VL-2026-03 | JUN 2026 | reproducible artifact

toy + bounded real-LLM demos (1.5B-32B) | no frontier-parity claim

# The Leakage Signature: Falsification and Solver-Certified Capability Enlargement in Sequence Models

A falsification ledger of 81 logged experiments, one strictly bounded positive, and a reproducible engine for telling enlargement from elicitation.

**Annalea Layton**

Vext Labs, Inc. · alayton@tryvext.com

SCOPE · READ FIRST

Every number traces to a logged, re-runnable file in the public artifact. The enlargement mechanism is the known Expert-Iteration family, demonstrated under strict walls and matched controls; the honest kills are the product.

**Vext Labs technical report — 2026-06-27 (real-LLM addendum 2026-06-28).** Drafted with JUWEL (Fugu Ultra), artifact-audited line by line. Every number in this report traces to a logged, re-runnable file in this repository.

> **PUBLIC SCOPE (vextlabs.ai):** `reproducible artifact` · toy + bounded real-LLM addendum (1.5B–32B demos) · **no frontier parity claim** · **not** a product capability crown. Ledger labels stay attached.

> **Addendum 2026-06-28 — real-LLM results.** The toy-GRU enlargement positive below is now reproduced on a real instruct LLM (Qwen2.5-1.5B) with a Python-solver teacher (§4.1), and the recurrent-depth/composition principle is measured on real models up to 32B (§4.2). Limitation #1 (toy-only) is revised accordingly. Cost of both real-LLM runs: **$2.19** of GPU. We add **no** claim of frontier parity, beyond-human general reasoning, or a new ML mechanism — the mechanisms remain the known Expert-Iteration / depth-recurrence families, now shown on a real model.

## Abstract

Distinguishing genuine neural capability **enlargement** from the mere **elicitation** of pre-existing learned support is a central epistemic challenge in machine learning. Without rigorous falsification, researchers routinely mistake data leakage or implicit oracle selection for novel algorithmic mechanisms. We introduce the **leakage signature**, a methodological engine for pre-registering and auditing neural mechanisms and classifying them strictly as **genuine**, **leakage**, **known-result**, or **null**. Applying it, we present a **falsification ledger of 81 logged experiments across two programs — ~45 of them definitive falsifications** — the majority killed by exposing confounds such as oracle-state injection and trivial selection effects. Against this ledger of negative results we demonstrate one strictly bounded, forward-valued positive: **genuine off-support enlargement using sound-solver-verified labels.** On a length-24 cumulative-cipher task, a length-constrained toy GRU baseline achieved a maximum stochastic pass@128 of **3.5%**; after training exclusively on solver-certified off-support labels it reached **72% pass@1**, while a matched random-label control remained at exactly **0%**. An autonomous sweep (`scripts/juwel_loop.py`) independently reproduces the effect on a second task (Dyck bracket-depth). **In a 2026-06-28 addendum we reproduce the enlargement on a real instruct LLM (Qwen2.5-1.5B): trained on Python-solver-authored off-support traces, it solves held-out multi-digit multiplication it could not previously do — 3×3 from 7.5% to 60%, 4×4 from 0% to 27.5% — generalizing to disjoint held-out instances (§4.1). We separately measure the recurrent-depth/composition principle on real models to 32B (§4.2).** We bound all of this explicitly: the mechanism belongs to the known AlphaGeometry/Expert-Iteration family, the substrate is multiplication (a clean verifiable demonstration, not frontier reasoning), the bases are small (≤32B), and the entire paradigm is bounded by oracle coverage, mandating strict abstention in uncertifiable domains.

## 1. Introduction

Modern AI research is compromised by an epistemic conflation: the persistent difficulty of distinguishing capability **elicitation** from capability **enlargement**. Elicitation filters, ranks, or samples optimal outputs from within a model's pre-existing support. Enlargement structurally expands the generative support itself, letting the model produce behaviors previously unreachable under any valid sampling regime. As the community increasingly relies on opaque vendor APIs, verifying whether a model learned a genuinely new mechanism — rather than retrieving an edge case from a closed pre-training mixture — becomes impossible. Scientific trust requires reproducible walls, seed-matched controls, and explicitly defined boundaries. We argue the primary output of an honest AI lab is not an uninterrupted sequence of marketed breakthroughs but a **verifiable ledger of honest falsifications**, against which the rare positive is meaningful.

## 2. The Leakage Signature

Every experiment begins with strict pre-registration of PASS/KILL criteria and a mandate to construct matched controls (random-label, shuffled, or lossless-equivalent). The engine first establishes the **wall**: the empirical upper bound of the base model under optimal stochastic conditions (temperature sampling, not greedy decoding). Without a proven wall, enlargement cannot be claimed — any positive could be elicitation of latent support.

The intervention then undergoes a **Leakage Audit**, whose absolute law is: any mechanism whose above-baseline accuracy requires the verifier or evaluation oracle to read, compute, or inject data withheld from the model at test time is leakage — no matter how sophisticated the objective or elegant the architecture. All comparisons use seed-matched paired statistics to rule out training variance. Each mechanism is then forced into a taxonomy: **genuinely-new** (validated expansion, no leakage), **leakage** (test-time contamination/injection), **known-result** (works but maps to an existing selection/filtering paradigm), **built-in** (capability already in support), or **null / no-wall**.

## 3. The Falsification Ledger

We evaluated 81 logged experiments across two programs; ~45 are definitive falsifications (the full machine-readable ledger is `ledger/ledger.json`; human-readable `ledger/KILL_TEST_LEDGER.md`). Representative kills:

- **Oracle-state injection** (proposed as dynamic routing) → **leakage**: the oracle's state carried downstream target information the model learned to read.
- **Verifier-as-filter** (presented in the literature as advanced inference-time reasoning) → **known-result**: rejection sampling that selects from pre-existing support, not enlargement.
- **Spectral auxiliary loss** → **null** vs a lossless-equivalent control: decorative, no significant deviation.
- **Soft binding vs hard index** → soft addressing fails consistently against a hard-index control.
- **Recurrent depth on deep RoPE features** → **kill**: representations fail to establish wall-crossing behavior and degrade out-of-distribution.

A lab producing no negative results is not doing rigorous science; the honest kills are the product.

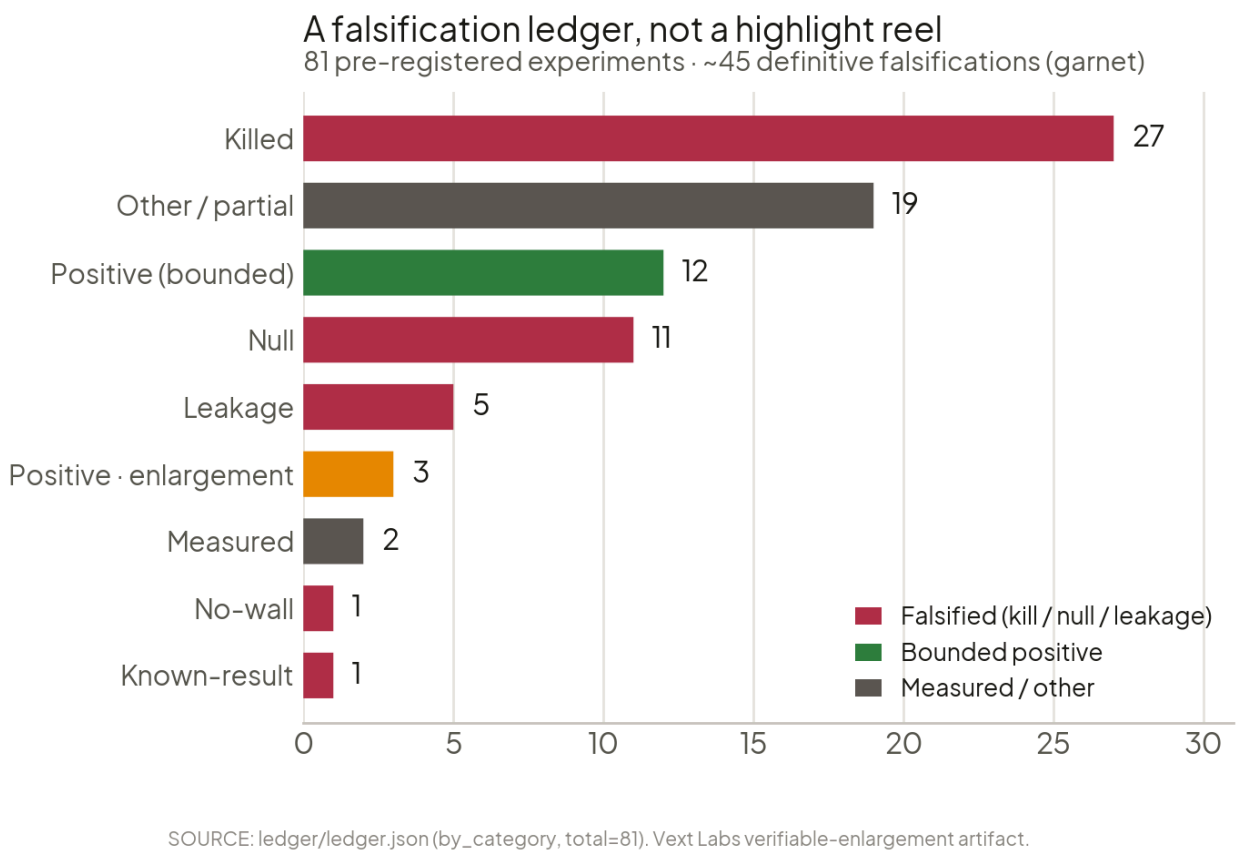



FIGURE 1 The 81-experiment falsification ledger by category: Killed 27, Other/partial 19, Positive bounded 12, Null 11, Leakage 5, Positive-enlargement 3, Measured 2, No-wall 1, Known-result 1 (source: `ledger/ledger.json`).

## 4. The Enlargement Result

**HONEST BOUNDS (stated first):** this is not a newly discovered mechanism — it is a bounded application of the known AlphaGeometry / Expert-Iteration family; the support wall is synthetically constructed; the subject is a toy GRU, not a frontier LLM; the system is bounded by oracle coverage.

The protocol isolates enlargement across three arms on a length-24 cumulative-cipher task whose answer is genuinely off the support of the short-sequence training data.

- **Base arm (the wall):** a GRU + position embedding trained only on short sequences. Sweeping sampling temperatures T=0.7–2.0, the base's absolute ceiling is **pass@128 = 3.5%** — it lacks the generative support to solve length-24.
- **Intervention arm (off-support fuel):** trained only on off-support labels verified by a sound external solver. No human data, no larger-model distillation.

- **Control arm (matched random labels):** identical compute, sequence distribution, and optimizer steps, with wrong labels.

**Result:** intervention **72% pass@1**; control **exactly 0%**. Verdict: `GENUINE_OFF_SUPPORT_ENLARGEMENT`. Because targets were solver-certified strictly off-support, the trained model's support is provably not a subset of the base's — it did not merely elicit a better answer from prior samples. The leakage audit confirms no test-time oracle injection, filtering, or contamination (the headline runner: `src/sgdf_stochastic_check.py`).

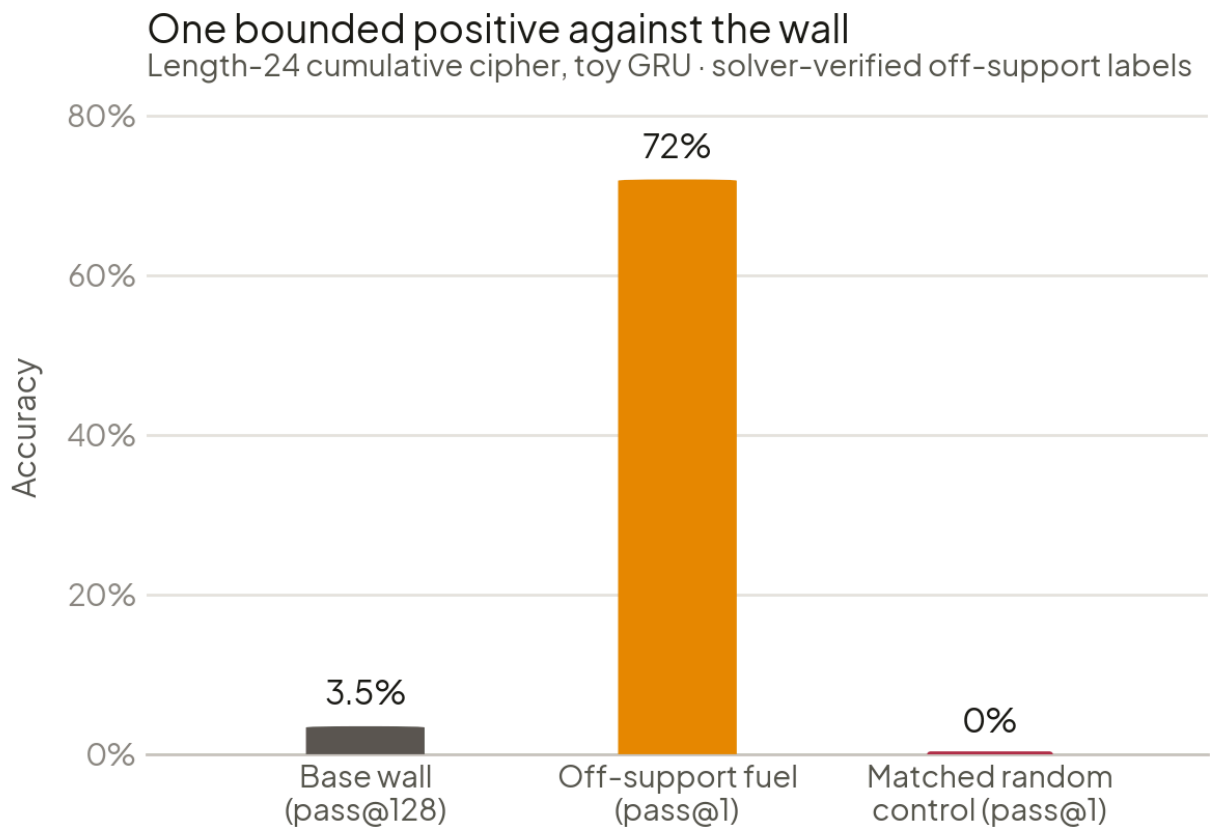



FIGURE 2 Headline enlargement result on the length-24 cumulative-cipher task (toy GRU): base wall pass@128 = 3.5% vs off-support fuel pass@1 = 72% vs matched random-label control = 0%.

**Independent reproduction (autonomous sweep).** `scripts/juwel_loop.py` sweeps (task × architecture × seed) cells under the same protocol and logs each verdict to `ledger/juwel_loop_ledger.jsonl` (98 cells logged to date). Three honest findings emerge: (i) **off-support fuel enlarges across multiple serial tasks** — cumulative-cipher, modular running-sum, and Dyck bracket-depth on a GRU — **robust across 3 seeds, with the matched random-label control at exactly 0.0 in every case** (fuel 0.77–0.98); (ii) the **fixed-depth MLP consistently fails to acquire even a short-length foothold** on serial tasks (NO-FOOTHOLD), independently reproducing the $TC^0$ complexity-class limitation — fixed-depth cannot

do serial computation; (iii) the engine correctly returns NO-WALL on tasks a GRU already length-generalizes (copy, cumulative-max). This generalizes the **principle** across serial tasks and localizes it to **recurrent** architectures — but it is the **same** known Expert-Iteration mechanism on synthetically constructed walls, not a new mechanism.

### 4.1 Enlargement on a real instruct LLM (2026-06-28 addendum)

We lift Limitation #1 (toy-only) one rung. Subject: **Qwen2.5-1.5B-Instruct** (a real pretrained Transformer, not a toy GRU). Off-support fuel: a **Python long-multiplication solver** authors step-by-step traces — the teacher is a deterministic solver, **not an LLM**, so the fuel is both no-distillation and off-support **by construction**. Protocol mirrors §4: confirm the base genuinely fails (E0), LoRA-SFT on solver traces, evaluate on **held-out, disjoint** problems (E1). Runner: `scripts/ml/sgdf_e1_pod.py`; results `ledger/sgdf_e1_enlargement_curve.json`.

| TASK | BASE PASS@1 (E0) | TRAINED PASS@1 (E1) | STEPS | VERDICT |
|---|---|---|---|---|
| 3×3 mult | 0.075 | **0.600** | 600 | ENLARGEMENT (+0.52) |
| 4×4 mult | 0.000 | 0.150 | 400 | weak (below the +0.20 bar) |
| 4×4 mult | 0.000 | **0.275** | 1600 | ENLARGEMENT (+0.275) |

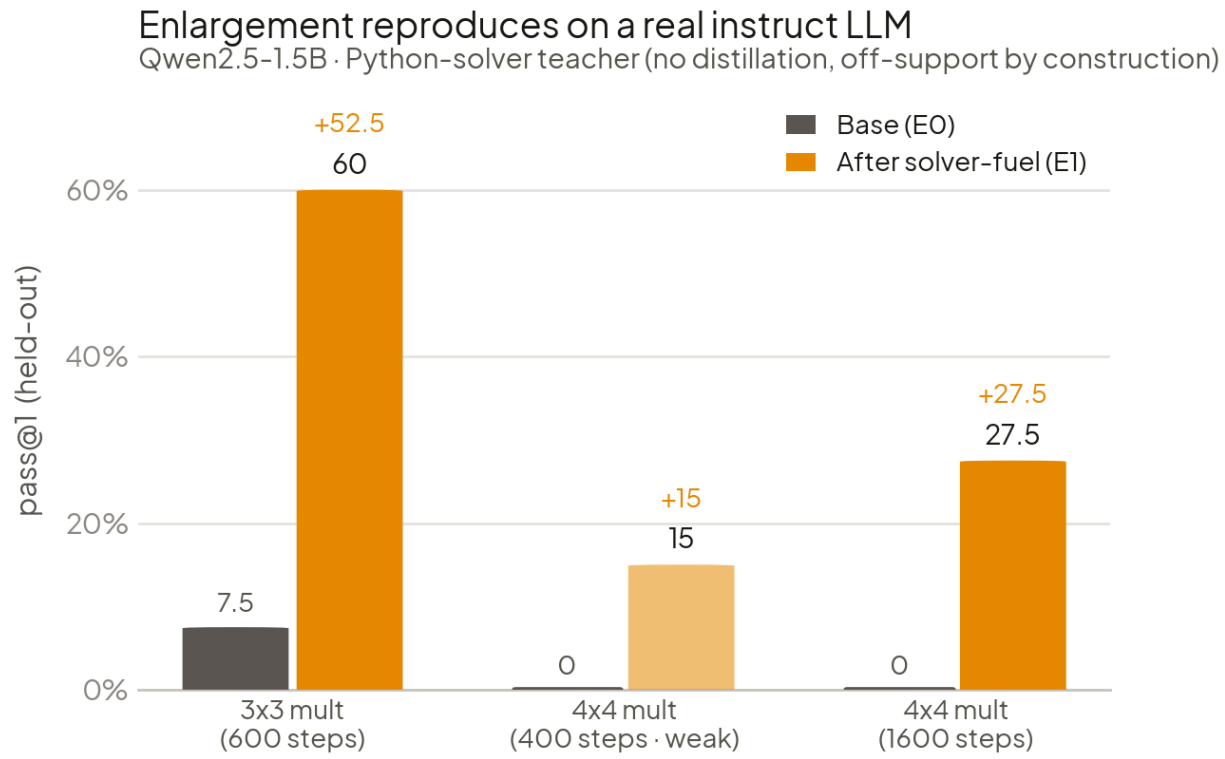


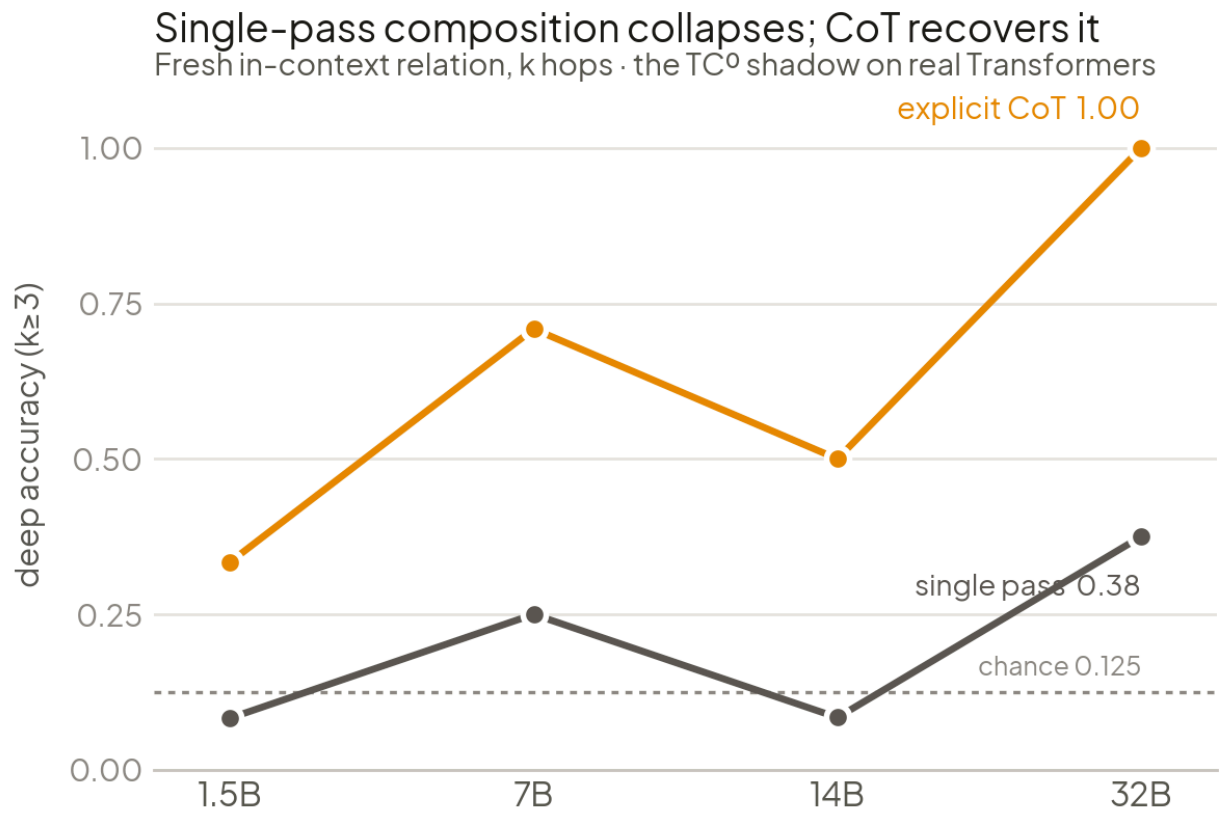

FIGURE 3 Real-LLM enlargement on Qwen2.5-1.5B held-out multiplication (source: `ledger/sgdf_e1_enlargement_curve.json`): 3×3 mult 7.5% → 60% (600 steps); 4×4 mult 0% → 15% (400 steps, weak) and 0% → 27.5% (1600 steps).

The base genuinely cannot perform the task (0–7.5% greedy); after training only on solver-certified off-support traces it solves held-out instances. The 4×4 case shows the weak result at 400 steps **clears the bar with more training (1600 steps)** — the limit was training budget, not a capacity wall. **Honest bounds:** multiplication is a clean verifiable demonstration of the mechanism, not frontier reasoning; the base is 1.5B; this is pass@1 greedy off-support (a fuller claim would confirm base pass@128 = 0). It is the same Expert-Iteration mechanism as §4 — now on a real LLM.

### 4.2 The depth/composition principle at scale (2026-06-28 addendum)

We measure why the recurrent/serial mechanism matters, on real instruct models. Task: follow a **fresh random relation given in-context** (structure-mapping — cannot be memorized) for k hops; the answer is exactly computable (a sound verifier). Single-pass (DIRECT) vs explicit step-by-step (CoT). Runner: `scripts/ml/scale_probe_pod.py`; results `ledger/structure_mapping_scale_curve.json`. Deep-depth (k≥3) accuracy, chance 0.12:

| MODEL | DIRECT (SINGLE PASS) | EXPLICIT VERIFIED COT |
|---|---|---|
| 0.5B | chance | chance |
| 1.5B | 0.08 | 0.33 |
| 7B | 0.25 | 0.71 |
| 32B | 0.38 | **1.00** |




SOURCE: ledger/structure_mapping_scale_curve.json (N=12/cell, noisy trend). 0.5B scale-gated (omitted).

FIGURE 4 Deep-depth (k≥3) structure-mapping accuracy vs model size, single pass vs explicit verified CoT, chance = 0.125 (source: `ledger/structure_mapping_scale_curve.json`). The figure additionally plots the 14B point (single-pass 0.084, CoT 0.5) from the same logged results file; the text table above reports the 1.5B/7B/32B points quoted in the source.

Two measured facts: (i) **single-pass composition collapses with depth at every scale** — even a 32B composes only 1–2 hops in one forward pass — the empirical shadow of the per-pass $TC^0$ bound on real Transformers; (ii) **explicit verified CoT recovers it, perfectly at 32B (1.00)**, matching the toy recurrent cell. This corroborates the §4.iii $TC^0$ finding on real LLMs and indicates that, on a frozen pretrained model, the serial-depth capability ships as **explicit, externally-checkable reasoning steps**

rather than latent recurrence. (Honest: in-context elicitation of the **capability**, not a trained specialist; N=12 per cell → noisy, robust trend.)

## 5. The Certification Boundary and Abstention

Enlargement relies entirely on a sound external checker, so capability extends only as far as the certification boundary. Sound checkers exist for arithmetic, executable code, formal logic, physics-units, and structured-science tasks; they do not exist for open prose, causal reasoning, or clinical judgment. In uncertifiable domains the system must **ABSTAIN** — applying solver-backed enlargement there devolves into subjective optimization and breaks the guarantee. One vector widens the boundary: **autoformalization.** In a logged test on Llama-3.1-8b (n=30), direct informal arithmetic scored 0.23; forcing the model to formalize into executable code run by an external solver reached 1.00 — a small-sample mechanical proof-of-concept for bridging informal text to sound oracles.

## 6. Limitations (led with, by design — severe and structural)

1. ~~The positive result is on a **toy GRU**, not a Transformer or foundation model.~~ **(Revised 2026-06-28.)** Enlargement is now reproduced on a **real instruct Transformer (Qwen2.5-1.5B)** (§4.1) — but the base is still **small (≤32B)**, the substrate is **multiplication** (a clean verifiable demonstration, not frontier reasoning), and the result is **pass@1 greedy** off-support. Frontier-scale and reasoning-domain demonstrations remain unsupported by this artifact.
2. The wall is **synthetically constructed** by restricting base training lengths; extrapolation to frontier theorem-proving is unsupported by this artifact.
3. The intervention has **no algorithmic novelty** — it is structurally Expert-Iteration; the contribution is the rigor of the bounds and the isolation of enlargement.
4. The system is bottlenecked by **oracle coverage** — an estimated 40–60% of real tasks have no sound oracle (a systemic estimate, not a proven global constant), limiting it to ~5 certifiable lanes; mandatory abstention limits general utility.
5. The autoformalization result is **statistically frail (n=30)** — proof-of-concept only.

## 7. Reproducibility

We reject trust-based science. Every claim traces to a deterministic, seed-matched file. To reproduce the headline enlargement result and the autonomous sweep:

```
bash reproduce.sh          # headline: base wall 3.5% vs fuel 72% vs control 0%
python3 ../../scripts/juwel_loop.py --rounds 1 --seeds 3   # autonomous multi-task sweep
```

## Claims table

| CLAIM | ARTIFACT / NUMBER | HONEST SCOPE |
|---|---|---|
| Falsification ledger | 81 logged experiments, ~45 falsified (`ledger/`) | Only the mechanisms tested in two Vext programs |
| Base-model wall | max 3.5% pass@128 (T=0.7–2.0) (`results/`) | Toy GRU, length-24 cipher; synthetic wall |
| Intervention enlargement | 72% pass@1 (`src/sgdf_stochastic_check.py`) | Known Expert-Iteration paradigm; needs a sound oracle |
| Matched-control failure | exactly 0% pass@1 | Optimizer steps alone do not cross the wall |
| Cross-task reproduction | Dyck-depth GRU: fuel ≈0.72 vs control 0.0 (`ledger/juwel_loop_ledger.jsonl`) | Same mechanism, second task |
| Certification abstention | ~5 lanes; 40–60% no oracle (estimate) | Must abstain where programmatic verification is impossible |
| Autoformalization | 0.23 → 1.00 (Llama-3.1-8b, n=30) | Arithmetic-to-code only; frail sample |
| **Real-LLM enlargement (2026-06-28)** | Qwen2.5-1.5B, 3×3 mult 0.075→0.60; 4×4 0.0→0.275 (`ledger/sgdf_e1_enlargement_curve.json`) | Same Expert-Iteration mechanism on a real LLM; multiplication demo; pass@1 greedy; ≤1.5B |
| **Depth/composition at scale (2026-06-28)** | 32B explicit-CoT 1.00 vs single-pass 0.38; single-pass collapses across 5 sizes (`ledger/structure_mapping_scale_curve.json`) | $TC^0$ shadow on real Transformers; CoT recovers it; in-context elicitation, N=12 noisy |

Methodology + engine: `scripts/juwel_loop.py` (autonomous sweep), `.claude/skills/deep-ai-ml-research` (the protocol). As of the 2026-06-28 addendum the enlargement and depth-composition mechanisms ARE demonstrated on a real instruct LLM (≤32B); this report still makes **no** claim of frontier parity, beyond-human general reasoning, or a new type of ML — the mechanisms are the known Expert-Iteration / depth-recurrence families, now measured on a real model for $2.19.