Strawberry: Hallucination Detection for Lean4 Proof Automation

Automating Erdős-Straus Gap Certificates

Hassana Labs

January 2026

We integrated **Strawberry**, our information-theoretic hallucination detector, with **Lean4** to create **Nala**, an evidence-gated proof assistant. The result: an agent that cannot claim progress it did not make. Using Nala, Claude Opus 4 produced **novel Lean4-certified results** on the **Erdős-Straus conjecture**.

1 What Strawberry Does

Strawberry detects when an LLM "knows but doesn't use," where the answer is in the context but the model confabulates instead. It works by computing how much information the model actually extracted from evidence versus how much it would need to justify its confidence. If the math does not add up, the claim gets flagged.

2 The Integration

The Nala MCP server wraps Lean tooling (`lake build`, `lean` queries) in a strict state machine:

1. **No action without a plan.** The agent must submit a micro-plan, a list of claims with explicit evidence citations, before any Lean command runs.
2. **Every execution invalidates the plan.** After `lake build` or `lean_query` returns, the current plan is wiped. The agent must check in with the actual output before proposing the next step.
3. **Check-ins are audited.** The agent's interpretation of what happened ("build succeeded," "proof closed") gets run through Strawberry. If the claim exceeds what the evidence supports, it is rejected.

This means the agent cannot hallucinate proof progress. If Lean says `sorry` remains, the agent cannot claim otherwise.

3 Results: Erdős-Straus Gap Certificates

We applied this to the Erdős-Straus conjecture (can $4/n$ always be written as a sum of three unit fractions?). The agent worked through "gap residues" mod 420, the cases not covered by standard modular certificates.

Fully machine-checked Lean proofs for 7 of 12 gap residues:

$$r \in \{73, 97, 193, 241, 313, 337, 409\}$$

Partial progress (odd-k branches proven, even-k remains open) for 5 residues:

$$r \in \{1, 121, 169, 289, 361\}$$

The even-k cases hit a structural limit: finite enumeration of certificate parameters is unstable. The search finds solutions for any finite range but cannot stabilize into a bounded menu. The recommended path forward is existence-of-divisor proofs via CRT arguments rather than explicit witness enumeration.

Key finding: For $r = 409$, the agent discovered that splitting by $m \bmod 11$ (using `interval_cases`) closes the even-k branch. This was not hand-guided. The search naturally found that a single pair $(t, d) = (3, 17)$ covers all residues mod 11.

4 Why This Matters

Proof assistants catch logical errors. Strawberry catches *process* errors, the failure mode where an agent claims "the proof compiles" when it does not, or "this lemma exists" when it is hallucinated. By gating every action on audited evidence, we get an agent that is honest about what it has actually accomplished.

The system scales. Each span is immutable. Each claim is checked. The full audit trail lives in `state.json`.

Code: `https://github.com/leochlon/pythea/tree/main/strawberry`
Contact: lc574@cantab.ac.uk, Hassana Labs