A Machine-Checked Backbone for the Thermodynamic Attention Framework

Lean/Mathlib Identities, Zero Sorries, and the Errata the Prover Caught

companion (formalization) to Predicting How Transformers Attend*, Parts I–III*

Carles Marín*

Independent Researcher

transformerkmarin@gmail.com

July 2026

Abstract

The Thermodynamic Attention Framework (TAF) of Parts I–III rests on a dense web of closed-form algebraic identities: a Möbius transform relating the attention exponent γ to the RoPE config, a renormalisation-group β-function, a Hagedorn heat-capacity C_V, critical-exponent tautologies, and a Cayley fixed point. Dense formula tables are where sign and factor errors hide. We mechanise this backbone in Lean 4 / Mathlib: **52 theorems, zero `sorrys`**, each statement pinned to the paper section it supports. Formalization is used here not as decoration but as an *adversarial audit*, and it earned its keep: it caught three factor/sign errors in the published formula tables — the V/β potential (off by a factor of 2), the C_V coefficient at the Hagedorn point ($\frac{1}{4} \to \frac{1}{12}$, a factor of 3), and the critical exponent identity $\eta = 2\gamma$ (residual non-zero throughout Phase A) — and it distinguished a genuine *prediction* from a *tautology* by proving that the "information horizon" $d_{\text{horizon}} = T$ is an identity of the Padé formula, not an empirical coincidence. It also yields two positive results: the γ-Möbius collapse $\gamma = (1-u)/(1+u)$ with $u = T/(\theta\sqrt{2})$ (the $\sqrt{2}$ is load-bearing), and an arithmetic–geometric-mean bound $\gamma_\chi \geq 2\sqrt{2}$. All errata were incorporated into the corrected Parts I–II; this paper is the honest-revision ledger, machine-checked line by line.

Repository: https://github.com/karlesmarin/lean-taf (Lean toolchain `leanprover/lean4:v4.30.0-rc2`; Mathlib pinned; `lake build` re-verifies the whole library).

1 Introduction

Why formalize a physics-flavoured ML paper. Parts I–III derive their practical tools — a KV-cache compression window, an NTK scaling factor, a family classifier — from a chain of closed-form identities. The empirical claims are validated on GPUs; the *algebra* is validated by hand, and hand-validated algebra in a table with dozens of exponents is exactly where a stray factor survives peer review. Part II already reported a Sage/Lean "machine-verified algebraic backbone" of 15 identities and, in the same spirit, seven withdrawn or demoted claims. This paper completes that programme: it expands the backbone to the full `lean-taf` corpus and, crucially, reports *what the prover disagreed with*.

Formalization as adversarial audit. We treat Lean not as a seal of approval but as a skeptic that refuses to accept a step it cannot derive. Three of our own published equations did not survive that skeptic; the corrected forms did. Reporting both — in the same register — is the point. A machine-checked erratum is stronger than a prose apology: it exhibits the exact hypotheses under which the old claim is false and the new one holds.

*Prepared with the assistance of an AI system (Claude, Anthropic).

Contributions.

F1. A machine-verified backbone for TAF: 52 theorems, 0 `sorry`s, each pinned to a paper section (§4). ✓

F2. Three errata the prover caught in the published formula tables — V/β (factor 2), C_V at Hagedorn ($\frac{1}{4} \to \frac{1}{12}$, factor 3), $\eta = 2\gamma$ (refuted in Phase A) — each with its corrected, re-verified form (§5). ★

F3. The γ-Möbius collapse and the proof that $d_{\text{horizon}} = T$ is a tautology of the Padé formula, not a prediction (§6). ★

2 The Framework in One Page

For self-containedness we recall the objects the theorems are *about*; their derivation and empirical validation live in Parts I–III and are not repeated here.

Object	Closed form	Source
Attention exponent	$A(d) \propto d^{-\gamma}$	Part I
Reduced variable	$u = T/(\theta\sqrt{2})$	Part I
Padé / Möbius γ	$\gamma = (2\theta - T\sqrt{2})/(2\theta + T\sqrt{2}) = (1-u)/(1+u)$	Part I
RG β-function	$\beta(\gamma) = -(1-\gamma^2)/2$	Part I §RG
Integrated potential	$V(\gamma) = \gamma/2 - \gamma^3/6$ (*corrected*)	§5
Hagedorn heat capacity	$C_V(\gamma{=}1, N) = (\log N)^2/12$ (*corrected*)	§5
Cayley fixed point	$z^\star = (\sqrt{17}-3)/2,\ \chi = (5+\sqrt{17})/4$	Part I
Information horizon	$d_{\text{horizon}} = \theta(1-\gamma)\sqrt{2}/(1+\gamma) = T$	§6

3 Mechanization

The library builds against a pinned Mathlib on Lean `v4.30.0-rc2`. Lean follows the "no news is good news" convention: a successful check prints nothing; any `sorry`, axiom, or type error surfaces immediately. The whole library re-verifies with `lake build` (≈ minutes after a Mathlib cache fetch); the newest module (`GammaMobius.lean`) elaborates in 8371 jobs.

Listing 1: Re-verify locally.

```
git clone --depth=1 https://github.com/karlesmarin/lean-taf
cd lean-taf
elan toolchain install $(cat lean-toolchain)    -- one-time
lake exe cache get                              -- fetch Mathlib oleans
lake build                                      -- verify the whole library
```

Every theorem carries a comment pinning it to a paper section; a machine-readable manifest (file, line, claim, tactic, source section) accompanies the repository. Proof tactics are elementary by design — `field_simp`, `ring`, `linear_combination`, `norm_num`, `div_eq_iff` — so a reader can audit each step without trusting a monolithic automation call.

4 The Backbone

Table 1 groups the 52 theorems by what they establish, each pinned to the paper section it supports.

5 Three Errata the Prover Caught

Each finding below is stated as the prover found it: the old claim is exhibited as *false* under explicit hypotheses, and the corrected claim as *true*. All three corrections were incorporated into the revised Parts I–II.

Group	#	Representative theorem (Lean name)
γ Möbius collapse	6	`gamma_mobius`, `d_horizon_eq_T`, `beta_in_u`
RG flow / β-function	6	`logistic_to_beta_algebraic`, `V_correct_matches_RG_beta`
Cayley fixed point + χ	5	`cayley_root`, `chi_root`, `cayley_floquet_stable`
Padé approximant identities	5	`pade_22_minus_pade_11`, `theta_eff_pade`
D-SAGE algebraic identities	10	`beta_chi_closure`, `nu_beta_id`, `fisher_residual`
Scaling-law identities	9	`logistic_beta`, `levy_index`, `rg_flow_tanh`
C_V coefficient erratum	5	`cv_hagedorn_correct_coefficient`, `cv_factor_three_decomposition`
Paper-I audit findings	3	`eta_2gamma_fails_phase_A`, `AM_GM_gamma_chi`
Misc identities	3	`rope_beta_form` ($\sqrt{3}/6 = 1/\sqrt{12}$), `cft_nu_identity`
Total	**52**	**0 sorrys**

Table 1: The machine-verified TAF backbone. Full names and line numbers are in the repository manifest. (The `lean-taf` repository also contains an unrelated effective-field-theory audit built with the same tactic library; it is not part of this paper or series.)

Finding 5.1 (The V/β potential is off by a factor of two)**.** The formula tables jointly asserted $V(\gamma) = \gamma - \gamma^3/3$, $\beta = -V'$, and $\beta(\gamma) = -(1-\gamma^2)/2$. These are mutually inconsistent for every $\gamma \notin \{-1, +1\}$: the residual is exactly a factor of 2.

Listing 2: `RGFlow.lean`: the inconsistency and its fix.

```
theorem V_derivative_ne_RG_beta (gamma : R) (hne1 : gamma != 1) (hne2 : gamma != -1) :
    -(deriv (fun x : R => x - x^3 / 3) gamma) != -(1 - gamma^2) / 2
-- corrected potential integrates to the stated beta:
theorem V_correct_matches_RG_beta (gamma : R) :
    -(deriv (fun x : R => x/2 - x^3/6) gamma) = -(1 - gamma^2) / 2
```

Correction. Restate $V(\gamma) = \gamma/2 - \gamma^3/6$ (the ϕ^4 kink potential rescaled by $\frac{1}{2}$), which integrates to the stated β. Adopted in Part I §RG.

Finding 5.2 (The Hagedorn heat capacity coefficient is $\frac{1}{12}$, not $\frac{1}{4}$)**.** At the Hagedorn point $\gamma = 1$, the earlier draft gave $C_V = (\log N)^2/4$. The cumulant derivation yields $(\log N)^2/12$; the original is exactly three times the correct value.

Listing 3: `ErratumCV.lean`: the factor of three, quantified.

```
theorem cv_hagedorn_correct_coefficient : ...   -- CV(1,N) = (log N)^2 / 12
theorem original_is_three_times_corrected (L : R) : ...   -- old = 3 * corrected
theorem original_ne_corrected (L : R) (hL : L != 0) : ...  -- old != corrected
```

Correction. $C_V(\gamma{=}1, N) = (\log N)^2/12$; for $N = 2000$, $C_V \approx 4.81$. Adopted in Part I §5.2. The integral derivation $\int (\log x)^2/x\, dx$ is itself verified (`CvHagedornCorrection.lean`).

Finding 5.3 ($\eta = 2\gamma$ fails throughout Phase A)**.** The critical-exponent map included $\eta = 2\gamma$. Its residual against the framework's own scaling relations is non-zero for all $\gamma \in (0, 1)$.

Listing 4: `Identities.lean`: the refutation.

```
theorem eta_2gamma_fails_phase_A (gamma : R) (hlo : 0 < gamma) (hhi : gamma < 1) : ...
```

Correction. $\eta = 2\gamma$ is withdrawn as a Phase-A identity (demoted to a boundary coincidence). Adopted in the Part II revision ledger.

The register. Three of our own equations were wrong; the prover said so before a reviewer could, and the corrected forms are re-verified. A machine-checked erratum names the exact hypotheses of its own falsity — which is more useful to a reader than either the original error or a bare apology.

6 The γ-Möbius Collapse and a Tautology in Disguise

The newest module (`GammaMobius.lean`, 6 theorems) collapses the Padé prediction of Part I to a single reduced variable and settles a question the prose left open: is the "information horizon" a prediction or a definition?

Finding 6.1 (Möbius collapse)**.** With $u = T/(\theta\sqrt{2})$, the Padé exponent is the Cayley/Möbius transform $\gamma = (1-u)/(1+u)$, with inverse $u = (1-\gamma)/(1+\gamma)$ and $\beta = \mathrm{d}\gamma/\mathrm{d}u = -2u/(1+u)^2$. The constant $\sqrt{2}$ is load-bearing: the collapse requires $s^2 = 2$.

Listing 5: `GammaMobius.lean`: the collapse and the load-bearing $\sqrt{2}$.

```
theorem gamma_mobius {s theta T : R} (hs2 : s^2 = 2) (hs : 0 < s) ... :
    (2*theta - T*s) / (2*theta + T*s) = (1 - u) / (1 + u)      -- u = T/(theta*s)
theorem beta_in_u {u : R} (hu : 1 + u != 0) : ...              -- d gamma / d u = -2u/(1+u)^2
```

Finding 6.2 (The information horizon is a tautology)**.** Part I reported $d_{\text{horizon}} = \theta(1-\gamma)\sqrt{2}/(1+\gamma)$ predicting the NIAH-collapse length to within 0.4%. Substituting the Möbius form makes this an *identity*: $d_{\text{horizon}} = T$ exactly. The 99.6% "prediction" is the statement that the model's collapse length equals its training length — true by construction, not a fitted success.

Listing 6: `GammaMobius.lean`: horizon collapses to T.

```
theorem d_horizon_eq_T {s theta T : R} (hs2 : s^2 = 2) (hs : 0 < s) ... :
    theta * (1 - gamma) * s / (1 + gamma) = T
```

This is the most useful kind of formal result: it does not refute the number, it *reclassifies* the claim. $d_{\text{horizon}} = T$ should be presented as a consistency identity of the Padé model, not as an independent quantitative prediction.

A positive bound. The same module family proves an arithmetic–geometric-mean bound $\gamma_\chi \geq 2\sqrt{2}$ (`AM_GM_gamma_chi`), a genuine inequality the prose had only conjectured.

7 Discussion and Limitations

What formalization bought here. Two things a GPU cannot give: (i) it caught three factor/sign errors in tables that had passed human review, each now corrected and re-verified; and (ii) it reclassified a headline "prediction" ($d_{\text{horizon}} = T$) as a tautology, which changes how the result should be reported. Both are epistemic corrections, not cosmetic ones.

What it did not. Formalization here covers the *algebra*, not the empirical claims: no GPU measurement, corpus fit, or causal intervention is inside Lean, and none is claimed to be. Mathlib gaps meant a few derivations (e.g. the C_V integral) were stated at the level of the cumulant identity rather than a full measure-theoretic integral; these are flagged in the manifest. The backbone certifies that the framework's closed forms are mutually consistent under stated hypotheses — it does not certify that the framework describes transformers, which is the job of Parts I–III.

Position. A dense closed-form theory should ship its algebra as machine-checked code. The cost is modest (elementary tactics, minutes to re-verify); the return is an adversarial audit that catches exactly the errors human review misses, and a public ledger that reports the corrections in the same register as the results.

8 Reproducibility

The full library, the per-theorem manifest (file, line, claim, preconditions, tactic, source section), and the build instructions are at `https://github.com/karlesmarin/lean-taf`. Each result in Parts I–III that this paper certifies is cross-referenced by Lean name.

References

[1] C. Marín. *Predicting How Transformers Attend: Analytic Power-Law Theory, Phase Transitions, and Practical Compression Tools*. Zenodo, 2026. DOI:10.5281/zenodo.20314038 (Part I).

[2] C. Marín. *Predicting How Transformers Attend, Part II: A Six-Axis Decomposition with the Learned Imprint $\nu = -1/(2\pi)$*. Zenodo, 2026. DOI:10.5281/zenodo.19960573.

[3] C. Marín. *Predicting How Transformers Attend, Part III: From Attention to Residual Computation*. 2026 (companion depth-axis paper).

[4] L. de Moura, S. Ullrich. *The Lean 4 Theorem Prover and Programming Language*. CADE, 2021.

[5] The Mathlib Community. *The Lean Mathematical Library*. CPP, 2020.