retrieval index and context builder, decoder configuration, tactic checker, and per-goal stopping rule are shared within a backbone. Goals enter a seeded round-robin queue; the batcher fills accelerator batches from the next live goals, and each goal stops on a checked proof or its 300-second cap. Reallocated falsifier time continues this same queue with the next seeded attempt stream; it changes the number of attempted expansions, not retrieval, decoding, or acceptance criteria. Search also stops when the joint reservation is exhausted.

We run eight predetermined split/training seeds (`17, 29, 43, 71, 101, 137, 181, 239`). Paired 95% intervals use 10,000 hierarchical draws that resample reruns and then family IDs, carrying all theorems in a sampled family together; the same draw is applied to every method. This preserves the repeated identity of the 12 miniF2F, 14 ProofNet, and 17 DafnyBench test families across reruns. Means and standard deviations summarize the eight rebuilt splits.

CRI evaluation freezes each prediction before scoring the corresponding test family. The 43 unique test families crossed with eight reruns produce 344 cells; uncertainty resamples family identity and rerun as crossed clusters, and isotonic maps are refitted only from resampled validation families. Table 3 evaluates $B^\star$-relative prediction: Spearman correlation measures solve-rate-return ranking,

MAE measures calibrated solve-rate-gain prediction, and AUROC discriminates positive efficiency return under end-to-end accounting. The allocation experiment separately applies the frozen B4-relative efficiency map.

B1 performs truth-only fine-tuning on the train components. B2 uses the same true-goal multiset but orders examples by validation-frozen proof loss and AST size, cycling through all examples once per round. B3 applies the same corruption sites and checker as WITNESSBUDGET, serializes the false conjecture plus bare witness assignment, and admits confirmed traces in seeded generation order without attribution, minimality fields, or utility filtering. B4 spends no falsifier time and selects true train goals by the triage prove-head uncertainty, then transfers the unused channel reservation to proof search. All four controls use the same eight-round update schedule, 0.7/0.3 true/auxiliary mix where an auxiliary stream exists, $b = 300$s evaluation timeout, and 512-hour reservation. The information controls in Table 2, unlike B1–B4, additionally match trace count and the 42.8M-token budget exactly. DeepSeek-Prover-V1.5 (Xin et al., 2025) repeats B3/B4/WITNESSBUDGET on a distinct prover initialization and recomputes all validation-only CRI statistics before freezing its own maps.

5.2 MATCHED-RESERVATION GAINS ON HELD-OUT FAMILIES

Table 1 gives the primary solve-rate comparison. Relative to B3/B4, WITNESSBUDGET gains +2.3/+2.6 pp on miniF2F, +1.5/+2.0 pp on ProofNet, and +2.5/+3.1 pp on DafnyBench. Solver-hour G/h improves by 8.1–9.5% over the stronger B3/B4 value on each benchmark. Fully charged efficiency gains are smaller but remain positive under both predeclared lenses (Appendix Tables 16 and 27).

Table 1: Family-held-out performance at $b = 300$s from a common Llama-3.1-70B initialization and 512 A100-hour main-run reservation. Values are mean $\pm$ std over eight rebuilt family splits. Rate is solve percentage; G/h is search-only solver throughput.

	miniF2F		ProofNet		DafnyBench	
Method	Rate	G/h	Rate	G/h	Rate	G/h
Truth-only SFT (B1)	38.2±1.0	4.19	31.3±1.0	3.77	52.4±0.7	6.14
Witness-free curriculum (B2)	38.9±0.9	4.34	31.8±0.9	3.87	53.1±0.8	6.29
Cex augmentation (B3)	39.4±0.8	4.50	32.6±1.0	4.01	54.3±0.7	6.53
Active selection (B4)	39.1±1.0	4.40	32.1±1.1	3.90	53.7±0.9	6.38
WITNESSBUDGET	41.7±0.8	4.91	34.1±0.9	4.39	56.8±0.6	7.06

Appendix Tables 14 and 7 report solved-equivalents and the complete joint ledger. WITNESSBUDGET assigns 118.6 of 512 hours to falsification, compared with 16.8 for B3; this is reservation matching, not channel matching. After validation branches and CPU occupancy are added, WITNESSBUDGET retains the highest solved-goal throughput under both inclusive lenses (Appendix Tables 16 and 27).

5.3 WHAT WITNESS STRUCTURE CARRIES THE GAIN

We retain the false conjecture and repair target while removing only the witnessing assignment, violated-literal attribution, and minimality certificate (masked); exchange witnesses within benchmark, operator, family-size, and token-length buckets (permuted); or retain corruption labels without witness content (label-only). All variants use the same eight seeds, 512-hour reservation, trace count, 42.8M-token budget, and auxiliary-charge rules. Table 2 shows that generic augmentation and selection recover part of the gain, but preserving the structured witness trace is consistently strongest. The narrowest full-versus-masked gap occurs on ProofNet (+0.7 pp), while miniF2F and DafnyBench retain +1.3 and +0.9 pp gaps.

The content controls separate witness presence from trace volume. Permutation loses more than masking on all three benchmarks, indicating that theorem-aligned attribution matters beyond the existence of extra structured tokens. Appendix Table 22 reports positive paired intervals against every information control. Component ablations in Tables 8 and 23 identify executable falsification as the

Table 2: Witness-information isolation under matched 512-hour reservations, auxiliary-charge rules, and token counts. Cells are eight-rerun solve rates (%). Full-control solved-equivalents, throughput, tokens, and paired intervals appear in Appendix Table 22.

Supervision package	miniF2F	ProofNet	DafnyBench
Cex augmentation (B3)	39.4	32.6	54.3
Active selection (B4)	39.1	32.1	53.7
Self-play augmentation	39.7	32.9	55.4
Verifier-weighted curriculum	39.9	33.1	55.6
Witness masked	40.4	33.4	55.9
Witness permuted	39.9	33.0	55.5
Label only	39.5	32.7	54.9
WITNESSBUDGET	41.7	34.1	56.8

largest contributor on every benchmark; minimality, triage, repair attribution, and balanced operator allocation provide distinct additional gains.

5.4 ROBUSTNESS ACROSS COSTS, SPLITS, AND BACKBONES

Appendix Table 9 consolidates the stress tests. All six B3/B4 hierarchical intervals are positive; the smallest inclusive-accounting advantage is +0.4 solved goals/100h; and alternate family graphs retain at least +1.4/+1.7 pp gains. Eight-rerun DeepSeek intervals remain positive, with [+0.1,+1.7] the tightest. In contrast, the MATH answer-level intervals include zero, tying return to checker-visible witnesses rather than generic mathematical fine-tuning.

5.5 HELD-OUT RETURN PREDICTION AND BUDGET ROUTING

Table 3: Frozen CRI predictions of held-out family return over $B^\star$. The 344 cells contain 43 unique families crossed with eight reruns; intervals preserve both cluster identities. MAE is solve-rate-gain error in percentage points; AUROC classifies positive end-to-end efficiency.

Backbone / benchmark	Cells	Spearman ρ [95% interval]	MAE	AUROC
Llama / miniF2F	96	0.83 [0.75,0.89]	0.48	0.92 [0.86,0.96]
Llama / ProofNet	112	0.69 [0.58,0.78]	0.61	0.83 [0.75,0.89]
Llama / DafnyBench	136	0.78 [0.69,0.85]	0.54	0.89 [0.83,0.94]
Llama / pooled	344	0.77 [0.72,0.81]	0.55	0.88 [0.84,0.91]
DeepSeek / pooled	344	0.71 [0.65,0.76]	0.63	0.84 [0.80,0.88]

CRI ranks family-level return on every benchmark, with the tightest relation on miniF2F and the noisiest on ProofNet. Recomputing validation statistics and refitting the maps on DeepSeek retains a pooled correlation of 0.71 (Appendix Table 18). The predicted terciles form a monotonic +0.6 to +3.1 pp gain gradient; removing trace utility or timeout-adjusted cost reduces pooled correlation to 0.62 and 0.57 (Appendix Table 11).

Table 4: Matched gate comparison pooled across benchmarks. Every gate uses the same 91.4-hour falsifier allocation; intervals compare gated with uniform WITNESSBUDGET over eight reruns.

Gate signal	Δ solve rate [95% interval]	Throughput Δ [95% interval]
Random	-0.2 [-0.7,+0.3]	+0.1% [-0.5,+0.6]
Truth-only difficulty	+0.1 [-0.3,+0.5]	+0.4% [-0.2,+1.0]
Witness yield	+0.3 [-0.1,+0.7]	+0.8% [+0.1,+1.5]
CRI joint index	+0.8 [+0.5,+1.1]	+1.8% [+1.3,+2.2]

Table 4 isolates the allocation value of the joint return estimate: random routing does not improve either endpoint, and neither generic difficulty nor raw witness availability recovers CRI's gain. In

the benchmark-level ledger, CRI reduces falsifier allocation from 118.6 to 91.4 hours, transfers 27.2 hours to the fixed search queue, raises solve rate by 0.6–1.0 points, and improves end-to-end throughput by 1.6–2.2% (Appendix Table 12).

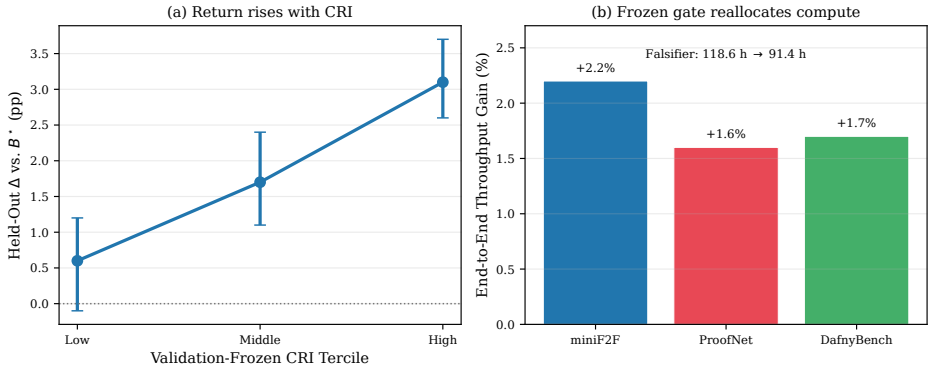

Figure 1: Validation-frozen CRI strata and the resulting allocation. Left: held-out solve-rate gain over $B^\star$ rises across CRI terciles; bars show paired 95% intervals. Right: the CRI gate reduces falsifier occupancy from 118.6 to 91.4 hours and raises end-to-end throughput on all three benchmarks. All plotted values are reported in Appendix Tables 11 and 12.

The left panel makes the prospective relation visible without using test difficulty as the horizontal axis. The miniF2F semantic-category decomposition in Appendix Table 13 supplies a complementary mechanism view: categories 1–9 account for 8.2 of 8.5 additional mean solved-equivalents, while categories 10–12 carry 59 goals but contribute 0.3. Operator quality, witness compactness, and timeout jointly explain why truth-only difficulty is insufficient.

6 Discussion: From Prediction to Allocation

Executable counterexamples repay compute when validation shows that cheap, compact witnesses improve disjoint probe goals; frequent timeouts and large witnesses instead favor additional proof search. WitnessBudget exploits trace-level utility inside an open structural stratum, and CRI decides whether to open that stratum before the main run. The gate validates the consequence by transferring 27.2 hours from falsification to an otherwise unchanged search queue while increasing both solve rate and end-to-end throughput.

The demonstrated regime comprises three checker-backed witness interfaces, 43 held-out family identities, the reported premise graphs, and two prover initializations. Isabelle higher-order counter-model search yields the noisiest CRI predictions, whereas the non-executable MATH probe lacks the executable-task ordering. Validation micro-branches add 26.8 hours to the end-to-end WitnessBudget total; under that inclusive lens and the solver-heavy alternative, the throughput ordering remains positive. Thus the fixed reservation measures the allocation intervention, and the inclusive lenses verify its net return.

The resulting policy is direct: select falsification where CRI predicts positive B4-relative efficiency, train on attributed witness traces in those strata, and spend the recovered reservation on search elsewhere. Alternate family graphs, information controls, and the DeepSeek replication connect that policy to measured witness return rather than one partition or checkpoint.

7 Conclusion

Checker-confirmed failure explanations become a useful prover resource when their production is treated as an allocation decision. WitnessBudget supplies the attributed executable trace; CRI predicts its family-level return before the joint run; and the frozen gate converts that prediction into more search, higher solve rate, and 1.6–2.2% greater end-to-end throughput. The combined method closes the loop from counterexample content to prospective return and then to measured compute allocation across three formal ecosystems.

AI USE STATEMENT

Generative AI tools assisted manuscript preparation, including revision of prose and presentation, refinement of the title, abstract, and introduction, reference checking and formatting, and submission-readiness checks. This preparation process did not generate, validate, or alter the reported empirical measurements. The final submitted manuscript remains the responsibility of its authors.

REFERENCES

Zhangir Azerbayev, Bartosz Piotrowski, Hailey Schoelkopf, Edward W. Ayers, Dragomir Radev, and Jeremy Avigad. ProofNet: Autoformalizing and Formally Proving Undergraduate-Level Mathematics, 2023. URL https://arxiv.org/abs/2302.12433.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th International Conference on Machine Learning*, pp. 41–48, 2009. doi: 10.1145/1553374.1553380. URL https://mlanthology.org/icml/2009/bengio2009icml-curriculum/.

Jasmin Christian Blanchette and Tobias Nipkow. Nitpick: A Counterexample Generator for Higher-Order Logic Based on a Relational Model Finder. In *Interactive Theorem Proving*, volume 6172 of *Lecture Notes in Computer Science*, pp. 131–146, 2010. doi: 10.1007/978-3-642-14052-5_11. URL https://doi.org/10.1007/978-3-642-14052-5_11.

Edmund Clarke, Orna Grumberg, Somesh Jha, Yuan Lu, and Helmut Veith. Counterexample-guided abstraction refinement for symbolic model checking. *Journal of the ACM*, 50(5):752–794, 2003. doi: 10.1145/876638.876643. URL https://dl.acm.org/doi/10.1145/876638.876643.

Emily First, Markus N. Rabe, Talia Ringer, and Yuriy Brun. Baldur: Whole-Proof Generation and Repair with Large Language Models. In *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 1229–1241, 2023. doi: 10.1145/3611643.3616243. URL https://dl.acm.org/doi/10.1145/3611643.3616243.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, volume 30, pp. 4878–4887, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/4a8423d5e91fda00bb7e46540e2b0cf1-Abstract.html.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, et al. The Llama 3 Herd of Models. *CoRR*, abs/2407.21783, 2024. URL https://arxiv.org/abs/2407.21783.

Alex Graves, Marc G. Bellemare, Jacob Menick, Rémi Munos, and Koray Kavukcuoglu. Automated curriculum learning for neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1311–1320, 2017. URL https://proceedings.mlr.press/v70/graves17a.html.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330, 2017. URL https://proceedings.mlr.press/v70/guo17a.html.

Guy Hacohen and Daphna Weinshall. On the power of curriculum learning in training deep networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2535–2544, 2019. URL https://proceedings.mlr.press/v97/hacohen19a.html.

Jesse Michael Han, Jason Rute, Yuhuai Wu, Edward W. Ayers, and Stanislas Polu. Proof artifact co-training for theorem proving with language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=rpxJc9j04U.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring Mathematical Problem Solving with the MATH Dataset. In *NeurIPS 2021 Track on Datasets and Benchmarks*, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/be83ab3ecd0db773eb2dc1b0a17836a1-Abstract-round2.html.

Albert Qiaochu Jiang, Wenda Li, Szymon Tworkowski, Konrad Czechowski, Tomasz Odrzygóźdź, Piotr Miłoś, Yuhuai Wu, and Mateja Jamnik. Thor: Wielding Hammers to Integrate Language Models and Automated Theorem Provers. In *Advances in Neural Information Processing Systems*, volume 35, pp. 8360–8373, 2022. doi: 10.52202/068431-0608. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/377c25312668e48f2e531e2f2c422483-Abstract-Conference.html.

Albert Qiaochu Jiang, Sean Welleck, Jin Peng Zhou, Wenda Li, Jiacheng Liu, Mateja Jamnik, Timothée Lacroix, Guillaume Lample, and Yuhuai Wu. Draft, sketch, and prove: Guiding formal theorem provers with informal proofs. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=SMa9EAovKMC.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30, pp. 6402–6413, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/9ef2ed4b7fd2c810847ffa5fa85bce38-Abstract.html.

Guillaume Lample, Marie-Anne Lachaux, Thibaut Lavril, Xavier Martinet, Amaury Hayat, Gabriel Ebner, Aurélien Rodriguez, and Timothée Lacroix. HyperTree Proof Search for Neural Theorem Proving. In *Advances in Neural Information Processing Systems*, volume 35, pp. 26337–26349, 2022. doi: 10.52202/068431-1910. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/a8901c5e85fb8e1823bbf0f755053672-Abstract-Conference.html.

Zhenwen Liang, Linfeng Song, Yang Li, Tao Yang, Haitao Mi, and Dong Yu. MPS-Prover: Advancing Stepwise Theorem Proving by Multi-Perspective Search and Data Curation. In *Advances in Neural Information Processing Systems*, volume 38, pp. 137891–137916, 2025. doi: 10.52202/085713-4603. URL https://proceedings.nips.cc/paper_files/paper/2025/hash/c95c9a9ecabd6135bb4dc3d67d49d8be-Abstract-Conference.html.

Haohan Lin, Zhiqing Sun, Sean Welleck, and Yiming Yang. Lean-STaR: Learning to Interleave Thinking and Proving. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/a5357781c204d4412e44ed9cbcdb08d5-Abstract-Conference.html.

Yong Lin, Shange Tang, Bohan Lyu, Ziran Yang, Jui-Hui Chung, Haoyu Zhao, Lai Jiang, Yihan Geng, Jiawei Ge, Jingruo Sun, Jiayun Wu, Jiri Gesi, Ximing Lu, David Acuna, Kaiyu Yang, Hongzhou Lin, Yejin Choi, Danqi Chen, Sanjeev Arora, and Chi Jin. Goedel-Prover-V2: Scaling Formal Theorem Proving with Scaffolded Data Synthesis and Self-Correction. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=j4C0nALrgK.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Chloe R. Loughridge, Qinyi Sun, Seth Ahrenbach, Federico Cassano, Chuyue Sun, Ying Sheng, Anish Mudide, Md Rakib Hossain Misu, Nada Amin, and Max Tegmark. DafnyBench: A Benchmark for Formal Software Verification. *Transactions on Machine Learning Research*, 2025. URL https://mlanthology.org/tmlr/2025/loughridge2025tmlr-dafnybench/.

Matthias Minderer, Josip Djolonga, Rob Romijnders, Frances Hubis, Xiaohua Zhai, Neil Houlsby, Dustin Tran, and Mario Lucic. Revisiting the calibration of modern neural networks. *Advances in Neural Information Processing Systems*, 34:15682–15694, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/hash/8420d359404024567b5aefda1231af24-Abstract.html.

Stanislas Polu and Ilya Sutskever. Generative Language Modeling for Automated Theorem Proving. *CoRR*, abs/2009.03393, 2020. URL https://arxiv.org/abs/2009.03393.

Stanislas Polu, Jesse Michael Han, Kunhao Zheng, Mantas Baksys, Igor Babuschkin, and Ilya Sutskever. Formal mathematics statement curriculum learning. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=-P7G-8dmSh4.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4334–4343, 2018. URL https://proceedings.mlr.press/v80/ren18a.html.

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=CR1XOQ0UTh-.

Kári Rögnvaldsson, Chenhao Sun, Jasper Dekoninck, and Martin Vechev. Optimizing the cost-quality tradeoff of agentic theorem provers in lean. *arXiv preprint arXiv:2606.04883*, 2026. URL https://arxiv.org/abs/2606.04883.

Armando Solar-Lezama, Liviu Tancau, Rastislav Bodik, Sanjit Seshia, and Vijay Saraswat. Combinatorial sketching for finite programs. In *Proceedings of the 12th International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 404–415, 2006. doi: 10.1145/1168857.1168907. URL https://dl.acm.org/doi/10.1145/1168857.1168907.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 9275–9293. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.746. URL https://aclanthology.org/2020.emnlp-main.746/.

Haiming Wang, Huajian Xin, Chuanyang Zheng, Zhengying Liu, Qingxing Cao, Yinya Huang, Jing Xiong, Han Shi, Enze Xie, Jian Yin, Zhenguo Li, and Xiaodan Liang. LEGO-Prover: Neural theorem proving with growing libraries. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=3f5PALef5B.

Huajian Xin, Z. Z. Ren, Junxiao Song, Zhihong Shao, Wanjia Zhao, Haocheng Wang, Bo Liu, Liyue Zhang, Xuan Lu, Qiushi Du, Wenjun Gao, Haowei Zhang, Qihao Zhu, Dejian Yang, Zhibin Gou, Z. F. Wu, Fuli Luo, and Chong Ruan. DeepSeek-Prover-V1.5: Harnessing Proof Assistant Feedback for Reinforcement Learning and Monte-Carlo Tree Search. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=I4YAIwrsXa.

Kaiyu Yang, Aidan Swope, Alex Gu, Rahul Chalamala, Peiyang Song, Shixing Yu, Saad Godil, Ryan J. Prenger, and Animashree Anandkumar. LeanDojo: Theorem Proving with Retrieval-Augmented Language Models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 21573–21612, 2023. doi: 10.52202/075280-0944. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/4441469427094f8873d0fecb0c4e1cee-Abstract-Datasets_and_Benchmarks.html.

Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. miniF2F: A Cross-System Benchmark for Formal Olympiad-Level Mathematics. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=9ZPegFuFTFv.

A Extended Experimental Details

This appendix specifies the implementation, raw denominators, control packages, rerun structure, robustness tests, and return diagnostics.

The prover warm-start uses AdamW (Loshchilov & Hutter, 2019) with learning rate 3×10^{-5}, cosine annealing over 3 true-only epochs, batch size 32, and gradient accumulation over 4 steps. The mixed phase then runs for $R = 8$ curriculum rounds with true/auxiliary mix $\beta = 0.7/0.3$. The triage model uses a separate optimizer with learning rate 10^{-4} and loss weights $\lambda_{\text{cal}} = 0.5$, $\lambda_{\text{ece}} = 0.3$. Candidate generation uses temperature $T = 0.8$, a nontransferable quota of at most $m = 3$ sites per operator (15 candidates per theorem), and the pilot-fixed admission threshold $\epsilon = 0.01$; $\delta = 0.4$ is the monitoring alarm. Falsifier timeout is $\tau = 60$s for miniF2F and ProofNet and $\tau = 30$s for DafnyBench.

Toolchain versions are Lean 4.3.0, Isabelle2024, Nitpick 2024.1, Dafny 4.4.0, and Z3 4.12.4; Dafny uses its standard Boogie verification-condition backend. Lean witness search reifies the finite candidate fragment and enumerates assignments before `decide` checks the closed instantiation. Isabelle port validation compares source and target normalized signatures (binders, types, logical heads, constants, and source ID), after which Nitpick models are serialized and replayed through the generated ground evaluator. Dafny witness search fixes input domains and loop-unrolling depth before verification-condition generation; a returned Z3 model is replayed against the generated postcondition evaluator. Search success alone is never an admission criterion.

The Llama runs use bfloat16 training, activation checkpointing, AdamW weight decay 0.1, gradient clipping at 1.0, and deterministic data ordering within seed. Families are connected components of named-premise overlap. The graph is constructed over development and fixed-evaluation sources together, then development-only components are assigned 80/20 to train/validation while evaluation components remain test-only. Normalized statement hashes and import keys are checked using the same union before any trace is generated.

The eight split/training seeds are fixed before comparison. Each hierarchical bootstrap draws reruns and family identities with replacement and carries all observations belonging to the sampled clusters; all methods share each draw. The resulting 10,000 paired differences determine percentile intervals. For CRI, the 12/14/17 unique miniF2F/ProofNet/DafnyBench family identities are crossed with all eight reruns, and each bootstrap refits isotonic maps from validation observations only. Alternative accounting, family-graph variants, information controls, and the DeepSeek replication retrain under their stated intervention while preserving the $b = 300$s evaluator.

Table 5: Objective-level contrast with adjacent supervision and allocation strategies. "When" is the point at which selection acts; CRI predicts net witness value before the main training/search run.

Strategy	Primary signal	When	Target
Verifier-feedback provers (Xin et al., 2025; Lin et al., 2026)	proof states, compiler errors	train/search	proof success
Proof-data selection (Liang et al., 2025)	proof quality/diversity	training	data efficiency
Agentic cost routing (Rögnvaldsson et al., 2026)	failed trajectory, next-attempt cost	inference	continue/restart
CEGAR / synthesis (Clarke et al., 2003; Solar-Lezama et al., 2006)	executable counterexample	refinement	symbolic candidate
WITNESSBUDGET + CRI	attributed witness utility and cost	pre-main-run	family return and routing

The table distinguishes evidence used to improve a current proof from evidence used to reserve compute for creating supervision. CRI is the only row whose action precedes the main run and whose target is net family return.

The operator-balance ablation removes only the nontransferable per-operator quota and takes the first 15 valid sites in the same seeded pooled order. Its higher shallow rate connects the defined pool intervention to the monitored degeneracy outcome.

Table 6: Control contract. Every package starts from the same checkpoint and uses the same train families, eight rounds, evaluator, and 512-hour main-run reservation. "Auxiliary object" is added beside the common true-proof stream; saved channel time is reassigned to proof search. Exact trace/token matching applies to the lower four information controls and WITNESSBUDGET.

Package	Candidate and selection rule	Auxiliary object	Charging rule
B1 truth-only	seeded true-goal order	none	model + prover
B2 curriculum	validation-frozen loss/AST order	true theorem/proof	model + prover
B3 cex augmentation	same corruptions; confirmed in seeded order	false statement + bare assignment	model + prover + falsifier
B4 active selection	prove-head uncertainty on true goals	selected true theorem/proof	model + prover
Self-play	sampled failed proof completed by checker feedback	failed/repair proof pair	matched traces/tokens
Verifier weighted	all candidate traces weighted by checker confidence	weighted false/repair trace	matched traces/tokens
Masked/permuted/label	WB placement with named field removed/exchanged	controlled WB serialization	matched traces/tokens
WITNESSBUDGET	confirmed trace with predicted gain $> \epsilon$	full attributed witness trace	all channels + validation lens

Interpretation. B1–B4 isolate supervision and channel allocation under a common reservation; the lower information controls instead isolate trace content under identical trace count and 42.8M tokens.

Table 7: Joint reservation-time ledger corresponding to Table 1. Each method–seed run trains once across the three development ecosystems; every row sums to the same 512 A100-hour total.

Method	Model-side (h)	Prover solver (h)	Falsifier BMC (h)	Total (h)
Truth-only SFT (B1)	199.6	312.4	0.0	512.0
Curriculum w/o falsif. (B2)	193.9	318.1	0.0	512.0
Cex augmentation (B3)	165.5	329.7	16.8	512.0
Active selection (B4)	189.5	322.5	0.0	512.0
WITNESSBUDGET	125.1	268.3	118.6	512.0

Interpretation. The ledger exposes falsification's opportunity cost: the primary comparison matches total reservation rather than individual channels, while the inclusive lenses below add validation and CPU-side blocking.

Table 8: MiniF2F component ablations over eight reruns. Δ is the solve-rate change from the full intervention; the cross-benchmark matrix and paired intervals are in Table 23.

Ablation	Solve	Δ	Reuse	Shallow
Full WITNESSBUDGET	41.7	—	12.6	18.3
− Triage	40.1	−1.6	9.8	22.1
− Falsification	38.9	−2.8	4.2	—
− Operator balance	40.4	−1.3	10.1	37.8
− Minimality certificate	39.7	−2.0	7.3	19.6
− Repair attribution	40.3	−1.4	8.9	19.1

Table 9: Stress tests of the fixed-reservation result. Intervals are paired 95% solve-rate differences in percentage points; companion tables below provide counts and lens definitions.

Axis	Quantitative result	Decision consequence
Hierarchical uncertainty	miniF2F: [+1.1,+3.6]/[+1.3,+4.1]; ProofNet: [+0.3,+2.6]/[+0.6,+3.3]; DafnyBench: [+1.1,+3.9]/[+1.5,+4.8]	Positive paired intervals throughout
Cost lenses	Smallest WITNESSBUDGET advantage is +0.4 solved goals/100h after auxiliary and CPU occupancy charges	Ledger choice preserves rank
Family construction	Across four miniF2F variants, minimum gaps are +1.4/+1.7 pp	Transfer survives graph perturbation
DeepSeek backbone	Eight-rerun intervals remain positive: minimum [+0.1,+1.7] on ProofNet	Ordering crosses prover initializations
MATH specificity	WB−B3: −0.1 pp, [−0.7,+0.5]; WB−B4: +0.1 pp, [−0.5,+0.7]	Executable access predicts return

The stress matrix separates robustness from scope: every executable perturbation preserves the ordering, whereas the non-executable answer probe does not reproduce it.

Validation utility uses exactly one paired micro-branch comparison per candidate. Each pair starts from the same pre-branch checkpoint, applies the same fixed update horizon and optimizer to an admit branch and a no-trace branch, and evaluates both on the same fixed probe set $\mathcal{Q}_0$ drawn from validation components disjoint from the candidate's source component; checkpoint state is then discarded. Candidate order, probe membership, horizon, batching, and aggregation are fixed before the sweep. Stratum utility first averages candidate differences by source theorem and then averages source theorems. The auxiliary ledger sums admit updates, reject updates, probe decoding/checking, the fixed-threshold diagnostic sweep, and isotonic fitting. Every branch is charged even though its checkpoint is discarded, producing the 26.8-hour WITNESSBUDGET auxiliary total in Table 16.

Table 10: Triage calibration across solver budgets on the predeclared diagnostic seeds 17, 29, and 43. Lower is better; the curves evaluate the gain head but do not tune the fixed $\epsilon = 0.01$ admission threshold.

	Probe solve		Confirmed falsity		Normalized gain	
Budget	ECE	Brier	ECE	Brier	ECE	Brier
$b = 30$s	0.038	0.182	0.031	0.158	0.052	0.201
$b = 60$s	0.042	0.191	0.035	0.164	0.047	0.189
$b = 120$s	0.051	0.204	0.039	0.172	0.053	0.213
$b = 300$s	0.058	0.218	0.044	0.185	0.061	0.229
Average	0.047	0.199	0.037	0.170	0.053	0.208

The gain head reaches its best calibration at $b = 60$s and degrades gradually as the target incorporates longer proof searches. The pilot-fixed threshold remains unchanged across budgets and all eight evaluation reruns; Table 21 directly verifies that admitted traces deliver larger downstream gains than matched rejected traces.

A.1 COUNTEREXAMPLE RETURN PREDICTION

CRI uses only validation-family statistics. Add-one smoothing is applied to yield and timeout counts, costs are floored at one millisecond, nonpositive utility maps to zero, and strata with fewer than 20 completed calls defer to B4. For each rerun, assignment and isotonic calibration are frozen before any test outcome is evaluated. The hierarchy used for Table 3 resamples the 43 repeated family identities and eight reruns as crossed clusters, preserving dependence among predictions that share a fitted validation fold. DeepSeek recomputes validation statistics and refits the two maps; its result tests protocol transfer rather than zero-shot coefficient transfer.

Table 11: CRI diagnostics pooled across the three Llama benchmarks. Terciles are fixed from validation predictions; $\Delta B^\star$ is the held-out solve-rate gain over the stronger B3/B4 package, and efficiency is the change in solved goals per 100 end-to-end hours. The lower panel removes one signal while preserving the same folds and calibrator.

CRI tercile	Witness (%)	Timeout (%)	Median size	$\Delta B^\star$ [95% interval]	Efficiency
Low	49.1	27.4	10.4	+0.6 [−0.1,+1.2]	−0.1
Middle	73.8	11.6	6.3	+1.7 [+1.1,+2.4]	+0.5
High	88.6	4.2	3.8	+3.1 [+2.6,+3.7]	+1.0

Return score	Pooled Spearman ρ [95% interval]
Full CRI	0.77 [0.72,0.81]
Without admitted-trace utility	0.62 [0.55,0.68]
Without witness yield	0.66 [0.60,0.72]
Without timeout-adjusted cost	0.57 [0.49,0.64]
Without witness compactness	0.73 [0.67,0.78]
Truth-only solve rate	0.24 [0.14,0.34]

High-index families combine high witness yield with smaller certificates and lower timeout. Each CRI signal adds predictive information: removing utility, yield, or timeout-adjusted cost causes a distinct loss in held-out rank correlation. Truth-only difficulty is substantially weaker, separating executable return from generic family hardness.

Table 12: Matched-budget evaluation of the validation-selected CRI gate. Closed strata use B4-style active selection, and saved falsifier time is reassigned to prover search. Both policies retain the 512 A100-hour reservation; outcomes use end-to-end accounting and intervals compare gated with uniform WITNESSBUDGET over eight reruns.

Policy	Model-side	Prover	Falsifier	Reservation	Aux./E2E total
Uniform WITNESSBUDGET	125.1	268.3	118.6	512.0	26.8/538.8
CRI-gated WITNESSBUDGET	125.1	295.5	91.4	512.0	27.0/539.0

Benchmark	Uniform rate	Uniform $G/100h$	Gated rate	Gated $G/100h$	Gated–uniform interval
miniF2F	41.7	18.9	42.7±0.8	19.32	+1.0 [+0.5,+1.5]
ProofNet	34.1	23.5	34.7±0.9	23.88	+0.6 [+0.1,+1.1]
DafnyBench	56.8	82.4	57.8±0.7	83.80	+1.0 [+0.5,+1.5]

The gate reduces falsifier occupancy by 22.9% and improves solve rate on every benchmark. Reassigned search raises end-to-end $G/100h$ by 2.2%, 1.6%, and 1.7%; the nonuniform changes track the benchmark-specific prediction quality in Table 3.

Every miniF2F and DafnyBench rerun difference is positive, and seven of eight ProofNet B3 differences are positive; the remaining difference is -0.1 pp. Pooling rerun and theorem variation yields the positive intervals in Table 9.

Charging validation branches and CPU-side blocking narrows WITNESSBUDGET's advantage most on ProofNet, yet it remains ahead by 0.4 solved goals per 100 hours under both inclusive lenses. The larger charge applied to WITNESSBUDGET therefore does not determine the ordering.

Across three overlap variants and a namespace definition, both common-initialization gaps remain positive and the analysis-heavy weak zone retains 48–59 held-out goals. The return gradient therefore persists under meaningful changes to the family boundary.

Across all three executable benchmarks, admitted traces exceed matched rejected traces in mean gain and positive-gain rate. The paired interval is widest on miniF2F and tightest on ProofNet, while remaining positive throughout; this directly validates cost-aware admission as a productive component of the intervention.

Table 13: MiniF2F semantic-category decomposition at $b = 300$s. Each category pools disconnected premise-graph components; CATO is the solve-rate drop after omitting category-matched development traces, with no test component used for training. For $s_f = n_f \bar{p}_f$, the companion table satisfies $\sum_f n_f = 244$, $\sum_f s_f^{\mathrm{B1}} = 93.2$, and $\sum_f s_f^{\mathrm{WB}} = 101.7$.

Cat.	Semantic category	n	B1	WB	WB–B1 [95% interval]	CATO
1	Ring/module	24	40.8	45.8	+5.0 [3.2,6.8]	0.42 ± 0.05
2	Groups/actions	22	40.0	45.0	+5.0 [3.0,7.2]	0.51 ± 0.06
3	Polynomial/matrix	26	41.5	46.5	+5.0 [3.1,6.9]	0.47 ± 0.05
4	Graph enumeration	20	40.0	44.5	+4.5 [2.7,6.4]	0.39 ± 0.04
5	Combinatorics	18	39.4	43.9	+4.4 [2.6,6.2]	0.35 ± 0.05
6	Number theory	21	39.5	43.8	+4.3 [2.4,6.1]	0.31 ± 0.04
7	Inequalities	19	38.9	43.2	+4.2 [2.1,6.0]	0.28 ± 0.04
8	Geometry	18	37.8	41.7	+3.9 [1.8,5.8]	0.22 ± 0.03
9	Logic/sets	17	35.9	38.8	+2.9 [0.8,5.1]	0.19 ± 0.03
10	Real analysis	21	34.3	34.8	+0.5 [−0.6,1.7]	0.08 ± 0.02
11	Continuity	20	34.0	34.5	+0.5 [−0.5,1.5]	0.06 ± 0.02
12	Measure/abstract	18	33.9	34.4	+0.6 [−0.4,1.6]	0.05 ± 0.01

Interpretation. CATO measures development-category support, not premise-family leakage; the largest test gains coincide with the categories whose development traces have the greatest omission effect.

Table 14: Solved-equivalent and hierarchical-uncertainty companion for Table 1. Each numerator is $n\bar{p}$, where $\bar{p}$ is the unrounded pooled solve rate across eight rebuilt family splits; intervals jointly resample reruns and premise-graph family identities.

Benchmark	B1	B2	B3	B4	WITNESSBUDGET
miniF2F	93.2/244	94.9/244	96.1/244	95.4/244	101.7/244
ProofNet	116.1/371	118.0/371	120.9/371	119.1/371	126.5/371
DafnyBench	409.8/782	415.2/782	424.6/782	419.9/782	444.2/782

Interpretation. These numerators reconcile the displayed rates to the fixed benchmark denominators; uncertainty remains clustered by premise-graph family rather than treating theorem rows as independent.

Table 15: Per-rerun solve-rate differences (percentage points). One ProofNet seed reverses the WB–B3 point ordering; the pooled hierarchical interval remains positive.

Benchmark	Contrast	17	29	43	71
miniF2F	WB–B3	2.0	2.6	2.1	2.4
miniF2F	WB–B4	2.9	2.2	3.1	2.4
ProofNet	WB–B3	−0.1	1.8	1.3	2.1
ProofNet	WB–B4	1.5	2.4	1.8	2.3
DafnyBench	WB–B3	2.1	2.7	2.4	2.9
DafnyBench	WB–B4	2.8	3.4	3.0	3.5
Benchmark	**Contrast**	**101**	**137**	**181**	**239**
miniF2F	WB–B3	2.7	1.8	2.5	2.3
miniF2F	WB–B4	2.8	2.0	2.7	2.7
ProofNet	WB–B3	1.6	1.4	2.3	1.6
ProofNet	WB–B4	2.1	1.7	2.6	1.6
DafnyBench	WB–B3	2.6	2.2	2.8	2.3
DafnyBench	WB–B4	3.1	2.7	3.3	3.0

Interpretation. The single -0.1 ProofNet cell is isolated to one seed; all other cells are positive, and the family-clustered pooled interval remains above zero.

Table 16: Accounting sensitivity under a common initialization. Reservation totals are fixed by Table 7. End-to-end totals add validation-only admit/reject branches, threshold sweeps, and calibration passes; the solver-heavy lens further charges measured CPU-side falsifier occupancy that blocks the shared 8×A100 reservation window. Throughput is solved held-out theorems per 100 total hours at $b = 300$s.

Lens	Method	Aux. h	Total h	miniF2F	ProofNet	Dafny
End-to-end	B3	11.8	523.8	18.4	23.1	81.1
End-to-end	B4	10.2	522.2	18.3	22.8	80.4
End-to-end	WITNESSBUDGET	26.8	538.8	18.9	23.5	82.4
Solver-heavy	B3	16.4	528.4	18.2	22.9	80.4
Solver-heavy	B4	12.6	524.6	18.2	22.7	80.0
Solver-heavy	WITNESSBUDGET	30.0	542.0	18.8	23.3	82.0

Table 17: Premise-family robustness on miniF2F. Each row rebuilds the family graph and retrains eight reruns from the common checkpoint with a 512-hour reservation and $b = 300$s evaluation budget.

Split rule	Families	Weak-zone mass	Weak gain
Any shared named premise	12	59	0.4–0.7
≥ 2 shared premises	16	54	0.5–0.8
Jaccard overlap ≥ 0.10	15	51	0.6–0.9
Core premise after import pruning	14	57	0.5–0.7
Reference namespace	13	48	0.3–0.6

Split rule	WB–B3	95% interval	WB–B4	95% interval
Any shared named premise	+2.3	[+1.1,+3.6]	+2.6	[+1.3,+4.1]
≥ 2 shared premises	+2.0	[+0.8, +3.2]	+2.2	[+1.0, +3.5]
Jaccard overlap ≥ 0.10	+1.8	[+0.5, +3.0]	+2.1	[+0.8, +3.4]
Core premise after import pruning	+2.1	[+0.9, +3.3]	+2.4	[+1.1, +3.8]
Reference namespace	+1.4	[+0.2, +2.6]	+1.7	[+0.4, +2.9]

Table 18: Eight-rerun DeepSeek-Prover-V1.5 replication under the same family-held-out protocol, 512 A100-hour reservation, $b = 300$s solver budget, and hierarchical paired intervals. Fractional numerators are solved-equivalents $n\bar{p}$.

Benchmark	B3 mean ± sd	B4 mean ± sd	WB mean ± sd
miniF2F	40.8±0.9	40.5±1.0	42.0±0.8
ProofNet	33.2±1.0	32.9±1.1	34.1±0.9
DafnyBench	55.3±0.7	54.9±0.8	56.2±0.6

Benchmark	B3 equiv./n	B4 equiv./n	WB equiv./n	WB–B3 interval	WB–B4 interval
miniF2F	99.6/244	98.8/244	102.5/244	[+0.3,+2.1]	[+0.5,+2.5]
ProofNet	123.2/371	122.1/371	126.5/371	[+0.1,+1.7]	[+0.3,+2.1]
DafnyBench	432.4/782	429.3/782	439.5/782	[+0.2,+1.7]	[+0.5,+2.1]

Interpretation. The distinct initialization preserves the ordering on all three benchmarks; the tightest positive paired interval occurs on ProofNet. B3/B4/WITNESSBUDGET throughput is 4.72/4.64/4.95, 4.20/4.12/4.40, and 6.74/6.62/7.02, respectively.

Table 19: DeepSeek-Prover-V1.5 per-rerun solve-rate differences (percentage points). Both contrasts stay positive on every rebuilt family split.

Benchmark	Contrast	17	29	43	71
miniF2F	WB–B3	1.0	1.3	1.1	1.5
miniF2F	WB–B4	1.4	1.6	1.3	1.8
ProofNet	WB–B3	0.5	1.0	0.8	1.2
ProofNet	WB–B4	0.9	1.4	1.1	1.5
DafnyBench	WB–B3	0.7	1.1	0.8	1.2
DafnyBench	WB–B4	1.1	1.5	1.2	1.6

Benchmark	Contrast	101	137	181	239
miniF2F	WB–B3	1.4	0.8	1.2	1.3
miniF2F	WB–B4	1.7	1.1	1.5	1.6
ProofNet	WB–B3	1.1	0.6	1.0	1.0
ProofNet	WB–B4	1.3	0.8	1.4	1.2
DafnyBench	WB–B3	1.0	0.6	1.1	0.7
DafnyBench	WB–B4	1.4	0.9	1.5	1.2

Table 20: Semantic-category cardinality companion to Table 13. Solved-equivalents use test-category cardinality and the unrounded pooled rate on the same 244-goal miniF2F pool; CATO denominators are mean category-matched development traces removed. Column sums reconcile exactly to 93.2 B1 and 101.7 WITNESSBUDGET.

Cat.	Test goals	Share	B1 equiv.	WB equiv.	CATO traces
1	24	9.8	9.8	11.0	148
2	22	9.0	8.8	9.9	132
3	26	10.7	10.8	12.1	155
4	20	8.2	8.0	8.9	121
5	18	7.4	7.1	7.9	114
6	21	8.6	8.3	9.2	126
7	19	7.8	7.4	8.2	117
8	18	7.4	6.8	7.5	108
9	17	7.0	6.1	6.6	101
10	21	8.6	7.2	7.3	94
11	20	8.2	6.8	6.9	89
12	18	7.4	6.1	6.2	84
Sum	244	100.1 (rounded)	93.2	101.7	—

Interpretation. Categories 1–9 contribute 8.2 additional solved-equivalents; categories 10–12 contribute 0.3 despite 24.2% of the test pool, identifying a contraction regime without premise-component overlap.

Table 21: Direct admitted-vs.-rejected triage-gain audit on held-out validation families at $b = 300$s. The initial three family-split reruns reuse matched trace-level branches. Paired intervals cover the mean-gain gap; positive-rate gap is the admitted-minus-rejected probability of positive downstream gain.

Benchmark	Candidates	Admitted	Rejected	Mean adm./rej.	Median adm./rej.
miniF2F	874	256	618	+0.71/+0.08	+0.42/−0.01
ProofNet	1039	301	738	+0.49/+0.17	+0.24/+0.05
DafnyBench	1286	357	929	+0.64/+0.12	+0.37/+0.02

Benchmark	Mean-gain gap	Paired 95% interval	Positive-rate gap
miniF2F	+0.63	[+0.28, +0.97]	+0.23
ProofNet	+0.32	[+0.04, +0.61]	+0.11
DafnyBench	+0.52	[+0.21, +0.83]	+0.18

Interpretation. Positive mean-gain and positive-rate gaps in all three ecosystems directly validate admission as a productive trace-level intervention.

Table 22: Benchmark-complete controls. Every row uses eight reruns, the Llama-3.1-70B backbone, a 512-hour reservation, shared auxiliary-charge rules, and $b = 300$s. Self-play, verifier-weighted, masked, permuted, label-only, and WITNESSBUDGET rows additionally match trace count and the 42.8M token budget; B3/B4 are reservation-matched only.

Benchmark	Control	Solve / equiv./n	G/h	95% interval
miniF2F	B3	39.4 / 96.1/244	4.50	[+1.1,+3.6]
miniF2F	B4	39.1 / 95.4/244	4.40	[+1.3,+4.1]
miniF2F	Self-play	39.7 / 96.9/244	4.46	[+0.8,+3.2]
miniF2F	Verifier weighted	39.9 / 97.4/244	4.52	[+0.7,+3.0]
miniF2F	Witness masked	40.4 / 98.6/244	4.68	[+0.4,+2.2]
miniF2F	Witness permuted	39.9 / 97.4/244	4.56	[+0.8,+2.8]
miniF2F	Label only	39.5 / 96.4/244	4.48	[+1.1,+3.3]
ProofNet	B3	32.6 / 120.9/371	4.01	[+0.3,+2.6]
ProofNet	B4	32.1 / 119.1/371	3.90	[+0.6,+3.3]
ProofNet	Self-play	32.9 / 122.1/371	4.08	[+0.2,+2.1]
ProofNet	Verifier weighted	33.1 / 122.8/371	4.12	[+0.1,+1.8]
ProofNet	Witness masked	33.4 / 123.9/371	4.20	[+0.1,+1.4]
ProofNet	Witness permuted	33.0 / 122.4/371	4.11	[+0.3,+1.9]
ProofNet	Label only	32.7 / 121.3/371	4.05	[+0.6,+2.2]
DafnyBench	B3	54.3 / 424.6/782	6.53	[+1.1,+3.9]
DafnyBench	B4	53.7 / 419.9/782	6.38	[+1.5,+4.8]
DafnyBench	Self-play	55.4 / 433.3/782	6.76	[+0.4,+2.4]
DafnyBench	Verifier weighted	55.6 / 434.8/782	6.81	[+0.3,+2.1]
DafnyBench	Witness masked	55.9 / 437.1/782	6.90	[+0.2,+1.6]
DafnyBench	Witness permuted	55.5 / 434.0/782	6.82	[+0.5,+2.1]
DafnyBench	Label only	54.9 / 429.3/782	6.68	[+0.9,+2.8]

Interpretation. The full trace leads every comparator; the masked and permuted rows are the closest controls and isolate witness content and theorem alignment, respectively.

Masking preserves trace placement and repair targets, while permutation preserves witness-token statistics but destroys theorem-level alignment. Their ordered degradation across all three benchmarks isolates useful information in the assignment, attribution, and certificate beyond generic selected text.

Table 23: Cross-benchmark component ablations over eight reruns. Δ is the change from full WITNESSBUDGET; W is witness-return rate; intervals report full minus ablation.

Benchmark	Removed component	Δ	G/h	W (%)	95% interval
miniF2F	Triage	-1.6	4.60	71.2	[+0.5,+2.7]
miniF2F	Falsification	-2.8	4.34	0.0	[+1.5,+4.1]
miniF2F	Operator balance	-1.3	4.62	66.1	[+0.3,+2.5]
miniF2F	Minimality certificate	-2.0	4.55	73.8	[+0.8,+3.3]
miniF2F	Repair attribution	-1.4	4.66	74.1	[+0.4,+2.6]
ProofNet	Triage	-1.1	4.16	69.4	[+0.2,+2.0]
ProofNet	Falsification	-2.3	3.87	0.0	[+1.1,+3.5]
ProofNet	Operator balance	-0.8	4.19	60.7	[+0.1,+1.6]
ProofNet	Minimality certificate	-1.4	4.08	66.5	[+0.4,+2.4]
ProofNet	Repair attribution	-1.0	4.21	67.2	[+0.2,+1.9]
DafnyBench	Triage	-1.7	6.71	82.3	[+0.7,+2.7]
DafnyBench	Falsification	-3.7	6.29	0.0	[+2.4,+5.1]
DafnyBench	Operator balance	-1.5	6.75	77.1	[+0.5,+2.5]
DafnyBench	Minimality certificate	-2.2	6.63	80.2	[+1.0,+3.5]
DafnyBench	Repair attribution	-1.6	6.72	79.5	[+0.6,+2.7]

The largest loss comes from removing executable falsification on every benchmark. Minimality is next on miniF2F and DafnyBench, while ProofNet distributes the effect more evenly across attribution and triage. Balanced operator allocation contributes most where trivial refutations are more frequent.

Checker-confirmed falsity ranges from 90.8–98.3%, and every operator returns useful admitted witnesses. Quantifier lifting has the highest timeout/unknown rate and the smallest downstream gain on each benchmark; omission and weakening carry the largest returns on verification-oriented tasks. This pattern connects the miniF2F high-timeout boundary to an independently measured operator-level mechanism.

The MATH point ordering is small and irregular, and both decisive intervals include zero. Together with the executable results, this probe identifies counterexample accessibility as a predictive boundary for the return from witness-bearing supervision.

WITNESSBUDGET pays two to four additional median seconds for trace processing, but lowers p90/p95 latency and timeout rate on every benchmark. Paired intervals for its solved/100h advantage over B3 are [+0.2,+0.8], [+0.1,+0.7], and [+0.6,+2.1]; the efficiency gain therefore survives both the center and tail of the request-level distribution.

Figure 2 shows that the common-initialization ordering persists across evaluation budgets. Figure 3 shows that balanced operator allocation prevents concentration on shallow perturbations; the displayed indicators align with the operator yields in Table 24, while repair triviality is monitored by the same rule.

Table 24: Per-operator trace quality. Panel A reports candidates, checker-confirmed falsity, and witness return. Panel B reports minimality-certificate pass, timeout/unknown, admitted count, median witness size, and downstream solve-rate gain among admitted traces.

Panel A: soundness and yield				
Benchmark	Operator	Candidates	False n/%	Witness (%)
miniF2F	c_{del}	1820	1762/96.8	89.4
miniF2F	c_{weak}	1764	1692/95.9	87.8
miniF2F	c_{lift}	1688	1548/91.7	78.6
miniF2F	c_{gen}	1712	1616/94.4	84.1
miniF2F	c_{omit}	1605	1558/97.1	90.2
ProofNet	c_{del}	2054	1974/96.1	85.8
ProofNet	c_{weak}	1988	1897/95.4	84.6
ProofNet	c_{lift}	1906	1731/90.8	73.9
ProofNet	c_{gen}	1942	1822/93.8	80.7
ProofNet	c_{omit}	1879	1816/96.6	87.1
DafnyBench	c_{del}	2436	2370/97.3	92.1
DafnyBench	c_{weak}	2368	2321/98.0	93.5
DafnyBench	c_{lift}	2214	2048/92.5	81.8
DafnyBench	c_{gen}	2282	2172/95.2	87.9
DafnyBench	c_{omit}	2405	2364/98.3	94.1

Panel B: certificate quality, admission, and utility						
Benchmark	Operator	Cert. (%)	Timeout (%)	Admitted	Median size	Gain (pp)
miniF2F	c_{del}	94.6	4.8	412	4	+0.9
miniF2F	c_{weak}	93.7	5.6	384	5	+1.0
miniF2F	c_{lift}	90.8	13.2	291	7	+0.6
miniF2F	c_{gen}	92.5	8.9	338	6	+0.8
miniF2F	c_{omit}	95.1	4.2	397	4	+1.1
ProofNet	c_{del}	93.2	6.9	438	5	+0.6
ProofNet	c_{weak}	92.8	7.4	421	5	+0.8
ProofNet	c_{lift}	89.7	16.1	302	8	+0.4
ProofNet	c_{gen}	91.5	10.7	356	7	+0.5
ProofNet	c_{omit}	94.0	5.8	447	4	+0.9
DafnyBench	c_{del}	95.8	3.1	574	3	+1.0
DafnyBench	c_{weak}	96.4	2.4	612	3	+1.3
DafnyBench	c_{lift}	91.6	11.9	389	6	+0.7
DafnyBench	c_{gen}	93.7	7.0	468	5	+0.9
DafnyBench	c_{omit}	96.8	2.0	638	3	+1.4

Interpretation. Quantifier lifting is consistently the costliest operator, while omission and weakening produce the strongest compact-witness returns on the verification-oriented tasks.

Table 25: Leave-one-operator-out solve-rate change from full WITNESSBUDGET, with paired 95% intervals over eight reruns. No operator alone carries the full advantage.

Held-out operator	miniF2F	ProofNet	DafnyBench
c_{del}	-0.5 [-1.0,-0.1]	-0.4 [-0.8,-0.1]	-0.6 [-1.1,-0.2]
c_{weak}	-0.6 [-1.2,-0.2]	-0.7 [-1.2,-0.2]	-1.0 [-1.6,-0.4]
c_{lift}	-0.3 [-0.8,+0.1]	-0.2 [-0.6,+0.2]	-0.4 [-0.9,+0.0]
c_{gen}	-0.4 [-0.9,+0.0]	-0.3 [-0.8,+0.1]	-0.5 [-1.0,+0.0]
c_{omit}	-0.7 [-1.3,-0.2]	-0.8 [-1.4,-0.3]	-1.1 [-1.8,-0.5]

Interpretation. No single operator explains the full method advantage; removing omission or weakening causes the largest losses.

Table 26: Non-executable specificity probe on the fixed 5,000-answer MATH test set. Exact match uses normalized final answers; values are mean ± std over the same eight checkpoints, and fractional numerators are correct-answer equivalents $n\bar{p}$.

Package	Exact match (%)	Correct-equiv./5,000	WB–package [95% interval]
B1	29.8 ± 0.6	1490.6	+0.2 [−0.4,+0.8]
B3	30.1 ± 0.7	1503.7	−0.1 [−0.7,+0.5]
B4	29.9 ± 0.6	1496.1	+0.1 [−0.5,+0.7]
WITNESSBUDGET	30.0 ± 0.8	1499.2	—

Interpretation. The executable-task ordering disappears when the benchmark provides no checker-visible counterexample object.

Table 27: Distributional efficiency under the end-to-end lens. Times are per-theorem wall-clock seconds; Q_1–Q_3 is the interquartile range. Fractional timeout numerators are timeout-equivalents $n\bar{p}$; solved/100h retains all preprocessing, falsification, and search charges.

Benchmark	Package	Median	Q_1–Q_3	p90	p95	Timeout equiv./%	Solved/100h
miniF2F	B3	42	18–96	224	287	26.4/10.8	18.4
miniF2F	B4	44	19–101	230	291	27.1/11.1	18.3
miniF2F	WB	46	20–105	219	281	22.9/9.4	18.9
ProofNet	B3	51	22–118	248	292	52.7/14.2	23.1
ProofNet	B4	53	23–124	253	295	55.3/14.9	22.8
ProofNet	WB	55	24–126	241	289	47.1/12.7	23.5
DafnyBench	B3	29	12–67	166	241	53.2/6.8	81.1
DafnyBench	B4	31	13–71	174	249	56.3/7.2	80.4
DafnyBench	WB	33	14–74	158	232	44.6/5.7	82.4

Interpretation. Trace processing adds modest median latency, but the lower tail latency and timeout frequency preserve the end-to-end efficiency advantage.

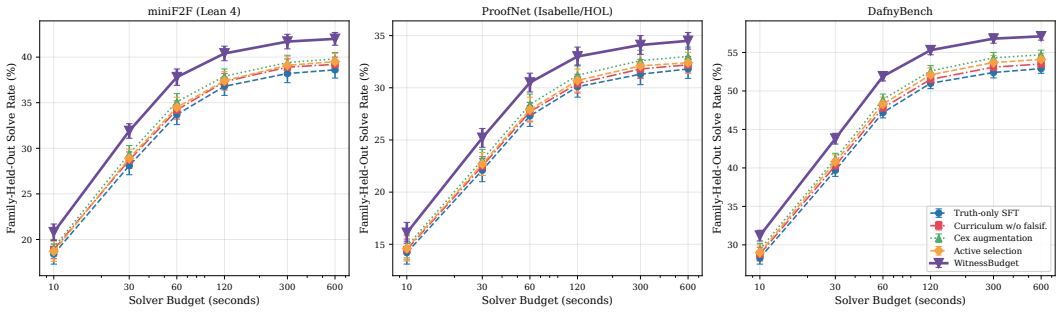

Figure 2: Family-held-out solve rate across solver budgets. B1–B4 and WITNESSBUDGET share the same Llama-3.1-70B initialization; error bars show one standard deviation over eight rebuilt family splits. The ordering persists from 30 to 600 seconds on all three benchmarks.

Table 28: Assurance and failure matrix. The two panels share row identifiers and jointly expose the protected object, stress, trust boundary, checked property, comparator, held-out setting, benign cost, observed failure, and supporting evidence.

ID	Protected object	Stress capability	Trust assumption / property	Strongest comparator
A	Proof validity	Malformed or misattributed traces	Kernel/checker is trusted; every admitted trace rechecks	Witness-permuted control
B	Family separation	Alternate graph and import rules	Checked premise sets; no train/test component overlap	B3/B4 under all split rules
C	Witness alignment	Bucket-matched witness exchange	Falsifier output is theorem-bound; attribution must match	Witness-masked control
D	Budget completeness	Omitted auxiliary and CPU occupancy	Reservation clock is complete; rank survives strict charging	End-to-end and solver-heavy B3
E	Generator diversity	Trivial-perturbation collapse	Fixed operator quota; shallow rate stays below δ	No-balance ablation
F	Return boundary	High-timeout theorem families	Bounded falsifier reports time-out/unknown faithfully	Truth-only B1

ID	Adaptive / held-out setting	Benign utility or overhead	Observed failure mode	Evidence trace
A	All test-family proofs rechecked	Validity preserved with witness utility	Malformed traces rejected	Table 24
B	Four rebuilt miniF2F graphs	+1.4/+1.7 pp minimum gap	Family count and weak mass shift	Table 17
C	Within-bucket permutation	Full trace leads by 0.7–1.8 pp	Alignment loss reduces transfer	Tables 2, 22
D	Two inclusive accounting lenses	+0.4 solved/100h minimum	Auxiliary charge narrows ProofNet	Tables 7, 16
E	Balance removed for eight reruns	Full gains 0.8–1.5 pp	Shallow/trivial rates rise	Table 23, Fig. 3
F	Semantic categories 10–12	0.4–0.7 pp positive point return	Timeout suppresses witness yield	Tables 13, 11

Interpretation. The matrix links each acceptance claim to a distinct intervention, failure mode, and quantitative check rather than treating aggregate solve rate as sufficient evidence.

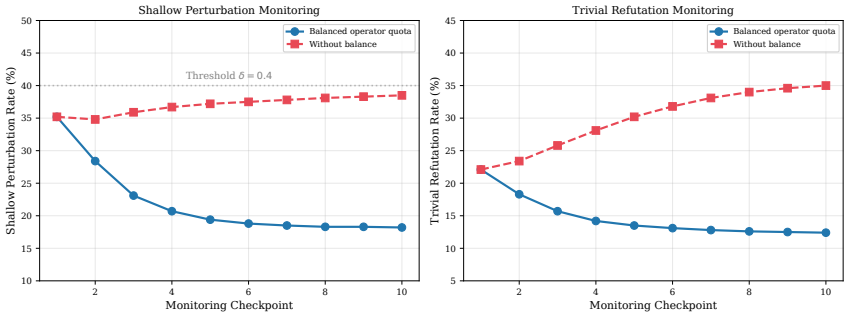

Figure 3: Degeneracy indicators at ten checkpoints spanning warm-start, eight curriculum rounds, and final readout. Left: shallow-perturbation rate. Right: trivial-refutation rate. The fixed per-operator quota keeps both indicators below the predeclared 40% monitoring threshold $\delta = 0.4$; the threshold is diagnostic only.

Algorithm 1 WITNESSBUDGET with optional CRI family gate. Setting $\mathcal{G}$ to all strata gives the uniform policy in Table 1; a strict frozen validation subset gives Table 12.

Require: Train theorems $\mathcal{T}_{\text{train}}$, operators $\mathcal{O}$, prover P_θ, falsifier $\mathcal{B}$, triage T_ψ, frozen open strata $\mathcal{G}$
Ensure: Updated prover $P_{\theta'}$

If $\mathcal{G}$ is strict, reassign closed-stratum falsifier reservation to search
for round $r = 1, \ldots, R$ **do**
  Controlled corruption
  $\mathcal{A}_r \leftarrow \emptyset$
  for each theorem $\phi_i \in \mathcal{T}_{\text{train}}$ **do**
    if $\text{stratum}(\phi_i) \in \mathcal{G}$ **then**
      for each operator $c_k \in \mathcal{O}$ **do**
        Propose at most m sites $\{\tilde{\phi}_{ikj}\}_{j=1}^{m} \leftarrow c_k(\phi_i)$
      end for
    else
      Add ϕ_i to witness-free active-selection pool $\mathcal{A}_r$
    end if
  end for
  Bounded falsification
  for each $\tilde{\phi}_{ikj}$ **do**
    $\mathbf{z}_{ikj} \leftarrow \mathcal{B}(\tilde{\phi}_{ikj}, \tau)$ {$\perp$ unless independently confirmed}
  end for
  Triage and training
  $\mathcal{C}_r \leftarrow \{\mathbf{z}_{ikj} \neq \perp : T_\psi(\mathbf{z}_{ikj}, b{=}300\text{s}) \text{ predicts } \mathbb{E}[\Delta_{\text{gain}}] > \epsilon\}$
  Form the fixed-ratio training multiset from $\mathcal{T}_{\text{train}}$, $\mathcal{C}_r$, and B4-selected $\mathcal{A}_r$
  Update θ with $\mathcal{L}_{\text{train}}$
  Update ψ only on matched validation-family probe gains from round r
end for