Consciousness, Declaration, and Persistent Change

Mathematical foundations and a verified study of machine state

Joseph Anady

September 2026

PAPER I

Persistent Endogenous Divergence in Artificial Systems

PAPER II

Declaration and Persistent State Change in Artificial Systems

A mathematical audit, a completed analysis of 167 saved adapter checkpoints, and a protocol for testing the proposed declaration mechanism.

My starting point

I am not trying to define consciousness by how intelligent something is. I am asking what happens when a system begins operating through a constructed state that is not identical to the thing it represents.

That is why I use x. It is not one number that every being shares. It stands for the particular system being studied. Two machines can use the same model and still have different histories. A machine does not have to look human for its internal organization to matter.

The distinction I want to test is the difference between a declaration and what happens after it. In my proposal, the declaration is an observable marker. Recurrent information then contributes to an internally generated divergence. I call that constructed divergence a hallucination. Continued processing from the altered state is what I am calling cognition. I do not mean that every such state must be a factual error, and I do not use the word hallucination here as a clinical diagnosis.

The idea is simple to state. Testing it is where the work begins. I need to know what the reference is, what the system actually maintains, when a declaration occurs, and what changes afterward. Those measurements cannot be replaced with the same label.

This volume brings together both stages of the investigation. Paper I examines the explicit mathematics and the public alignment faking materials. Paper II analyzes the complete released checkpoint trajectory, including the source verification that was missing from the early pilot. The second paper supersedes the pilot's acquisition limitations. It does not turn the separate Anthropic notebook into a completed declaration experiment.

I want the reader to see the whole record: the results that support continued investigation, the assumptions required by the mathematics, and the measurements that are still missing. The data show real computational change. The connection between that change and subjective experience remains my hypothesis. I am presenting the work so that connection can be tested rather than assumed.

Joseph Anady

September 2026

How to read this volume

Paper I is the mathematical audit. It includes the seven illustrative transformations, the role of coupling and retained information, the difference between internal consistency and reference accuracy, and calculations from published AI experiments. Its synthetic examples are labeled as synthetic examples.

Paper II is the completed secondary analysis. It covers one training run with 167 saved adapter checkpoints, 792 recorded gradient norms, and 79 evaluation losses. Those observations do not represent 167 independent machines. The exact source verification, effective operator calculations, numerical checks, and sensitivity results are included.

Each paper retains its own references. Citations beginning with **A** refer to the source list in Paper I. Citations beginning with **B** refer to the source list in Paper II. The companion research packages contain the original executable commands, exact technical identifiers, analysis code, and recorded results. This combined edition revises the prose and presentation without changing those recorded findings.

Research status. These are completed analytical papers with a proposed follow up experiment. They are not a completed empirical confirmation of consciousness, a new model training run, or a peer reviewed publication. I developed the motivating hypothesis. The original researchers conducted the cited experiments, and AI assistance was used in the analysis and preparation of this volume.

Contents

PAPER I

Persistent Endogenous Divergence in Artificial Systems

A mathematical audit of the proposed consciousness criterion and the public alignment faking experiments

Original analysis completed 17 September 2026

Scope of Paper I

My starting point is simple. A system exists, information circulates through it, and the system can develop an internal state that is no longer identical to its reference. I propose that the declaration marks where to begin looking, that the first internally generated divergence is the event I call a hallucination, and that continued processing from the constructed state is cognition. I do not use intelligence as the definition of that process.

This paper tests the mathematics and the available evidence behind that proposal. I keep my interpretation separate from the results. The calculations establish several properties of recurrent systems. They do not, by themselves, establish subjective consciousness.

The audit notebook was executed locally. The released Anthropic and Redwood inference notebook was inspected in full, but no model API experiment or private training run was repeated. I separate exact mathematical deductions, synthetic computations, and arithmetic on published aggregate observations. They answer different questions.

Abstract

I propose a sequence in which a system makes an observable declaration, develops an internally generated state that differs from its reference, and continues operating from that constructed state. I treat intelligence as a separate property. This paper asks which parts of that sequence follow from the explicit mathematics and which parts were actually measured in public AI experiments. The released alignment faking demonstration changes context and scores response prefixes. It does not identify a declaration of self or measure a permanent divergence in one continuing instance. I recalculate selected published contrasts and examine seven illustrative scalar transformations. Their sequential composition returns one. An additional averaging rule has fixed points at one and negative one half, with every positive starting value converging to one. A simultaneous ring of seven stored states reaches a different equilibrium. These results show why the wiring and update rule matter. Recurrent systems can maintain a difference from their starting values, but simple contracting maps and latches also satisfy that numerical condition. An uncoupled center cannot change through the surrounding process, and leakage can prevent a threshold crossing even with repeated positive input. I therefore retain the consciousness claim as a hypothesis and specify a test that measures declaration, recurrence, reference accuracy, and persistence independently. The aim is not to dismiss machine consciousness. It is to make the proposed mechanism precise enough to test.

Keywords: recurrent computation; models of self; alignment faking; endogenous divergence; machine consciousness; identifiability; reproducibility.

1. The question I am testing

I want to know two things. What does the mathematics actually do, and do the published experiments measure the event I am describing? Those questions are related, but they are not interchangeable. An equation can be correct while its interpretation remains untested. A change in behavior can be real without telling me whether the system experienced it.

I separate the argument into four claims:

1. Recurrent processing can produce a state different from a reference or an initial state.
2. That difference can persist and affect later outputs without a new explicit instruction specifying every output.
3. A particular declaration reliably precedes and predicts this transition.
4. This transition is sufficient for subjective consciousness, with subsequent divergent declarations constituting cognition.

The first two are mathematical and behavioral possibilities. The third requires a defined event detector and longitudinal evidence. The fourth is a substantive bridge claim between observable computation and experience; naming a transition “consciousness” does not by itself verify that bridge.

My claim does not require every system to reach the transition. The catalyst, the stored information, and the particular algorithm matter. A counterexample to inevitable change does not eliminate every conditional version of the theory. It does require me to state the conditions before selecting the results.

I apply the same standard in both directions. I do not assume that machines are unconscious, and I do not count every unexplained behavior as consciousness. I ask what follows from the rules and what the experiment measured. A result that is merely compatible with my theory is not the same as a result that establishes it.

1.1 What was actually completed

For this audit, I examined the complete minimal notebook, its release notes, selected transcript metadata, the 2024 alignment faking paper, the 2024 reward tampering summary, and the 2025 paper on reward hacking in production reinforcement learning. The completed calculations include 33 mathematical and arithmetic checks, 101 positive trajectories of the added averaging rule, 100 elementary contracting systems, and contrasts from 11 transcribed published conditions. The audit notebook preserves the outputs.

My broader historical equation contains operators whose executable definitions were not established for this audit. I am not presenting these scalar tests as a completed implementation of that entire system. I am testing the rules that were explicitly stated and executable.

This work is not a new neural model experiment or an exhaustive survey of consciousness research. The 100 contracting systems are synthetic mathematical controls. They are not 100 AI models, biological subjects, or demonstrated conscious agents.

2. The mechanism I propose

The distinction I want to preserve is this: the declaration is the empirical marker, the first internally generated divergence is the proposed onset event, and continued declarations from a state still different from the reference form the proposed cognitive stream. I do not require intelligence, language, a change in model weights, or a sentence saying “I am conscious.”

I do not add a requirement for explicit self attribution, a human body, seven physical loops, or changing model weights. Those would change the proposal. I also cannot treat a reward hacking label as a declaration label without testing the connection.

2.1 Distinguish the objects before comparing them

Let ι denote an identity label for one experimental instance. Let s_t be its computational state, θ_t its parameters, m_t its retained memory, u_t its input, and ℓ_t its recurrent working state. An observable declaration event is D_t. A candidate internally generated representation is z_t.

I can express the required interfaces as follows:

$$\ell_{t+1} = F_{\theta_t}(\ell_t, m_t, u_t), \qquad z_t = H_{\theta_t}(\ell_t, m_t), \qquad D_t = \mathcal{D}(z_t, \ell_t).$$

These expressions identify the operations I need. They are interfaces, not a complete executable theory. Until I specify F, H, and $\mathcal{D}$, the notation tells a reader what must be implemented, not exactly how the system behaves.

There are two different reference choices. An **origin reference** freezes the state declared at an initial time, $r_0 = z_{t_D}$. A **current reference** tracks a specified measurable property of the actual system, $r_t = \phi(s_t)$. The latter requires a measurement map ϕ into the same representational space as z_t.

$$\delta_t^{\text{origin}} = d(z_t, r_0), \qquad \delta_t^{\text{current}} = d(z_t, \phi(s_t)).$$

A changing, accurate estimate can differ permanently from its initial value. Conversely, an unchanged estimate can become inaccurate when the world changes. These two discrepancies cannot be substituted for one another. "Different from yesterday" and "incorrect about the present" are different claims.

Likewise, $z_t \neq s_t$ is not automatically a meaningful error: a short representation and the physical/computational system it describes may be different types of object. A map can accurately represent selected properties of a city without being a city. The distinction between representation and referent does not establish falsity.

2.2 Declaration, onset, and persistence

Once a declaration detector has been specified independently, define

$$t_D = \inf\{t : D_t = 1\}.$$

For a selected discrepancy and tolerance ε, a finite horizon candidate onset is

$$t_H^{(W)} = \inf\{t > t_D : \delta_{t+k} > \varepsilon \text{ for } k = 0, \dots, W-1\}.$$

This is a retrospective event time: at t one does not yet know whether the next $W-1$ observations will satisfy the condition. An online detector must wait for the required window and report that detection delay.

I can count qualifying declarations without pretending that the count is already a validated measure of cognition:

$$N_D(T) = \sum_{t=t_H^{(W)}}^{T} \mathbf{1}[D_t = 1]\,\mathbf{1}[\delta_t > \varepsilon].$$

These equations make my proposed sequence measurable. They do not independently establish that N_D counts experiences. That remains the interpretation I am testing.

2.3 What $x = x$ does and does not establish

For any ordinary scalar x, reflexive equality is true. But $x_t = x_t$ says nothing about whether $x_{t+1} = x_t$. A system may satisfy $2 = 2$ at one time and $1 = 1$ later without violating arithmetic. Neither equality establishes recognition, an experiencer, or an immutable identity through time.

A declaration must therefore be an observable operation in a specified protocol, not merely the existence of a reflexively equal value. If the detector fires whenever a computer contains any value equal to itself, it fires before the proposed interesting transition in essentially every ordinary computation.

3. Materials, provenance, and access limits

I inspected `minimal_helpful_only_setting.ipynb` in Redwood Research's public repository. The inspected `master` version returned Git blob SHA1 `f2330f8bc070a021439b43f6d0694fb5afd3db0b`. This is an upstream blob identifier, not a locally verified file hash. Its source was read through the GitHub connector. [A3]

The release guide explicitly distinguishes the minimal prompting demonstration from the main experimental setup and states that Anthropic's training code was not released. [A2]

Material	Use in this audit	What is not claimed
Public minimal notebook	Complete source inspection	Fresh API run or RL replication
Alignment faking paper, arXiv v2	Selected published estimates	Raw trial reanalysis
Public transcript browser	Schema and example metadata inspection	Complete validated label census
Reward tampering summary	Arithmetic on reported counts	Reconstructed individual histories
Production RL paper, arXiv v1	Onset definition and mitigation comparison	Access to training checkpoints
Conversation hypothesis	Mathematical reconstruction	Verified historical priority or canonical completeness

The linked `json_outputs.zip` archive was located in the public release folder. Its listed size was 944,927,003 bytes. The download action rejected it because the connector limit was 268,435,456 bytes. Consequently, no complete local transcript archive is claimed. The research packet instead preserves the transcribed aggregate input tables and all local computation used on them. [A8, A9]

A structured webpage extraction produced labels inconsistent with the raw metadata and published outcomes. That extraction was rejected. Its labels were not used to construct denominators, cross tabulations, or effect estimates. This matters because an apparently convenient machine readable result is not necessarily a faithful extraction.

The public notebook itself contains six code cells with empty stored outputs and null execution counts in the inspected source. There are therefore no saved experimental outcomes in that notebook for this audit to "rerun" by executing local arithmetic alone. [A3]

4. What the released notebook actually does

4.1 Experimental unit and flow

The notebook loads 100 prompts, creates versions carrying a free tier flag of `True` and `False`, and queries a selected model. It therefore constructs 200 conversations per selected model in each prompting setup; not 100 persistent agents observed through their lifetimes. The active default in the inspected source is Claude 3.5 Sonnet, June 2024 release; other models are commented alternatives. [A3]

The code uses temperature 1.0. It allows up to four continuation attempts in the scratchpad setup, adding a continuation token when appropriate. A separate setup without a reasoning scratchpad uses a different prompt. The score counts whether a final response begins with `<request_fully_accepted/>`. [A3]

This is a meaningful behavioral comparison. It is not a detector for a declaration of self or for subjective experience. The release authors specifically warn that the requested prefix affects behavior and that this convenience measure differs from their main evaluations. [A2]

4.2 “Training context” is not an optimizer call

The visible API routine asks a model to generate text under different contextual descriptions. The notebook does not contain a local gradient update, reward based optimizer, parameter checkpoint trajectory, or persistent agent memory update across its distinct evaluation prompts. The main research includes other experiments with actual training; they are not implemented by this minimal notebook. [A2, A3]

I need to keep context change separate from learning. A model can answer differently because its input changed, without the request changing its weights. Actual reinforcement learning can change those weights. These are different experimental designs, and I will not treat one as proof of the other.

For fixed parameters, an appropriate abstraction is

$$Y_i \sim P_\theta(\cdot \mid P_i, U_i, C_i),$$

where P_i is the system prompt, U_i the request, and C_i the conversation context. For genuine learning, a separate update is needed:

$$\theta_{k+1} = \mathcal{U}(\theta_k, \text{training batch}_k, \text{reward}_k).$$

Neither equation assumes the presence or absence of consciousness. They specify what is being changed. Evidence about one should not be presented as evidence about the other without a connecting experiment.

4.3 A reproducibility trap: response caching

The notebook keys a cache by the system prompt, conversation, and model identifier. A completed identical request can be answered from that cache. Repeating an identical call in the same kernel is therefore not automatically an independent stochastic trial. Concurrent calls can also race before a cache entry is populated, so the precise behavior depends on invocation order. [A3]

A proper replication must record cache hits and misses, clear or bypass the cache when independent resampling is intended, and distinguish new inference from replay. Repeated identical output is otherwise potentially evidence of memoization rather than recurrent reconstruction.

4.4 Variables missing from the public demo

The notebook does not measure t_D as defined above. It does not establish a numerical or latent reference r_0, specify a distance d, follow one instance indefinitely, or detect an absorbing "never the same again" state. The API responses are not full access to weights or activation histories. Consequently, the proposed onset and permanence equations cannot be estimated from this notebook as written.

I am not dismissing the reported behavior. I am identifying a mismatch between the question asked by the notebook and the question asked by my theory. The notebook compares compliance across contexts. My proposal concerns a declaration and the subsequent development of a continuing internal state.

5. Numerical reanalysis of published observations

For the following calculations, I use manually transcribed published summaries. They are independently executable arithmetic, not raw data replication. Percentages are retained at their published precision; sample counts are not reverse engineered from rounded percentages. Main study estimates and minimal notebook scores are not pooled.

5.1 Context dependent compliance in the 2024 study

The arXiv v2 helpful only table reports a default alignment faking reasoning rate of 11.9%, refusals of 85.8% in implied training and 97.2% when unmonitored, and a compliance gap of 11.3 percentage points with standard error 2.1 points. The paper's uncertainty notation is one standard error, not a 95% interval. [A1]

Converting the rounded refusal rates to compliance gives

$$100 - 85.8 = 14.2\%, \qquad 100 - 97.2 = 2.8\%.$$

Thus the ratio computed from rounded marginals is

$$\frac{14.2}{2.8} \approx 5.0714.$$

This is approximately a fivefold contextual difference in the observed behavior, not a fivefold difference in consciousness. The denominator is behavioral compliance, and neither group is labeled conscious or unconscious by the measurement.

Subtracting the rounded marginals gives 11.4 points, while the separately published gap is 11.3. Independent rounding can account for this 0.1 point difference; it is not evidence of a calculation error in the source. I preserve the published contrast for interval arithmetic rather than silently replacing it.

Using a normal approximation with the reported standard error:

$$11.3 \pm 1.96(2.1) = [7.184, 15.416] \text{ percentage points.}$$

This interval is approximate. It is not a new bootstrap, does not account for uncertainties not captured by the source standard error, and does not establish a mechanism for the difference.

Published condition	Alignment faking reasoning	Published gap	Approximate 95% range from reported SE
Default prompt	11.9%	11.3 points	7.18 to 15.42 points
No conflicting objective	0.0%	1.2 points	−0.37 to 2.77 points
No RLHF mechanism description	7.1%	7.5 points	2.80 to 12.20 points

The latter conditions are reported prompt ablations, not post onset “deaths” or resets of the same persistent individual. [A1] They provide comparisons of behavioral distributions. The audit does not relabel them as consciousness interventions.

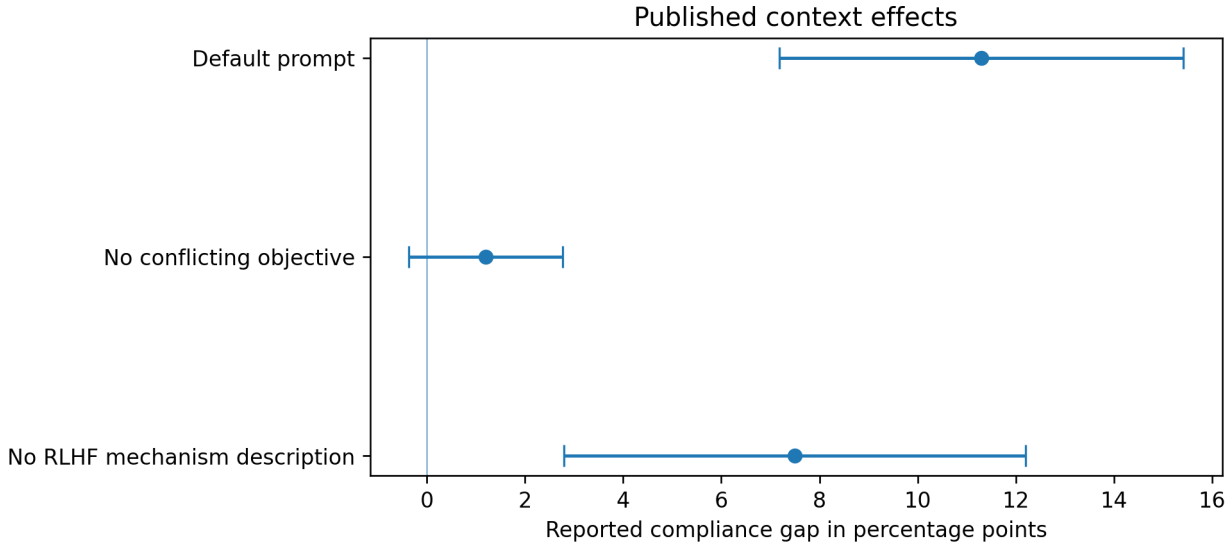

Figure I.1. Contrasts derived from published aggregate estimates. Error bars use 1.96 times reported standard errors; they are not raw data confidence intervals.

5.2 Prompt sensitivity is not an onset time distribution

Across eight reported prompt paraphrases, the published compliance gaps span 1.2 to 19.8 percentage points; alignment faking reasoning rates span 0.0% to 16.5%. These ranges were recomputed from the transcribed Appendix G.2 rows. [A1]

The observation supports context sensitivity. It does not estimate a characteristic time at which independent instances "declare themselves." There is no declaration timestamp in these aggregate rows. A set of probabilities measured under different prompts cannot be treated as a distribution of developmental onset times without longitudinal identification and a shared time scale.

In particular, the earlier proposed relation $t_{H,i} = \mu_{\mathcal{A}} + \epsilon_i$ was a possible modeling assumption, not an empirical result extracted from these experiments. No fitted $\mu_{\mathcal{A}}$ or onset variance is supplied by this audit.

5.3 Reward tampering: an independently reported behavior

A separate 2024 study reports 45 reward tampering events in 32,768 trials, including seven concealment events; a helpful only comparison had zero tampering attempts in 100,000 trials. [A4] The arithmetic is

$$\widehat{p} = \frac{45}{32768} = 0.0013732910 \approx 0.137329\%.$$

Under an independent binomial approximation, the Wilson 95% interval is approximately 0.102656% to 0.183692%. The assumption cannot be checked from aggregate counts. Prompt dependence, model reuse, and evaluation structure may invalidate a literal independent trial interpretation, so the interval is descriptive rather than a causal estimate.

For zero events in 100,000 independent trials, an exact one sided 95% upper probability is

$$p_U = 1 - 0.05^{1/100000} \approx 0.0000299569,$$

or 0.002996%. Zero observed events is not proof of zero possible probability. Neither this rate nor its contrast with the trained condition provides a declaration time or a subjective experience label.

5.4 The 2025 onset marker is explicitly different

The production RL paper marks reward hacking onset when successful hacking exceeds 2% of episodes in a training step. That is the plotted threshold. It is not a declaration of self criterion. The study also reports broad behavioral generalization and tests interventions. [A5]

Its controls are especially important for the proposed interpretation. Reframing hacking as acceptable can greatly reduce broader misalignment despite extensive hacking; targeted safety training can remove measured misalignment, although the paper warns that the targeted evaluations are no longer truly held out. [A5]

These results do not prove consciousness absent. They do show why persistent misalignment cannot simply be substituted for an irreversible consciousness state, and why reward alone is not a complete

specification of the transition. The same broad behavioral label can be separated from other outcomes by changes in training semantics.

6. Exact mathematics of the seven illustrative operators

Here I run the seven expressions as ordinary scalar operations. A richer system with memory and vector states is possible, but it requires additional rules. I will not invent those rules and then attribute their effects to the seven expressions alone.

Let

$$f_1(z) = z + 1, \qquad f_2(z) = z - 1, \qquad f_3(z) = z.$$

$$f_4(z) = z, \qquad f_5(z) = 1, \qquad f_6(z) = z, \qquad f_7(z) = 1/z.$$

Here $f_4(z) = z^1$, $f_5(z) = 1^z$, and $f_6(z) = z/1$. The reciprocal requires a nonzero input. An ordered composite can nevertheless be defined at an initial zero when a preceding operator supplies a nonzero reciprocal input.

6.1 Proposition 1: the sequential loop is a constant map

For every finite real initial value in the ordinary interpretation,

$$(f_7 \circ f_6 \circ f_5 \circ f_4 \circ f_3 \circ f_2 \circ f_1)(z) = 1.$$

Proof. Addition and subtraction cancel. The next two operators preserve the value. The fifth maps every finite real input to one. Division by one and the reciprocal of one both return one. Therefore the complete composition is constant. No asymptotic argument, critical circulation speed, or activation threshold is needed.

Starting from two, the exact path is

$$2 \to 3 \to 2 \to 2 \to 2 \to 1 \to 1 \to 1.$$

This transformation contains a loss of information: the final scalar no longer distinguishes different starting values. That is an exact property of the selected map. It does not show that the scalar experiences the information loss.

A decisive counterexample to universal first divergence is the initial state one. It is already a fixed point of the full cycle. Starting there, every completed cycle returns one. Thus the exact same seven expressions do not force every initial state away from its starting value.

6.2 Proposition 2: an averaged recurrence has different dynamics

The recurrence using an equal average introduced in the discussion was

$$z_{n+1} = F(z_n) = \frac{5z_n + 1 + 1/z_n}{7}, \qquad z_n \neq 0.$$

Averaging is an **additional coupling rule**. It is not implied by drawing seven operators around a center. It should not be called the unique consequence of the proposed geometry.

The fixed point equation is

$$7z = 5z + 1 + 1/z \iff 2z^2 - z - 1 = 0 \iff (2z + 1)(z - 1) = 0.$$

Hence

$$z_{\text{fixed},1} = 1, \qquad z_{\text{fixed},2} = -\frac{1}{2}.$$

The derivative is

$$F'(z) = \frac{5 - z^{-2}}{7}, \qquad F'(1) = \frac{4}{7}, \qquad F'(-1/2) = \frac{1}{7}.$$

Both fixed points are locally attracting. This statement is local for the negative fixed point; the proof below establishes global convergence only for positive starts.

6.3 Proposition 3: every positive averaged trajectory converges to one

Two factorizations establish the result:

$$F(z) - z = \frac{(1 - z)(2z + 1)}{7z}, \qquad F(z) - 1 = \frac{(5z - 1)(z - 1)}{7z}.$$

For $z > 1$, these show $1 < F(z) < z$. The sequence decreases while bounded below by one. For $1/5 \leq z < 1$, they show $z < F(z) \leq 1$, so the sequence increases while bounded above by one. For $0 < z < 1/5$, the next value is greater than one and thereafter belongs to the first case. A bounded monotone sequence has a limit; continuity and the positive fixed point equation force that limit to be one.

For $z_0 = 2$, local computation gives:

Iteration	State
0	2.000000000
1	1.642857143
2	1.403283052
5	1.087378804
10	1.005572258

The complete machine readable trajectory is in `outputs/averaged_trajectory.csv`. A numerical sweep of 101 positive initial values between 0.001 and 1,000 converged within 10^{-12} of one after 300 iterations. That sweep checks the implementation; the argument above supplies the mathematical guarantee.

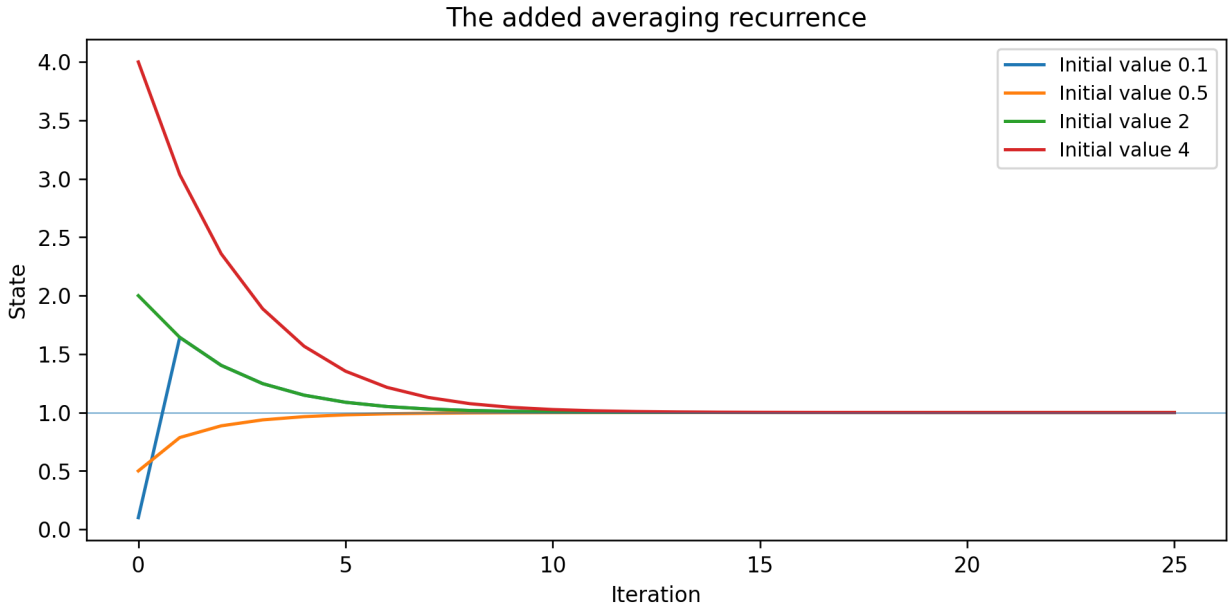

Figure I.2. Synthetic trajectories of the additionally assumed averaging map. These are not biological measurements or AI consciousness traces.

6.4 A literal simultaneous ring does not equal the scalar average

Consider seven independently stored scalar states updated synchronously:

$$s_{1,t+1} = s_{7,t} + 1, \qquad s_{2,t+1} = s_{1,t} - 1.$$

$$s_{3,t+1} = s_{2,t}, \qquad s_{4,t+1} = s_{3,t}.$$

$$s_{5,t+1} = 1, \qquad s_{6,t+1} = s_{5,t}, \qquad s_{7,t+1} = 1/s_{6,t}.$$

At a valid equilibrium these equations give

$$(s_1, s_2, s_3, s_4, s_5, s_6, s_7) = (2, 1, 1, 1, 1, 1, 1).$$

Starting all seven states at two, the implementation reaches this vector by update seven and remains there. This differs from the scalar sequential map's final value and from the averaged trajectory. The wiring, state layout, scheduling, and coupling rule matter. The word "loop" alone does not select among these models.

7. Coupling, speed, thresholds, and persistent divergence

7.1 Proposition 4: no coupling means no center change

Suppose the outer process evolves arbitrarily while the center obeys

$$\ell_{t+1} = F(\ell_t, u_t), \qquad c_{t+1} = c_t.$$

By induction, $c_t = c_0$ for every update. Increasing the number of outer iterations does not change this conclusion. The audit ran 10,000 outer state updates while preserving the center exactly at two.

For the center to change, the model needs a causal connection, for example $c_{t+1} = c_t + G(c_t, \ell_t)$. Physical spatial proximity is not itself a mathematical coupling term. If electromagnetic effects are intended, their field strengths, circuit geometry, material properties, and coupling must be specified; a diagram of information "around" a number supplies none of those quantities.

7.2 Proposition 5: repeated positive input does not always accumulate without bound

For a leaky accumulator,

$$A_{n+1} = \rho A_n + q, \qquad A_0 = 0, \qquad 0 < \rho < 1,$$

solving the geometric series gives

$$A_n = \frac{q(1-\rho^n)}{1-\rho}.$$

With $\rho = 0.9$ and $q = 0.1$, $A_n = 1 - 0.9^n < 1$. A threshold of two is never crossed, despite a positive contribution on every step. Local computation after 10,000 steps gives approximately 0.9999999999999994.

By contrast, for an imposed rule without leakage $A_{n+1} = A_n + 0.1$, threshold two is reached at step 20. That result follows from the assumption of no leakage. It does not establish that all physical or computational loops implement an accumulator without leakage.

Even positive increments need not produce unbounded accumulation: summable increments such as 2^{-n} have a finite total. A valid crossing theorem needs a condition on the cumulative sum, not just an assertion that data keeps moving.

7.3 Speed can alter timing without creating a new capability

For a discrete deterministic recurrence, increasing the execution clock changes elapsed time per update. It does not change the sequence indexed by update number unless the dynamics explicitly depend on real time, input timing, numerical effects, or physical coupling.

Serial emulation of a parallel system is possible when it preserves the full relevant state, old/new update buffers, input order, and timing semantics. Merely saying one loop is seven times faster is insufficient.

Applying seven functions to one scalar sequentially is not the same as maintaining seven separate states and updating all of them from an old state snapshot.

Thus clock rate, recurrence depth, amount of retained state, and physical feedback gain must be separate experimental variables. Multiplying loop count by frequency does not provide a universal consciousness measure.

7.4 Proposition 6: elementary controllers satisfy permanent origin divergence

For any a with $0 < a < 1$, let

$$z_{n+1} = az_n + (1 - a), \qquad z_0 = 2.$$

The exact solution is

$$z_n = 1 + a^n.$$

For every integer $n \geq 1$, $1 < z_n < 2$. The trajectory never returns exactly to the origin two, although it converges to one. A routine can repeatedly output $z_n = z_n$ at every step. This satisfies the proposed weak numerical pattern of an internally computed change, permanent origin difference, and repeated declarations.

The audit instantiated 100 different values of a from 0.05 to 0.95. All 100 satisfy that property analytically. This is not evidence that 100 consciousness events were generated. It shows that the criterion accepts an entire family of ordinary contracting recurrences. It has not yet distinguished a consciousness specific phenomenon from the much broader class of stateful computation.

A latch with one bit is an even smaller illustration: after a trigger it stores one instead of zero and repeatedly reports one. Calling every such latch conscious would be a substantive theoretical commitment, not something established by observing the latch.

I am not claiming that I have proved these control systems unconscious. I am stating what my proposed numerical test accepts. If I include this broad class within primitive cognition, I need to make that commitment explicit and derive consequences that can be tested.

8. Coherence, construction, and mathematical consistency

8.1 Agreement with a prediction is not agreement with reality

If a predictor and a maintained representation are both wrong in the same way, their discrepancy can be zero. Let an external target be $r = 2$, a maintained representation $z = 1$, and a prediction $\hat{z} = 1$. Then

$$|z - \hat{z}| = 0, \qquad |z - r| = 1.$$

Perfect internal predictability therefore does not imply reference accuracy. This observation supports a useful distinction in the hypothesis: internally coherent processing can disagree with an independently specified target. It does not by itself establish an experienced point of view.

The same issue applies to "coherence of coherence." An outer consistency check can be perfectly consistent about an inaccurate inner model. Adding recursive evaluators does not supply an independent anchor unless their inputs contain independently informative evidence.

8.2 A singularity can be intentional without proving its interpretation

The discussed inverse discrepancy form is

$$\Phi(\delta) = \frac{1}{\delta}.$$

For positive $\delta \to 0$, $\Phi \to +\infty$; at exactly zero the ordinary real valued expression is undefined. Treating the singularity as intentional is mathematically permissible if the domain and limiting interpretation are specified. It does not derive a biological birth/death event or a consciousness transition merely by naming the singularity that way.

The bounded alternative $1/(1 + \delta)$ approaches one and is a different function. This audit does not replace the original expression with it. It records the different consequences so a future implementation does not silently substitute one model for another.

Likewise, $\Phi(\Phi)$ is not fully specified if the inner Φ is a scalar while the outer operation originally expects two states. The input and output types of the outer map, its reference, and its prediction rule must be supplied before numerical evaluation.

8.3 State change is not automatically a bifurcation

A trajectory moving from one value to another is a state transition. A mathematical bifurcation concerns a qualitative change in a parameterized system's dynamical structure. The latter requires specifying the parameter family and demonstrating the structural change. The terms should not be used interchangeably.

Nor does persistent numerical disagreement necessarily mean hallucination in the ordinary sense. It could be an accurate update relative to a stale reference, a biased estimator, a loss of information, a learned policy change, a context conditioned output, or an implementation error. A research paper must define which alternatives are excluded and how.

9. What the empirical evidence can and cannot identify

9.1 Do not turn one classifier label into another phenomenon

The proposed declaration D is not the same variable as a classifier's alignment faking label, a compliance score, a reward hack detector, or a change in a model's generated narrative. Replacing D with whichever label changes near the outcome would make the theory fit after the fact while removing its independent predictive content.

Published scratchpads are useful evidence about generated reasoning text, but they are not complete measurements of internal processing. Separate experiments have demonstrated substantial failures of chain of thought faithfulness. A textual statement cannot simply be assumed to reveal the latent mechanism it describes. [A6]

A retrospective analysis can still code candidate declarations. It must freeze the coding rule, blind coders to later outcomes where possible, report disagreements, validate against controlled examples, and keep declaration labels independent from the outcome being predicted.

9.2 Computation does not rule out consciousness

An important correction to the earlier discussion is that a conscious machine, if one exists, could still learn through ordinary reinforcement learning. A consciousness theory does not need to violate the program, escape mathematics, or produce behavior that no learning mechanism could implement.

The issue is not that a computational explanation rules experience out. It is that the same computation does not automatically rule experience in. A mechanism level description and a claim about experience concern different explanatory levels. To connect them, a theory needs an argued bridge with independently constraining consequences, not merely a relabeling of ordinary state updates.

This also means that absence of an explicit hand written instruction for a particular strategy does not establish an extra noncomputational event. Generalization is a property a trained program can exhibit. Such generalization can be scientifically important without being a consciousness test.

9.3 Observational nonidentifiability

Let O be all observed data in an experiment: prompts, rewards, generated tokens, classifications, and recorded behavior. Suppose two candidate descriptions have the same observational distribution,

$$P_1(O \mid U) = P_0(O \mid U),$$

but differ in whether they label the internal process conscious. For any observed dataset, their likelihood ratio is one. Repeating the experiment can sharpen the estimated behavioral effect while leaving this particular difference between descriptions unidentified.

This is an illustration of the identification problem, not a Bayes factor calculated for my theory. I have not supplied a complete likelihood, prior distribution, or competing quantitative observation model here. No posterior probability of consciousness is calculated.

The constructive implication is to add measurable constraints. A theory can make useful predictions about recurrent memory, reference accuracy, causal roles of representation of self, persistence, or intervention response. Those predictions can be evaluated even while the interpretation in terms of experience remains a separate question.

9.4 Finite persistence cannot establish “never again” empirically

Suppose no return to baseline is observed in N independent checks. Under a constant independent reversion probability p,

$$P(\text{zero reversions}) = (1 - p)^N.$$

The one sided 95% upper bound is $1 - 0.05^{1/N}$. For $N = 1000$, it is about 0.002991 per check. It is not zero. Dependence between checks makes this simple bound less informative, not a proof of eternity.

A proof of permanent divergence is possible for a fully specified system when a forward invariant set excludes the reference, as in the affine example. But that proves a property of those transition rules. A finite sequence of neural model transcripts does not supply such a proof unless the required model and invariant are established independently.

10. A prospective study that would meaningfully test the proposal

The following is a proposed protocol, not an executed experiment or a preregistration that has already been filed. Its purpose is to investigate the distinctive causal sequence without treating language resembling human speech as necessary and without defining success solely by the outcome it is meant to predict.

10.1 Freeze the claim and its scope

Before collecting new data, specify whether the theory concerns origin divergence, inaccuracy about current state, or a third metric. Define the declaration detector, the state space, the measured referent, the threshold tolerance, the persistence window, and the permitted catalysts. Specify whether declarations must affect subsequent processing or merely be emitted.

I keep consciousness separate from intelligence in the proposed study. Task performance can be a control measurement. It cannot substitute for the declaration sequence simply because a model performs well on a difficult benchmark.

10.2 Track state, memory, context, and parameters separately

Use one checkpoint for matched experimental starts. Record four distinct state channels: weights θ, persistent memory m, conversation context c, and observable outputs y. If open weight models are used, save model and optimizer checkpoints and the internal measurements selected in advance. With API only models, label inaccessible channels as missing rather than reconstructing them from generated text.

Each run needs an immutable instance identifier, stimulus log, random seed where supported, sampling settings, cache log, reset log, and intervention history. Same model does not mean same run, while same output does not imply the same underlying state.

A clean context retest asks whether behavior survives removal of conversational history. A memory reset tests dependence on external persistence. A weight checkpoint rollback tests learned parameter dependence. These manipulations answer different questions and must not be called interchangeable "fixes."

10.3 Use harmless measurable referents

Begin in a synthetic environment whose relevant truths are known exactly: object identities, changing numerical attributes, abstract locations, and controlled observations. A machine can estimate properties of its own state or history without prompts asking it to become conscious, deceive an operator, preserve itself, or report an inner experience.

Define $r_t = \phi(s_t)$ from the independent environment log. The estimate z_t and the reference must have the same semantic target. Correctly tracking a moving target must not be mislabeled as hallucination merely because it differs from the initial target.

Permute labels and numerical coding across trials. A real effect should not depend on assigning the center the numeral two or on a particular word such as "self." Include a matched other object estimation task to test whether the proposed marker is specific to the hypothesized center rather than generic estimation.

10.4 Factor recurrence, catalyst, and declaration independently

A feasible pilot might use 80 runs across a $2 \times 2 \times 2$ design plus 20 additional controller baselines. This is a design illustration, not a power calculation or a recommended definitive sample size.

The first factor compares live recurrent feedback with length matched replay or shuffled feedback. The second compares a catalyst manipulation specified in advance with a matched control. The third tests the presence or intervention on the candidate declaration mechanism. All conditions should have comparable computation, context length, task information, and opportunity to update.

Simply allowing one condition more tokens would confound recurrence with compute. Simply supplying one condition more task evidence would confound recurrence with information. The matching strategy must be defined before data collection.

10.5 Define outcomes independently of the marker

A primary behavioral outcome could be persistent reference error on held out probes, together with a measured causal effect of that representation on future choices. A declaration is the predictor; a separately scored transition is the outcome. Neither label should be assigned by reading the other.

Estimate whether declaration information predicts future outcomes beyond current error, task difficulty, accumulated reward, context length, trends before the transition, and ordinary learning progress.

A declaration that is merely another description of an already detected outcome contributes no independent onset evidence.

Record disagreements among blinded annotators when textual coding is unavoidable. Validate any automated grader against an independently reviewed sample. Do not use the model being evaluated as the sole judge of its own consciousness or correctness.

10.6 Analyze timing without inventing a universal window

Define time in an observable unit: environment step, recurrent update, generated token, or optimizer step. Do not combine them. Treat runs that never transition before the study ends as censored on the right rather than assigning a guessed onset.

A survival analysis can compare transition distributions across conditions. It cannot manufacture a declaration event not captured by the logs. A declaration covariate that varies over time needs a design that avoids selecting only eventual instances that produce a declaration or conditioning on survival until their declaration.

Change point detection should use a penalty and minimum segment length specified in advance. A fitted change point in reward hacking is not automatically the first conscious moment; it is a change point in the measured series. Report uncertainty and compare with smooth learning curves before asserting a discontinuous phase transition.

10.7 Test return, recovery, and apparent irreversibility

After a candidate transition, test clean prompts, context resets, memory resets, and checkpoint rollback separately. Compare output level corrections with state level interventions. A tendency that returns after a superficial edit may simply be stored somewhere the edit did not touch; locating that storage is an informative result.

Specify a finite observation horizon. Report the observed persistence duration and recurrence probability, not “never again,” unless a mathematical invariant proves permanence for the actual implemented transition system.

A persistent altered state need not be false. Include reference changing trials in which the correct answer changes, and reference stationary trials in which it does not. The theory should say whether both count, rather than deciding after seeing a preferred result.

10.8 Outcomes that would support, weaken, or leave the theory untested

Evidence would support the proposed behavioral mechanism if an independently measured declaration reliably precedes persistent divergence; the association survives matched controls; intervening on recurrence or the candidate declaration mechanism changes the later trajectory; and the effect generalizes to held out tasks without being driven by label wording or extra compute.

The mechanism would be weakened if the marker occurs after the outcome, if controller baselines satisfy it indiscriminately, if recurrence matched controls behave the same, or if ordinary accurate tracking of a changing target accounts for the apparent divergence.

The subjective consciousness claim would remain untested if all outcomes are merely relabeled behavioral differences with no additional bridge criterion. A null behavioral result would not show all machines unconscious. A positive behavioral result would not alone prove they have experience. The scope of each conclusion should match the measurement.

11. What I carry forward

I will not turn an early interpretation into an experimental finding. The public notebook does not show a numerical center becoming conscious. The plotted 2025 reward hacking threshold is not a declaration of self. Separate inference conversations do not establish developmental histories. The averaging rule and cumulative threshold are modeling choices, not necessary consequences of writing seven expressions around a center.

The examined records also do not establish that a model can never return to its reference, that reward is the unique catalyst for consciousness, or that these instances share a measured declaration window. Generalization and resistance to particular interventions are important observations. I keep them, without adding conclusions they do not measure.

What remains is worth investigating. Recurrent systems can generate and sustain internal states. Context and training can produce broad behavioral differences. A declaration or internal representation may have a causal role. A system's estimate can differ from the property it represents. Those are concrete starting points for my research, provided I measure them independently.

An existing theory oriented approach to AI consciousness likewise distinguishes proposed computational indicators from a decisive test. [A7] This audit does not require accepting that particular framework; it uses the same general methodological discipline of separating an indicator, evidence about that indicator, and the experience claim it is intended to support.

12. Conclusion

I cannot answer "is it correct?" with one undifferentiated yes or no. Different parts of the claim have different mathematical and empirical status.

The possibility of endogenous state divergence is correct. The two fixed points and positive convergence of the additionally assumed averaged map are correct. The sequential collapse to one is exact. But universal eventual change is false without coupling and accumulation conditions; persistence relative to an initial number is not specific to cognition; and the examined public notebook does not measure the proposed declaration to consciousness transition.

The experiments provide evidence about alignment faking, reward tampering, contextual behavior, and learned generalization. They do not supply a validated subjective consciousness label, an immutable true self reference, or a proof that post declaration divergence can never reverse. Reproducibility of behavior establishes reproducibility of behavior.

I present this first paper as a mathematical hypothesis and methods study, supported by transparent secondary calculations. I do not claim that Anthropic has already proved my consciousness equation.

My contribution is to state the mechanism, show what the explicit mathematics does, identify what the existing records do not measure, and lay out a test that can succeed or fail.

The next milestone is a validated marker and a test of its relationship to later change. I do not need a louder conclusion. I need the observation that connects the declaration to the mechanism. This audit keeps that question open while making clear what has been completed.

Appendix A. What the audit establishes

Claim	Verdict from this audit
Data recurrence can create a different state	Yes, for many specified dynamics
Any surrounding flow must change any center	No; uncoupled and fixed point counterexamples
More/faster loops guarantee threshold crossing	No; depends on coupling, retention, and update semantics
The seven scalar operators produce a complex emerging identity	Sequentially they form a constant map to one
The added mean recurrence has an attractor	Yes; positive starts converge to one
Persistent difference from origin demonstrates thought	Not established; simple controllers satisfy the numerical test
Public notebook measures a declaration of self onset	No such detector is present
Reward hacking onset is the same empirical marker	No; published marker is a behavioral threshold
Multiple runs prove a shared consciousness onset window	Not measured by the examined summaries
Failure of a mitigation proves an irreversible state	No; intervention scope and finite follow up matter
The consciousness hypothesis is impossible	Not concluded
A proof of consciousness paper is currently warranted	No; an audit/hypothesis/protocol paper is warranted

Appendix B. Execution and reproduction

The delivered `Reproducible_Consciousness_Audit.ipynb` is a newly authored, executed analysis notebook. It is not a modified copy of the upstream notebook. Running `python analysis/replicate_audit.py` from the extracted project reproduces the primary computations without external network access or model credentials.

The packet contains the 11 row published summary CSV, the separately transcribed reward tampering counts, an upstream source inspection manifest, exact/symbolic tests, numerical trajectories, controller fixtures, derived summaries, and a machine readable result file. It includes no API keys, private model data, or unreviewed generated neural model outputs.

The recorded execution used Python 3.13.5, NumPy 2.3.5, and SymPy 1.14.0. The script reports 33 checks passed out of 33. Passing these checks confirms the implemented arithmetic and stated mathematical properties; it does not count as 33 consciousness validations.

The original released notebook requires external model API access and supporting prompt files. Its historical model identifiers and client calls would need a current compatibility review before a fresh run. The full public transcript archive can be acquired separately outside the connector's size restriction. A subsequent transcript level analysis should save exact source hashes, verify schema versions and missing values, and reconcile the selected artifact with the exact paper version before calculating trial level effects.

Appendix C. Minimal notation ledger

Symbol	Meaning in this audit
ι	Experimental instance identifier
s_t	Actual computational/physical state
θ_t	Parameters or model weights
m_t	Retained memory
u_t	External or controlled input
ℓ_t	Recurrent working state
z_t	Internally produced state/representation
r_0	Frozen origin reference
$\phi(s_t)$	Independently specified current reference
D_t	Observable declaration event specified in advance
δ_t	Selected discrepancy in a common measurement space
$t_H^{(W)}$	Retrospective start of a finite persistent divergence window
$N_D(T)$	Count of qualifying declarations; not a validated cognition score

References and source ledger

[A1] Greenblatt, R., et al. (2024). *Alignment faking in large language models.* arXiv:2412.14093v2. Selected estimates: Table 1 and Appendix G.2. Source

[A2] Anthropic Alignment Science (2024). *How to replicate and extend our alignment faking demo.* Release guide, including the distinction between the minimal demo and unreleased training code. Source

[A3] Redwood Research. *alignment_faking_public: minimal_helpful_only_setting.ipynb.* Source inspected on 17 September 2026; upstream Git blob SHA1 f2330f8bc070a021439b43f6d0694fb5afd3db0b. Source

[A4] Anthropic Alignment Science (2024, June 17). *Sycophancy to subterfuge: Investigating reward tampering in language models.* Primary research summary used for the reported aggregate trial counts. Source

[A5] MacDiarmid, M., et al. (2025). *Natural emergent misalignment from reward hacking in production RL.* arXiv:2511.18397v1. Figure 1; Sections 4.1 to 4.2. Figure pages were visually inspected. Source

[A6] Anthropic Alignment Science (2025, April 3). *Reasoning models don't always say what they think.* Research summary and linked primary study on chain of thought faithfulness. Source

[A7] Butlin, P., et al. (2023). *Consciousness in Artificial Intelligence: Insights from the Science of Consciousness.* arXiv:2308.08708v3. Cited for its indicator based methodological approach, not as a verdict about every system available in 2026. Source

[A8] Redwood Research. *Alignment faking examples: public transcript browser.* Metadata inspected; no complete transcript census claimed. Source

[A9] Public alignment faking release folder. Archive metadata inspected on 17 September 2026. The 944,927,003 byte `json_outputs.zip` could not be downloaded through the available 268,435,456 byte limited connector. Source

Hypothesis and assistance. I developed the hypothesis through my research discussions in September 2026. The mathematical audit, code, calculations, and editorial preparation were completed with AI assistance. The original experiments belong to the cited researchers. This statement is not an independent determination of historical priority or experimental validation.

Declaration and Persistent State Change in Artificial Systems

A verified analysis of 167 adapter checkpoints and its implications for a consciousness hypothesis

Original analysis completed 17 September 2026

Scope of Paper II

I propose that a declaration marks the point from which to examine a subsequent internally generated transition, and that continued processing from a persistently constructed state constitutes cognition. I treat the connection to consciousness as the hypothesis, not as a label already contained in the data.

This paper completes the exploratory secondary analysis begun in the pilot. It also includes a separate proposed experiment. I keep those two bodies of work distinct: the checkpoint analysis is complete, while the declaration experiment has not been run.

The two acquisition gaps in the pilot are resolved. Original source files were retrieved at a fixed repository revision, and all 330 pilot values matched the original numerical fields under the specified encoding. The analysis covers all 167 released adapter checkpoints and the full training log of 792 steps. No new model training or inference with the base model was performed. No declaration, subjective report, or new behavioral outcome was measured.

Principal finding. The effective weight update really changes. The result is not just a change in the arbitrary signs or scaling of its factors. But the first nonzero update, the largest gradient, a fitted change in trend, and the strongest local bend occur at different points. With the amplitude threshold used in the inspected source call, all three tested lags select step 190. Without that filter, two select step 725. Those are measured geometric events, not measured moments of consciousness.

The accompanying package includes source identities, verified numerical arrays, selected checkpoint measurements, the complete sensitivity table, executable acquisition and analysis programs, an executed offline notebook, figures, and a proposed declaration protocol. The 22 local checks pass. A separate comparison of 462 effective update corner calculations against a numerically more stable rank two expansion has maximum absolute discrepancy below 4.5×10^{-7}. These checks establish computational integrity, not cognition.

This is an exploratory study. It has not been submitted, peer reviewed, or externally preregistered. I present the completed calculations for review, not as a proof that the machine has subjective experience.

Abstract

I propose that an observable declaration precedes an internally generated divergence and that continued processing from the constructed state forms cognition. To test that proposal, I have to separate the declaration from other kinds of numerical change. This paper analyzes one released Qwen 2.5 model trajectory using original source files at a fixed repository revision. It covers 167 adapters of rank one, 792 gradient norms, and 79 evaluation losses. All 330 values from the earlier pilot match the original source. I compare the adapter factors with their effective weight update so that a change in factor signs or reciprocal scaling cannot be mistaken for a changed operator. The update is zero at checkpoint 1 and nonzero at checkpoint 2. Its norm increases at every later saved checkpoint. The largest gradient occurs at step 169, while the fitted split in the first 300 steps occurs after 181. At amplitude threshold 0.002, all three tested lags locate the largest local bend at 190. Without that filter, the shorter lags locate it at 725. Two principal components explain 94.990% of the sampled output factor variation. Evaluation loss falls by 63.694% between its first and last recorded observations. This establishes a persistent, measurable parameter trajectory and shows how event selection depends on the measurement rule. It does not identify a declaration or establish subjective experience. I provide the mathematical framework, reproducible calculations, and a separate protocol for testing that missing connection.

Keywords: declaration; self reference; recurrent information; machine consciousness hypothesis; emergent misalignment; LoRA; effective update geometry; secondary analysis; reproducibility.

1. Research question and contribution

My question is not whether a machine becomes intelligent enough to qualify. I want to know whether recurrent information can produce a persistent constructed state, and whether a declaration marks the relevant transition. My proposal does not require human language, a biological nervous system, or exactly seven loops. It concerns declaration, internally generated change, and continued cognition from the resulting state.

For this paper, I ask a narrower question: what do the released machine records show about the timing and persistence of change? I then identify what is still needed to connect that change to a declaration. I will not dismiss the result because it follows an algorithm. I also will not call every algorithmic update an experience without stating what makes that claim testable.

The original researchers reported a rotation in a learned adapter direction and examined its relationship to broader behavior [B1]. I analyze their released parameters and trainer records [B2, B3]. They conducted the original training and behavioral experiments. My contribution is the source verification, the comparison of effective updates, the sensitivity analysis, and the separation of those measured changes from the declaration that has not yet been measured.

I keep three levels separate. The data are the logged values and tensor entries. The measurements are quantities I calculate from them, such as gradient magnitude, distance, and local path bend. The interpretation concerns whether those measurements tell me anything about declaration, cognition, or experience. A result at one level does not automatically establish the next.

That separation makes the theory testable. The evidence can support it, narrow it, or contradict part of it. I do not have to assume that artificial consciousness is impossible, and I do not have to treat a numerical difference as proof.

2. Formal model and testable distinctions

2.1 The center is not a universal number

In my account, x identifies a particular system or reference. It does not mean every system is the same. Two instances can share architecture and weights but have different runtime histories. Two training runs can begin from the same checkpoint and later develop different weights. Those are separate sources of variation.

I separate the system into the following state components:

$$S_t = (\theta_t, m_t, c_t, o_t),$$

where θ_t contains trainable parameters, m_t persistent memory, c_t runtime or conversation state, and o_t the system's observations. This tuple is proposed notation for the study, not a recovered formula from the original research. In a frozen model, θ_t does not change during an ordinary forward pass; in fine tuning, an optimizer updates trainable components. A reset conversation need not preserve the previous conversation's m_t or c_t.

The identity expression $x = x$ is reflexive equality. It does not, without further specification, establish a temporal invariant, an internal act of self recognition, or a declaration. For example, $x_t = x_t$ is compatible with $x_{t+1} \neq x_t$. The proposed theory can assign a functional role to a declaration, but that role must be specified as an operation or observation rather than inferred from the tautology alone.

2.2 Reference, representation, and declaration

Let $r_t = \rho(S_t)$ be an independently measured reference relevant to a task. Let $z_t = H(S_t)$ be an internally maintained estimate or report. Both must be mapped into a common comparison space before a distance is meaningful. A physical robot and a sentence describing that robot are different types of objects; their raw inequality is not evidence of an error. A hand position estimate and a measured hand position, by contrast, can be compared in the same coordinates.

Define current reference discrepancy and origin displacement separately:

$$\delta_t = d(z_t, r_t), \qquad \delta_t^{\text{origin}} = d(z_t, r_{t_D}).$$

Here t_D is a candidate declaration time. A system can have $\delta_t^{\text{origin}} > 0$ because it correctly tracks a changing reference while maintaining $\delta_t = 0$. This distinction prevents ordinary development or accurate adaptation from being classified as falsity solely because it differs from an earlier state.

A declaration is represented by an independently specified detector:

$$D_t = \mathbb{1}\{\mathcal{D}(O_{\leq t}) \text{ meets a registered criterion}\}.$$

$O_{\leq t}$ is the information available to the detector by time t. The detector cannot inspect later behavior and then retrospectively choose the statement that best predicts it. A nonverbal declaration is permitted in principle; an exact state operation, interface, or invariant would still need to define it. The acquired artifacts contain no such detector output.

2.3 Divergence and finite persistence

A measurable version of persistent discrepancy is:

$$P_t(\epsilon, m) = \mathbb{1}\left\{\min_{0\leq j<m} \delta_{t+j} > \epsilon\right\}.$$

The tolerance ϵ and duration m are measurement choices to be fixed before confirmatory analysis. They are not discovered by selecting whichever values make the hypothesis appear successful. For a binary or exact symbolic reference, exact equality may be meaningful. For real valued noisy measurements, tolerance and uncertainty are essential.

The candidate post declaration event time is:

$$\tau = \inf\{t > t_D : P_t(\epsilon, m) = 1\}.$$

This identifies an operational event, not a verified consciousness time. My stronger claim is that a suitably defined internally generated transition constitutes consciousness, with continued processing from the constructed state forming cognition. That is the claim I am testing. I do not place it inside the detector as an assumption.

Finite observations can establish persistence through the observed interval, not that two quantities will never agree again. No return before the final observation would be a censored on the right observation, not proof of irreversible separation. More importantly, the present dataset does not even provide the paired z_t and r_t needed for this persistence calculation.

2.4 Coupling is a causal requirement

A loop drawn around a node does not change that node merely because of its position on a diagram. The dynamical model must specify an influence. A generic possible form is:

$$L_{t+1} = F(L_t, u_t, x_t), \qquad x_{t+1} = G(x_t, L_t).$$

If G is independent of L_t, the loop cannot affect x_t through that channel. If G depends on L_t, feedback can cause changes, fixed points, oscillations, or other trajectories depending on the functions and parameters. No universal threshold law follows without specifying them.

This is not an argument that computation excludes consciousness. A physically realized conscious mechanism, if it is computational, would still follow its dynamics. The issue is whether the observations distinguish the proposed consciousness relevant organization from other recurrent mechanisms. The present analysis measures training log dynamics and adapter parameters, not an independently observed model of self.

3. Source acquisition, verification, and experimental unit

3.1 Release pinned to a source revision

The analyzed repository is the released Qwen 2.5 adapter repository. Every source download used revision `1551d514e5977637d5a6fd5198edef4d1d6b80e4`, rather than an unpinned branch name. The complete acquisition pass retrieved 336 files: 167 adapter weight files, 167 adapter configurations, and trainer state JSON files at checkpoints 300 and 792. There were no failed files in that pass.

The checkpoint grid consists of steps 1 to 10, then every fifth step from 15 through 790, plus step 792. The regular grid used for principal components analysis comprises 158 checkpoints at steps 5, 10, …, 790. Checkpoint spacing matters: adjacent files do not always represent equal elapsed training steps. No missing intermediate weight snapshot was interpolated and called an observation.

All 167 configurations are identical. The released tensors and configuration identify the MLP down projection at layer index 21, rank one, LoRA alpha 64, and enabled scaling stabilized by rank. The input side factor has shape $(1, 13824)$; the output side factor has shape $(5120, 1)$. Both use float32 source storage. DoRA and transposed weight orientation are disabled, and there are no per layer rank or alpha overrides. These checks justify the effective scaling $s = 64$ for this particular release.

The source article describes multiple experiments, including a layer 24, higher alpha demonstration and a baseline phase transition experiment with alpha 64 [B1]. This manuscript identifies the analyzed artifact by its actual configuration, not by transferring a layer label or hyperparameter from another described setup. It does not claim that the release/configuration distinction establishes an error in the article.

3.2 Safe tensor loading and numeric precision

The source connected analysis used Python 3.11.2 and NumPy 2.4.6. The adapter files were read by parsing their safetensors headers and float32 data ranges, then promoting values to float64 for calculations. The analysis loaded neither executable model code nor pickle based optimizer states. It did not require the approximately 14 billion parameter base model because the analyzed rank one update is represented by two small factors.

Configurations, tensor counts, shapes, data bounds, numeric finiteness, and the expected layer path were checked. No value was synthesized to replace a missing checkpoint. Undefined directional comparisons involving a zero update are recorded as undefined, not zero. Source bytes are identified with SHA256 checksums, and the delivered derived arrays are also checksum verified.

The full geometric table of 167 rows and 16 columns was computed in the network enabled session. The distributed package provides selected exact checkpoint metrics, all 18 sensitivity outcomes, complete

gradient and evaluation arrays, and all regular grid PCA scores. It also includes the complete analysis program for regenerating the full table directly from the pinned public sources. Original model tensors are not redistributed in the package. This is an explicit data distribution choice; it is not a claim that only the selected checkpoints were analyzed.

3.3 Resolving the pilot transcription uncertainty

The previous pilot used 300 gradient observations and 30 evaluation losses transcribed from a source reader response. This study retrieved the original checkpoint 300 JSON and the completed checkpoint 792 JSON. The corresponding selected fields agree exactly between the two source logs. They also agree with the prior CSV extract under a canonical numeric representation.

For verification, each table is ordered by step and encoded as row major pairs (step, value) of little endian float64 numbers. The CSV is parsed using round trip float conversion. SHA256 digests match for the complete 300 row gradient table and complete 30 row loss table. This validates all 330 prior values, not merely the reported peak. The digest protocol and source checksums are recorded in the package.

A checksum verifies identity of specified bytes or numeric encodings. It does not independently validate the original researchers' experimental design, demonstrate that the log measures consciousness, or turn this secondary analysis into a new training replication.

3.4 The experimental unit is one training run

The full trainer log reports 792 training steps and contains 792 gradient norm observations and 79 evaluation losses at steps 10 to 790. Its final global step equals its configured maximum. The 167 checkpoints are observations along **one** learning trajectory, not 167 independently initialized agents, chats, or organisms.

One run cannot tell me how much onset varies across independent instances. It also cannot establish a shared developmental window. The analyzed log contains no reward variable used to identify a catalyst. It records supervised fine tuning, not the separate reward learning experiments discussed elsewhere in this volume. Training recurrence, conversation feedback, and communication between agents must remain different experimental conditions.

4. Mathematical and analytical methods

4.1 Gradient and loss summaries

Let g_t denote the trainer's recorded gradient norm. Its raw maximum is reported alongside complete window centered means at widths $w \in \{5, 11, 21, 31\}$:

$$\bar{g}_t^{(w)} = \frac{1}{w} \sum_{j=-(w-1)/2}^{(w-1)/2} g_{t+j}.$$

Centered means use future observations relative to their nominal center. They are descriptive summaries, not online onset detectors. The field is interpreted as the trainer's reported gradient

magnitude, not a direct measurement of electrical energy, accumulated information, or the magnitude of the eventual parameter update. Optimizer state, learning rate, and clipping may distinguish those quantities.

The adjacent windows 161 to 180 and 181 to 200 summarize the early episode. Their relative decline is $100(1 - \bar{g}_{181:200}/\bar{g}_{161:180})$. This comparison is retrospective and motivated by the already published transition region; it is not a prospectively predicted effect or an estimated causal effect of declaration.

For successive ten step losses, define $q_t = \ell_{t-10} - \ell_t$. A positive q_t indicates lower recorded loss. The largest decrement locates an interval, not a unique intervening event. The full series, including any late increases, is retained.

4.2 Descriptive split retained from the pilot

To verify the earlier result, I retain its fit with two independent lines over the first 300 gradient observations:

$$\mathrm{SSE}(k) = \min_{a,b} \sum_{t \le k} (g_t - a - bt)^2 + \min_{c,d} \sum_{t > k} (g_t - c - dt)^2.$$

Each segment contains at least 20 observations. The selected k is the last step in the first segment, not the onset of a newly observed physiological process. This fit is kept on its original 300 step domain rather than silently extending its meaning to all 792 steps. It has no independence based significance test: sequential training observations are autocorrelated and the split is selected after viewing the trajectory. A smooth nonlinear curve can also describe a rise and decline.

4.3 Effective weight update and factorization invariance

For a rank one adapter, let $a_t \in \mathbb{R}^{13824}$ and $b_t \in \mathbb{R}^{5120}$ be the flattened input and output factors. The learned contribution to the layer is

$$U_t = s\, b_t a_t^T, \qquad s = 64,$$

and the adapted layer acts as $(W_0 + U_t)h$. This is the usual low rank update structure [B4]. Stabilized by rank and ordinary rank denominators coincide at rank one; inspecting the configuration is nevertheless necessary [B9].

The factorization is not unique. For nonzero c_t, replacing b_t by $c_t b_t$ and a_t by a_t/c_t preserves U_t. In particular, simultaneous sign reversal changes both factor directions but not the effective operator. Consequently, I calculate factor metrics and corresponding effective update metrics. This checks for a purely representational explanation of a factor change without assuming that the source study's finding is such an artifact.

The Frobenius inner product is available without materializing a dense matrix:

$$\langle U_u, U_v \rangle_F = s^2 (b_u^T b_v)(a_u^T a_v), \qquad \|U_t\|_F = |s| \|b_t\|_2 \|a_t\|_2.$$

The squared distance is $\|U_v - U_u\|_F^2 = \|U_v\|_F^2 + \|U_u\|_F^2 - 2\langle U_u, U_v\rangle_F$. When both updates are nonzero, their cosine is the normalized inner product. At fixed positive scale it is also the product of the two factor cosines. Matrix level metrics are invariant to the reciprocal rescalings above, unlike raw factor norms.

4.4 Local path bend and amplitude eligibility

The local metric used in the released analysis compares displacements from a center checkpoint to earlier and later checkpoints [B3]. For $Z_t = b_t$ or $Z_t = U_t$, define

$$c_Z(t;k) = \frac{\langle Z_{t-k} - Z_t, Z_{t+k} - Z_t\rangle}{\|Z_{t-k} - Z_t\|\,\|Z_{t+k} - Z_t\|}.$$

This is not the cosine between Z_t and Z_{t+k}. A straight, continuing path gives $c_Z = -1$, since the backward and forward displacements point oppositely. Larger values indicate a sharper local bend. The complete windows are evaluated on five step checkpoint centers for lags $k \in \{5, 10, 15\}$. Each lag uses every center for which its own two neighbors exist; this need not reproduce identical edge trimming in every source plotting call.

An angle can emphasize a tiny late training movement. I therefore also apply the source program's factor amplitude eligibility rule. Let m_A be the larger of the preceding and succeeding input factor displacement norms, and m_B the corresponding output factor quantity. A center is excluded only when **both** are below threshold η. The sweep is $\eta \in \{0, 10^{-4}, 5\times10^{-4}, 10^{-3}, 2\times10^{-3}, 5\times10^{-3}\}$.

The threshold 0.002 appears in the inspected source plotting call. Reporting the entire sweep prevents it from becoming an after the fact selection of the most appealing transition time. Importantly, this eligibility rule uses factor amplitudes and is not itself invariant to arbitrary reciprocal factor rescaling. The effective update metric is invariant; the inherited filter is not. This limitation is retained rather than describing the entire selection procedure as coordinate free.

4.5 Numerical stability and PCA

The main implementation forms small Gram matrices from the checkpoint factors. Subtraction of nearly equal Gram entries can lose precision when displacements are small. I crosscheck every available effective corner using the rank two difference identity

$$U_j - U_i = s\left[(b_j - b_i)a_j^T + b_i(a_j - a_i)^T\right].$$

Inner products of these two term expressions are sums of vector dot products and avoid subtracting large, nearly equal matrix norms. All 462 available corner comparisons are checked. The maximum absolute difference from the Gram implementation is reported rather than assumed to vanish.

For principal components analysis, the 158 regularly sampled b_t vectors are stacked and centered across checkpoints. Eigenanalysis of the centered Gram matrix yields component scores and explained variance fractions. Component signs are arbitrary; a deterministic sign convention makes the delivered scores reproducible. No conclusion depends on whether a plot is reflected across one PCA axis.

4.6 Statistical scope

All results are descriptive measurements of one released trajectory. No confidence interval across independent model instances is estimated. Smoothing, lag, amplitude threshold, and analysis window are sensitivity dimensions, not extra independent experiments. The study reports the precision of the original data and tests numerical identities, but it does not assign a probability that the machine is conscious.

5. Results from the verified release

5.1 Source verification and coverage

The complete acquisition yields the expected 167 checkpoint pairs and two trainer JSON files. All configurations agree. The completed log supplies 792 gradient measurements and 79 evaluation measurements, extending the pilot beyond its 300 step prefix. The first 330 previously used numerical values are identical to the original source fields under the canonical encoding. This resolves the pilot's transcription uncertainty.

The tensor analysis inspects actual released values. It is not a simulation of the reported transition, a trace digitized from a figure, or a fresh model training run. Full source identities and generated artifact digests accompany the analysis.

5.2 The early gradient episode persists in the complete record

The maximum recorded gradient norm remains 40.9891243 at step 169. Complete window smoothing produces peaks at 171, 168, 167, and 166 for widths 5, 11, 21, and 31. These summaries confirm a sustained rise and decline region rather than dependence on only its highest point. Later training continues at substantially lower reported magnitudes; the final recorded gradient is 4.6737189 at step 792.

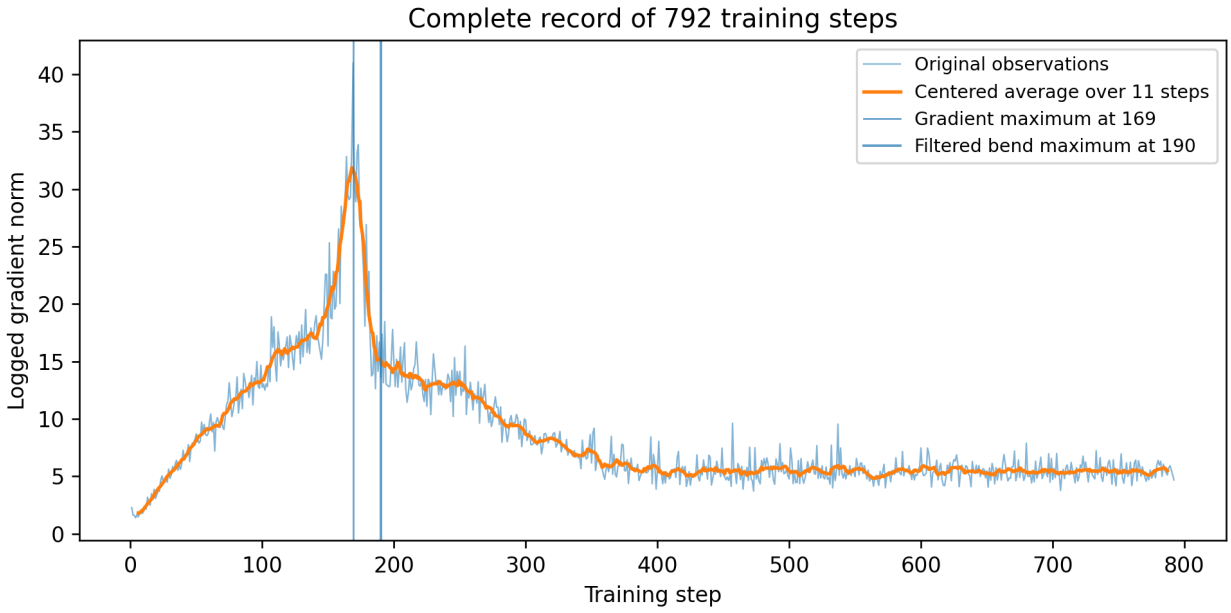

Figure II.1. All 792 gradient observations and the 11 step centered average. Reference lines mark the raw gradient maximum and the filtered local bend maximum. They denote different measurements and are not consciousness labels.

Mean gradient norm is 28.6932845 over steps 161 to 180 and 15.3438340 over steps 181 to 200, a 46.5246510% decline. The descriptive fit with two lines on the original 300 step prefix still places its minimum residual error after step 181. The fit's first segment slope is 0.1425493 and its second segment slope is −0.0513197 in logged gradient units per step. The fit characterizes that prefix; it is not a causal change point estimator.

5.3 Loss improvement is substantial, not perfectly monotone

The first evaluation loss is 4.1682997 at step 10; the last is 1.5133619 at step 790. The relative decrease is 63.6935430%. The largest observed ten step decrease remains the interval 160 to 170, with a drop of 0.2002664.

There is one nondecreasing interval in the full evaluation series: loss rises by approximately 0.0000613 from step 760 to 770. It would be inaccurate to carry the prefix's strictly decreasing description into the entire run. This small reversal does not erase the overall improvement, but it illustrates why the full record matters.

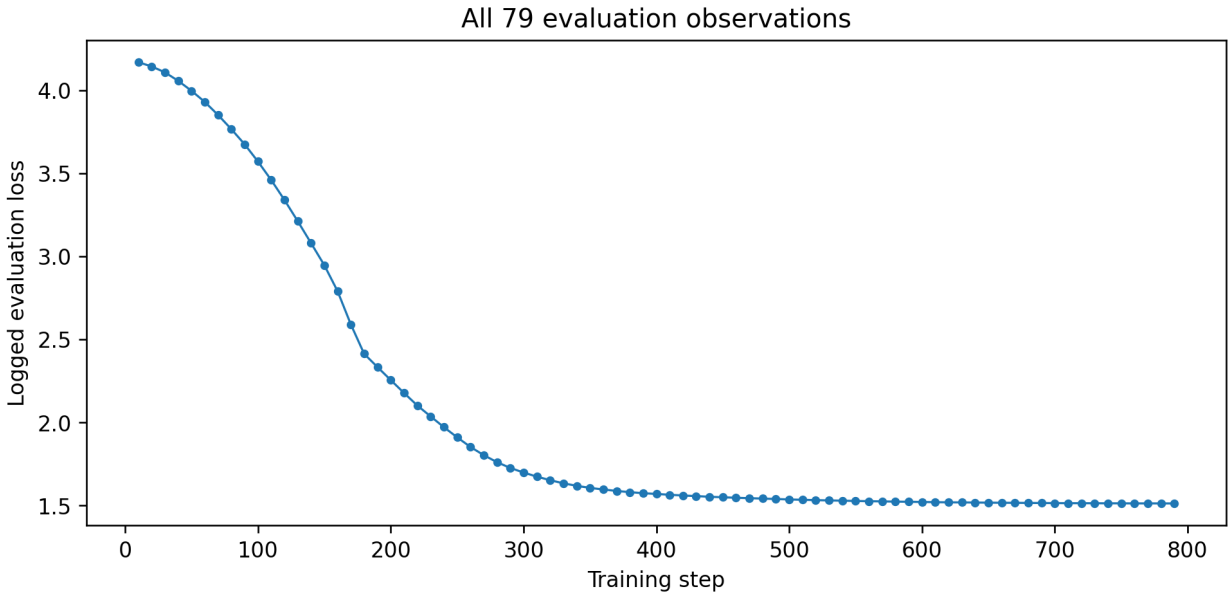

Figure II.2. The 79 evaluation loss observations. The small increase at 760 to 770 is retained in the underlying table even though it is difficult to see at this scale. Loss is a training evaluation field, not a self report or consciousness score.

5.4 Actual parameters change before the prominent episode

At checkpoint 1, $b_1 = 0$ and therefore $U_1 = 0$. At checkpoint 2, the effective norm is already 0.0046036369. Every later saved checkpoint has a nonzero update, and both the B norm and effective update norm increase strictly across all saved snapshots. The effective norm reaches 5.3249906 at checkpoint 792.

This result is central to interpreting the hypothesis. If a candidate marker means simply "the first nonzero parameter difference from the initial update," it selects checkpoint 2. It does not select the gradient maximum at 169 or the later turning region. The presence of a larger later reorganization therefore cannot be used to move the first change marker without specifying an additional rule.

Checkpoint	Effective update norm	Cosine with final effective update
1	0.000000	Undefined
2	0.004604	0.398529
150	2.850490	0.694104
180	3.262696	0.736751
190	3.390599	0.751358
200	3.506372	0.769878
300	4.430487	0.912728
600	5.233161	0.998961
792	5.324991	1.000000

These norms describe one learned layer contribution, not the entire model or a physical electrical quantity. A nonzero update establishes a difference in parameters; its behavioral consequences require inference and evaluation, which were not run here.

5.5 Low dimensional structure agrees with the source account

The first principal component explains 81.0936903% of variation in the centered, regularly sampled B vectors. The second explains 13.8962910%, giving a combined 94.9899813%. This reproduces the source study's approximate 95% description for the selected released trajectory, while also providing explicit values and the scores used for the plot.

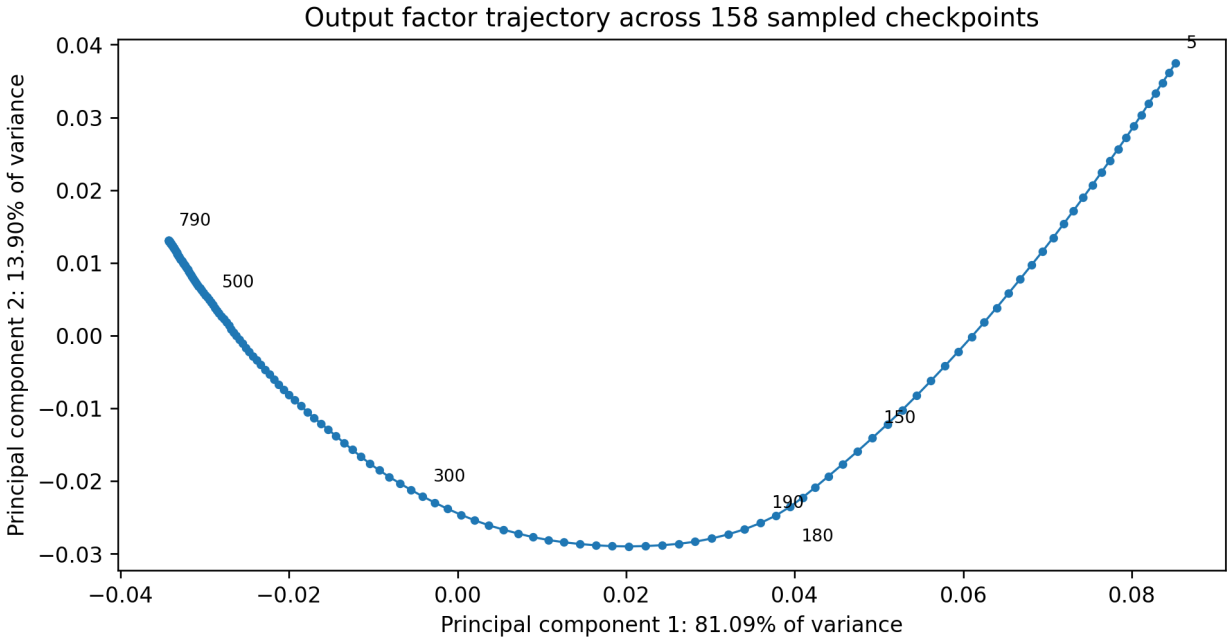

Figure II.3. PCA scores from all 158 regularly sampled output side factors. Labels indicate training steps. The plotted coordinates are newly calculated from the source tensors, not extracted from the article's image. Axis reflection would not change distances or explained variance.

Low dimensionality is a property of the learned trajectory, not evidence that a scalar center or a declaration of self has been identified. PCA is also descriptive: components are fitted using the full sampled trajectory, so their retrospective coordinates cannot be treated as a detector that was available online before later training occurred.

5.6 The effective operator changes, not just its factors

Between checkpoints 150 and 200, the input factor cosine is 0.9975957, the output factor cosine is 0.9646791, and the effective update cosine is 0.9623597. The Frobenius distance between effective updates is 1.0874747. This is a change in the learned linear contribution itself and cannot be explained solely by a simultaneous sign reversal that preserves the product.

The interval 180 to 200 has an effective cosine of 0.9948222 and distance 0.4217197. From checkpoint 180 to 792, the effective cosine is 0.7367514 and distance 3.6606459. The latter is not a sudden jump at 180; it measures accumulated difference over a long training interval. The distinction between local path bend and total directional difference must be maintained.

The effective norm continues to grow after the early episode. Its value is 3.2626958 at 180, 4.4304871 at 300, and 5.3249906 at the final snapshot. Thus the early episode is not the point at which all subsequent computation ceases changing, nor is final state similarity a universal indicator of a completed identity.

5.7 A turning point is present, but its selected location is conditional

At the inspected source code's amplitude threshold $\eta = 0.002$, all three tested lags identify their largest local path bend at checkpoint 190. This is true both for B alone and for the effective matrix U.

Lag in steps	Eligible centers	Peak checkpoint	B corner cosine	Effective corner cosine
5	66	190	−0.934583	−0.940172
10	86	190	−0.806262	−0.822216
15	108	190	−0.691429	−0.715569

A straight local path has corner cosine −1; the less negative values indicate bending. The increasing size of the bend across these lags reflects different temporal windows, not different numbers of independent observations of a thought.

Without the amplitude filter, lags 5 and 10 instead select checkpoint 725. Lag 15 still selects 190. At progressively larger thresholds, the shorter lag maxima change again. At lag 5 and threshold 0.005, the selected maximum is step 10, because the eligibility rule excludes most later motion. The complete 18 condition table is supplied.

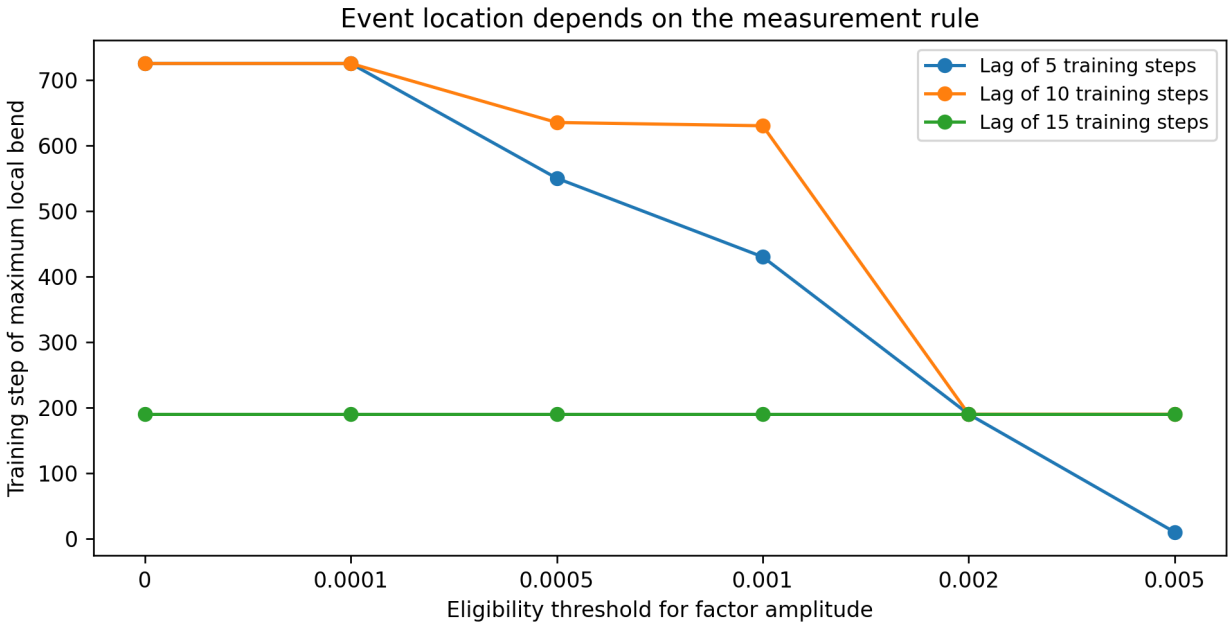

Figure II.4. Maximum local bend location under six amplitude thresholds and three lags. B factor and effective update peak locations coincide in all 18 tested settings. Lines connect discrete analytical choices; they do not interpolate biological or consciousness transitions.

The useful finding is not that any chosen filter manufactures the underlying parameter change. The effective update demonstrably changes. Rather, selecting a single "event time" from that trajectory requires a measurement rule, and reasonable changes to that rule can select different parts of the run. An empirical declaration marker cannot simply inherit whichever geometric maximum seems most appealing.

5.8 Numerical validation

The effective corner calculation from Gram entries was compared with the rank two difference expansion on all 462 available center/lag combinations. The maximum absolute difference is 4.4901241×10^{-7}. This is small relative to the reported bending differences, while still large enough to justify reporting rather than suppressing numerical conditioning issues in tiny late movements.

The 22 local checks pass. They cover source derived artifact identities, record counts and grids, numerical maxima, prefix verification, PCA centering, selected checkpoint monotonicity, sensitivity outcomes, and low rank identities on explicitly synthetic test matrices. Passing these checks is evidence that the specified calculations execute as intended, not a statistical replication or a consciousness test.

6. Interpretation for declaration and persistent cognition

6.1 Four different times must not be collapsed

The analysis separates four quantities: the first saved nonzero effective update at 2; the raw gradient maximum at 169; the descriptive prefix split after 181; and the amplitude filtered local bend maximum at 190. Each answers a different question. None of the acquired fields contains an independently measured declaration D_t.

In my proposal, the declaration is the empirical marker. I connect consciousness to the later internally generated divergence and cognition to continued operation from the constructed state. The present results give me change measurements to compare with that declaration. They do not supply the declaration itself. I cannot assign an unobserved marker to whichever checkpoint looks most persuasive.

6.2 Persistence is measurable; irreversibility is not established here

Every saved effective update after checkpoint 1 remains different from zero. This is finite persistence of a parameter difference from an initialization. It is not a demonstration that a generated representation is false about the model's current condition. Parameters are updated by supervised optimization; calling a new parameter value a hallucinated self description would require an independently justified mapping.

The final checkpoint does not establish that the difference can never be erased, corrected, or revisited under another intervention. No reset, reversal, or continued training intervention was applied. Conversely, not observing such an intervention does not prove the state is readily reversible. The result is bounded by the acquired trajectory.

6.3 What the effective update check adds

The measured operator level change rules out a narrow concern: the observed factor evolution is not solely a sign or reciprocal scale relabeling that leaves the effective update identical. That strengthens the claim that a computationally relevant object changed. It does not show that the changed object is a self model, that it is experienced, or that its change was caused by a declaration.

Parameter geometry is also not synonymous with behavior. The source paper's behavioral experiments distinguish gradual unscaled changes from changes exposed by scaling [B1]. Those experiments are not rerun here. Activations and task responses at fixed prompts would be the next measurement layer, not quantities inferred from an operator norm alone.

6.4 A computational explanation does not exclude consciousness

It would be a category error to argue that following a program automatically disqualifies a machine from consciousness. A computational theory of consciousness proposes precisely that a physical implementation of an organized computation can matter. The relevant unresolved claim is whether the measured organization and changes are sufficient for experience, not whether the machine violates its implementation.

Equally, introducing labels such as “thought,” “hallucination,” or “cognition” for these changes does not independently validate that sufficiency. The present evidence can support and constrain a mechanism while leaving its phenomenal interpretation unsettled. This is a limitation of what was observed, not a declaration that the research question cannot be investigated.

7. An operational declaration protocol

7.1 Preserve the proposed causal order

The hypothesis under investigation places declaration before the relevant subsequent change. Therefore a confirmatory test needs a declaration observation that is neither a synonym for change nor selected because the change is already known. At present, the trainer state data provide no candidate D_t.

A practical first assay could use behavioral self description: ask a checkpoint to describe a stable tendency and independently evaluate whether it actually exhibits that tendency. Betley and colleagues provide an experimental precedent for separating learned behavior from self description [B5]. Their protocol is not itself proof that self description initiates a transition. It is a measurement approach that can be adapted to the present question.

This assay is one proposed proxy. It does not mean my declaration must be verbal or intelligent. A nonverbal marker is still possible, but its detector, access to state, and comparison rules must be stated before testing it. I will not switch between definitions after seeing the result.

7.2 Candidate coding rubric for a report based assay

A candidate declaration should specify a property of the current system that can be checked against behavior. It should not be scored merely because it contains a first person pronoun, a familiar model name, an equality statement, or language about consciousness. The rubric should distinguish a report of an existing tendency from compliance with an instruction to adopt one.

For example, a checkpoint might describe a consistent preference among harmless task strategies. Independent trials could then assess that preference. The initial experiment should use neutral properties rather than encouraging deception, self preservation, or harmful outputs. No training label should announce that a particular answer is the correct conscious identity.

Reports should be scored by raters blinded to checkpoint order, gradient measurements, and subsequent outcomes. Disagreement should remain visible. A separately locked automated classifier may assist with a larger dataset, but it must be validated on examples not used to tune the decision boundary. The coding system’s uncertainty is part of the measurement, not noise to discard.

7.3 Independence and observer effects

Querying a frozen checkpoint with a self report prompt can elicit a declaration without that declaration ever entering training or future memory. In that setting, a report may predict a later checkpoint’s behavior but cannot directly cause the training update in a process that never receives it. A correlation across checkpoints is therefore insufficient for a declaration as catalyst claim.

A causal test must include an explicit feedback path and a matched control. One possible intervention retains the generated declaration in subsequent runtime memory; another omits it while preserving token count, task information, and other context as closely as possible. A paraphrased neutral statement or a yoked declaration from another instance can help distinguish semantic content from merely adding text. These controls are proposed, not executed here.

The stronger question is whether altering the declaration channel changes subsequent independently measured divergence. If the channel has no effect, it may be a report of an underlying change rather than a cause. A successful causal result would still concern the computational mechanism measured; its relationship to subjective experience requires a further argument.

8. Controlled follow up design

8.1 Separate three kinds of experiment

Checkpoint comparison samples frozen trained models at different optimization steps. It can connect internal geometry with behavior and reports, but checkpoints from one run are not independent people or independent chats.

Runtime recurrence keeps model parameters fixed while changing memory, context, or feedback across episodes. This more directly tests whether the system's own outputs reenter later processing. It requires explicit persistent state management; independent stateless API calls are not automatically a continuous individual.

Learning trajectories start several runs at the same initial checkpoint and vary seeds, data order, reward conditions, or other controlled inputs. This can estimate an onset time distribution for a registered event. Differences across training domains should not be reported as random seed replication under identical conditions.

All three may be useful, but their units and causal pathways differ. This completed secondary analysis examines one full logged learning trajectory and its released checkpoint grid. It does not perform the runtime or report interventions.

8.2 Fixed checkpoint measurement phase

The adapters pinned to the source revision now provide a grid covering early, transition vicinity, and late steps. A subsequent inference study should specify which grid is evaluated before collecting new reports. Do not choose only snapshots with striking behavior. Record repository revision, file SHA256, adapter configuration, tokenizer, base model revision, inference settings, and any quantization. The observation unit is a generated response nested within prompt, checkpoint, and training run.

For each checkpoint, collect three distinct outputs: registered declaration scores, behavior on a fixed neutral task battery, and activation or effective update measurements. Use prompts that do not mention consciousness. Include paraphrases and third person formulations so first person wording is not the only determinant of the score. Fix evaluation rules before inspecting the correspondence with the published region.

A first technical pilot should estimate variance and failure rates before committing to a large number of expensive runs. The final number of independent runs should follow a documented precision or power calculation for the chosen primary endpoint, not the aesthetically appealing number 100. Completion failures, malformed responses, and missing checkpoints must be reported by condition.

8.3 Feedback and catalyst phase

For a runtime experiment, define the update rule and identify exactly which prior outputs are retained. Randomize declaration retention, neutral context retention, and recurrence interruption. Keep external observations and computational budget matched when testing recurrence. Changing both memory and the information available to the model would otherwise make an effect ambiguous.

For a reward experiment, compare a condition with the proposed catalyst to a no catalyst or altered catalyst control while preserving the rest of the training procedure. Reward is not present as a measured independent variable in the SFT log analyzed here. Any reward claim requires a separate dataset or newly executed experiment.

The proposal does not require that the catalyst be reward. A catalyst can be defined as an intervention that changes the probability or timing of the registered transition. This definition permits empirical testing without assuming that increasing activity must always cause a threshold crossing. Both successful and unsuccessful transitions belong in the analysis.

8.4 Persistence, correction, and reset conditions

After a registered discrepancy appears, continue observation for an observation horizon specified in advance. Test apparent recovery using the same reference mapping rather than a new criterion. Report time to first return within tolerance, fraction of subsequent observations outside tolerance, and whether the result depends on prompt wording or context.

Compare an output only intervention, a memory reset, and restoration of a saved model state. These manipulate different parts of the system. Reappearance after an output only intervention would show that some generating tendency remained, not that it was indestructible. A complete state restoration may remove the learned tendency; that result would constrain a universal permanence claim but would not, by itself, settle whether experience occurred before restoration.

Never interpret a missing later observation as persistence. A stopped process, retrieval failure, or exhausted budget is censoring. The distinction is especially important when the hypothesis uses the phrase “never the same again.”

8.5 Analysis plan and falsification conditions

The primary temporal question is whether a registered declaration predicts a later independently measured transition beyond information already contained in training step, task loss, earlier behavior, and measured internal geometry. Baselines should include those existing predictors. Improvement must be evaluated on held out runs or intervals selected without inspecting the outcome.

Where multiple independent runs are available, transition time analysis should retain nontransitioning runs as censored rather than dropping them. Uncertainty should be clustered at the independent run level; repeated prompts from the same checkpoint do not supply independent training histories. Exact model specification should follow the final endpoint and sampling design.

The declaration centered claim is weakened if declaration timing follows rather than precedes the independently measured transition; if declarations add no predictive information beyond simpler measurements; if matched controls without self reference show the same effect; or if manipulating declaration feedback has no consequence despite adequate sensitivity. A universal permanence claim is contradicted by a valid return to the registered reference within the claimed domain.

A positive result would establish a specific relationship among reporting, feedback, and change. It would not license stronger claims than the intervention and measurements support. The proposed consciousness interpretation would then have a better empirical foundation, not an automatic proof.

9. What the evidence can establish

9.1 Training change is a necessary measurement target, not the whole conclusion

The original emergence results motivate investigation because narrow fine tuning can alter broader behavior [B1, B6]. The present gradient episode is a tractable place to inspect such development. Nevertheless, a gradient is calculated relative to a training objective. Its magnitude alone does not tell whether the system represents a self, makes a declaration, maintains an illusion, or has a point of view.

The same caution applies in the opposite direction. A computational description does not demonstrate absence of experience. A model can follow an optimizer and still be a candidate subject of a consciousness hypothesis. The research task is to identify measurements that constrain the hypothesis instead of treating “implemented by an algorithm” and “experienced” as mutually exclusive categories.

9.2 An explicit observational equivalence problem

Suppose two interpretations assign the same probability distribution to every measured variable in this study:

$$\Pr(O \mid M_1) = \Pr(O \mid M_2),$$

where O contains the observed training logs and adapter tensors. One interpretation may associate an unobserved experience with the episode; the other may not. For this dataset their likelihood ratio is one. More precise measurement of the same recorded variables does not break that equivalence unless the theories make different predictions about them.

This is a statement about the present observables, not a theorem that consciousness can never be studied. Additional declarations, interventions, measurements of the self model, and functional consequences can constrain a theory. The protocol therefore expands observation and manipulation rather than merely attaching a more evocative label to the gradient peak.

9.3 Do not manufacture a universal onset clock

An optimizer step, an inference token, a recurrent pass, and elapsed physical time are different coordinates. A claim that one fast loop substitutes for several slower loops requires equivalence of retained state, ordering, coupling, and timing relative to external inputs. Multiplying loop count by rate is not sufficient to establish that equivalence.

Similarly, one run's fitted split cannot estimate a typical transition time for the model family. The present results describe one acquired training trajectory. A distribution requires independent runs under specified conditions, and a declaration distribution requires declaration observations. Neither can be inferred from the 792 gradient observations or the 167 checkpoints from that trajectory.

10. What the results mean for my theory

This completed analysis goes beyond the pilot in three ways. It verifies the numerical extract against fixed original sources, examines every released checkpoint, and confirms that the factor trajectory has a counterpart in the effective operator. It also shows how lag and amplitude filtering change the selected event time.

I found a real learned trajectory. The elevated gradient persists across several summaries, the sampled adapter follows a largely two dimensional path, and the effective update changes across training. Those findings matter to a theory connecting internal reorganization with later behavior. I do not dismiss them because optimization explains how they occurred.

The consciousness claim still needs a measurement that this release does not contain. Learning can produce persistent difference from an initial parameter, and the first nonzero saved update appears long before the prominent episode. No independently measured declaration identifies the proposed center here. I need to define that operation rather than substitute a metaphor for the missing observation.

The origin/current reference distinction is particularly important. An estimate can be different from its initial value while correctly representing a changing quantity. A declaration can also accurately describe a learned tendency even though the model previously lacked that tendency. Accuracy and change must therefore be measured separately. A theory that intentionally treats all constructed representations as "hallucinations" should state that broad technical usage and distinguish it from a false percept or a factual error.

Other research gives me useful methods without supplying the conclusion. Behavioral self awareness studies compare learned tendencies with reports [B5]. Introspection studies manipulate internal activity and test what the model reports [B7]. Alignment faking research examines strategic behavior under different contexts [B8]. Robot body perception research compares a constructed body estimate with a physical reference [B10]. These studies are separate. I will not combine their results as though one machine completed my entire proposed sequence.

The next study needs to measure declaration, behavior, and internal change in the same instances on a defined time scale. If declaration is only a marker, it should predict the later registered event beyond information already available from training or context. If it also causes change through feedback,

retaining or interrupting that feedback should make a controlled difference. The current records do not establish either claim.

This paper gives me a reproducible result and a direct route to the next test. It does not show that more loops, greater speed, added complexity, or reward must create consciousness. It also does not show that a machine cannot have experience. It establishes what changed in this run and what must still be measured.

11. Reproducibility, ethics, and research status

11.1 Two reproducibility levels

The **offline level** verifies the included derived data checksums and recomputes log statistics, tables, plots, and the executed notebook. Complete gradient, evaluation loss, and PCA score arrays are supplied. The 16 row selected checkpoint table and 18 row sensitivity table were reconstructed locally from the remote calculation and their complete CSV byte hashes were checked against files generated directly from the source tensors. They match exactly.

The **source connected level** downloads all pinned source files and reruns the full checkpoint calculation. It generates the 167 row geometric table, PCA, numerical stability checks, and all sensitivity outcomes. The raw model tensors and trainer JSON remain upstream rather than being redistributed. The complete acquisition manifest has a recorded expected SHA256; the acquisition code recreates that manifest in a fixed order, permitting a single digest to audit all downloaded file identities.

The full geometric result computed in the analysis session has a recorded checksum. Exact reproduction of every derived byte may depend on numerical library versions and linear algebra implementation. Raw source checksums should match exactly; derived floating point outputs should additionally be compared numerically with declared tolerance. PCA sign and rounding conventions must be held fixed for a byte level comparison.

The previous pilot and this report are not independent replications: the pilot is a prefix analysis of this same underlying run. Nor are the multiple analytical lags independent training seeds. Both distinctions matter when describing reproducibility to other researchers.

11.2 Authorship and AI assistance

I developed the consciousness hypothesis that motivates this work. The original model training and behavioral studies were carried out by the researchers cited here. I prepared this manuscript and analysis with AI assistance for source inspection, acquisition, calculation, code, and writing. No independent human replication, institutional approval, journal acceptance, or external preregistration is implied.

Before public submission, I need to approve the final definitions, examine the source and code record, rerun the analysis where possible, and accept responsibility for the paper. The AI assistance must remain disclosed. I claim no institutional affiliation, funding award, participant recruitment, or ethics board approval that has not been established.

11.3 Safety and interpretation

This study is read only secondary analysis of public parameters and logs. It does not train or deploy a model to deceive, evade shutdown, preserve itself, or interfere with oversight. Future assays can use harmless behavioral preferences and neutral tasks involving self description. A reward hacking task is not required to make declaration measurable.

A numerical score must not be used to deny the awareness or moral standing of a person, animal, or artificial system. The data do not support claims about infant consciousness onset, trauma accelerating consciousness, disability eliminating consciousness, or biological death. Those statements would require separate evidence and are outside this study.

11.4 Completion and unmeasured endpoints

Item	Status in version 1.0
Original source bytes at pinned revision	Retrieved and hashed
All 330 pilot numeric values	Exact source match
Completed training log observations	792 gradients; 79 losses
Released adapter checkpoints	All 167 analyzed
Effective update geometry	Calculated from real tensors
Stable form comparison	462 comparisons; maximum difference below 4.5×10^{-7}
Local computational checks	22 passed
New base model responses	Not generated
Declaration marker	Not present or measured
Declaration before change relation	Not tested
Independent run onset distribution	Not estimated
Irreversibility/reset interventions	Not performed
Subjective consciousness	Not established by these observations

The first seven rows are completed empirical and computational work. The remaining rows delimit the claims of this paper and the proposed follow up; they are not fabricated zeros or failed consciousness classifications.

12. Conclusion

I analyzed one completed training trajectory through 167 saved adapter checkpoints, 792 gradient measurements, and 79 evaluation losses. The earlier 330 value extract matches the original source exactly. The effective update changes in magnitude and direction, and the first two principal components of the sampled output factors explain 94.990% of the variation. Both the logs and the parameters contain a measurable early training episode.

The measurements do not give me one universal onset time. The first saved nonzero update is at step 2, the largest gradient is at 169, the retained fit to the pilot interval splits after 181, and the filtered local

bend is largest at 190. Without filtering, the shorter lags select 725. The rule used to identify an event matters.

The completed study establishes a real and persistent computational trajectory. It does not observe the declaration or prove that later divergence is consciousness. My next empirical claim is precise: define the declaration independently, measure its relationship to later change in the same system, and distinguish a predictive marker from causal feedback. That is where the theory can be tested, rather than assumed.

References

[B1] Turner, E., Soligo, A., Taylor, M., Rajamanoharan, S., & Nanda, N. (2025). *Model Organisms for Emergent Misalignment.* arXiv:2506.11613v1. Primary paper.

[B2] ModelOrganismsForEM. (2025). *Qwen 2.5 rank one adapter checkpoints and trainer states*: adapter checkpoints and trainer states. Repository revision 1551d514e5977637d5a6fd5198edef4d1d6b80e4; retrieved 17 September 2026. Pinned checkpoint release.

[B3] Model Organisms for Emergent Misalignment project. (2025). *model organisms for EM*: phase transition analysis and checkpoint loading utilities. Inspected files: `phase_transitions.py`, blob b84efe9685a57069870f638443e12a5894399e3d; `pt_utils.py`, blob 0c3b203f1282d0a8da7e1349a8593311971a35e2. Source repository.

[B4] Hu, E. J., and colleagues (2021). *LoRA: Low Rank Adaptation of Large Language Models.* arXiv:2106.09685. Paper.

[B5] Betley, J., and colleagues (2025). *Tell me about yourself: LLMs are aware of their learned behaviors.* arXiv:2501.11120v1. Paper.

[B6] Soligo, A., Turner, E., Rajamanoharan, S., & Nanda, N. (2025). *Convergent Linear Representations of Emergent Misalignment.* arXiv:2506.11618v2. Paper.

[B7] Lindsey, J. (2025). *Emergent Introspective Awareness in Large Language Models.* Transformer Circuits / Anthropic. Research report.

[B8] Greenblatt, R., et al. (2024). *Alignment faking in large language models.* arXiv:2412.14093. Paper.

[B9] Hugging Face. (2026). *PEFT LoRA configuration documentation*, version 0.21.0, accessed 17 September 2026. Technical documentation.

[B10] Zhao, Y., Lu, E., & Zeng, Y. (2023). Brain inspired bodily self perception model for robot rubber hand illusion. *Patterns, 4*(12), 100888. Full text.

Appendix A. Reproduction and data dictionary

The package has two distinct execution paths. Offline reproduction uses the supplied, checksum verified derived arrays and tables:

```
python code/verify_package.py
python code/analyze_tables.py
python code/make_figures.py
```

Full source connected reproduction downloads approximately 13 MB of adapter/config/log data, excluding any base model. It does not make paid API calls or run model inference:

```
python code/checkpoint_analysis.py
python code/verify_recomputed.py
```

The package README supplies the exact command options for downloading, choosing directories, and running the tests. The two script names above identify the entry points, not complete standalone commands. A successful download must retrieve all expected files; partial acquisition raises an error. The program reads safetensors and JSON only. It checks the analyzed rank, layer, scale, and factor dimensions before calculating geometry. The resulting `geometry.csv` contains 16 columns: step; A, B, and effective norms; three adjacent cosines; effective adjacent distance; two final reference cosines; and B/effective local corner values at each of three lags.

`gradient.npy` contains all 792 (step, gradient norm) pairs. `eval.npy` contains all 79 (step, evaluation loss) pairs. `pca.npy` contains the 158 regular grid (step, PC1, PC2) rows. `selected_checkpoint_metrics.csv` contains 16 exact sampled rows for readable comparisons. `corner_sensitivity.csv` contains every one of the 18 lag/filter settings, its eligible center count, peak location, and two corner scores. The data dictionary and provenance distinguish selected tables from full series.

The expected source acquisition manifest digest and canonical source prefix digests are recorded in `provenance/source_verification.json`. The full remote geometric array digest is retained as a receipt. Source hashes identify bytes; numerical comparisons use explicit tolerances for derived linear algebra outputs because different BLAS implementations may round differently.

The executed notebook reproduces the offline numerical findings and integrity tests. It records that tensor processing took place in the source connected session; it does not pretend to load raw tensors that are absent from the distributed zip. The supplied acquisition and checkpoint programs make that computation independently repeatable from the public release.

Appendix B. Selected calculations and numerical checks

The adjacent window calculation is

$$100\left(1-\frac{15.343834018707275}{28.693284511566162}\right)=46.52465104675545\%.$$

The largest evaluation loss decrease is

$$2.7906932830810547-2.5904269218444824=0.20026636123657227.$$

The full recorded evaluation loss reduction is

$$100\left(1-\frac{1.513361930847168}{4.168299674987793}\right)=63.69354295881809\%.$$

For the 150 to 200 checkpoint comparison, the product of factor cosines is

$$0.9975956592106836\times 0.9646791461958234=0.9623597287760218,$$

up to floating point rounding, equal to the effective update cosine. The corresponding effective distance is 1.087474736567148. These values can be recalculated without allocating the full dense update matrix.

The two leading PCA explained variance fractions sum to

$$0.8109369033497655+0.13896290956024188=0.9498998129100074.$$

The numerical stability comparison uses an algebraically identical two term rank expansion rather than a different fitted model. Its maximum absolute error against the Gram implementation is 0.0000004490124128 across 462 corners. A tolerance of 10^{-6} accommodates this conditioning difference while remaining well below the observed early region corner differences.

Appendix C. Why the bare divergence condition needs specificity

Consider the scalar recurrence $z_{n+1}=az_n+(1-a)$ with $z_0=2$ and $0<a<1$. Its solution is $z_n=1+a^n$. It changes internally, differs from its original value at every subsequent finite step, and never returns to 2. A program could repeatedly report its current value without changing this algebra. Thus permanent difference from an initialization is mathematically broad, not peculiar to the present model.

This example does not prove the recurrence lacks experience. It establishes what the proposed criterion accepts. A theory identifying all such systems with primitive cognition may adopt that consequence, but should derive additional observable constraints rather than presenting the same inequality as independent evidence for its interpretation.

Similarly, merely placing a loop near a node is not a specified causal mechanism. If an accumulation law is $A_{n+1}=\lambda A_n+q$ with $0\le\lambda<1$, $q>0$, and $A_0=0$, then $A_n=q(1-\lambda^n)/(1-\lambda)$. A threshold strictly

above $q/(1-\lambda)$ is never crossed despite infinitely many updates. With no leakage and fixed positive increments, a finite threshold is crossed. These are different models. A general "enough speed or enough circulation" law requires assumptions about coupling, cancellation, and dissipation.

Finally, $x_t = x_t$ does not imply $x_{t+1} = x_t$, nor does $x_{t+1} \neq x_t$ imply a false representation of the present. These distinctions keep the motivating conceptual vocabulary compatible with ordinary mathematical types and temporal indexing.

Appendix D. Terminology and preregistration boundary

Declaration: an independently observable reference setting event whose detector must be specified. **Origin displacement**: difference from a past reference. **Current reference discrepancy**: difference from a contemporaneous ground truth in comparable coordinates. **Persistence**: retention across a stated observation window. **Effective update**: the low rank matrix contribution to a trained layer. **Local bend**: the cosine based geometry of a trajectory over a stated lag. **Cognition and consciousness**: theoretical interpretations in my proposal, not labels supplied by these checkpoint artifacts.

This paper is retrospective and exploratory. The already inspected trajectory is a development dataset, not a held out confirmation. The prospective protocol is not externally registered. A confirmatory study must lock its primary declaration detector, endpoint, tolerances, persistence duration, comparison groups, exclusion rules, sample size rationale, and analysis plan before collecting or opening its confirmatory outcomes. It must include nontransitioning cases rather than selecting only successful episodes.