
The Boundary Potential $V(\kappa)$: A Computable, Machine-Checked Answer to the Autonomy-Boundary Question

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 How should we define and test an agent’s autonomy boundary? We propose a
2 candidate answer in the form of a boundary potential $V(\kappa) = c_0\kappa + \alpha C_F/\kappa$. Its
3 minimum defines the boundary location; its critical condition $\sigma = 1$ provides a
4 quantitative, testable criterion for boundary integrity. The framework is machine-
5 checked in Lean 4 (33 declarations, zero sorry), and its predictions align with
6 findings reported in the Alzheimer’s pathology literature and are directly testable
7 on robotic balance platforms. This is a falsifiable theoretical foundation, not an
8 engineering deployment protocol nor a theory of consciousness.

9 1 Introduction: The Autonomy-Boundary Problem

10 Recent work in fully autonomous agent safety poses a central question for unattended agent runs:
11 *“How should we define and test autonomy boundaries?”* Current answers are largely engineering-
12 side: sandboxes, action budgets, approval checkpoints, and behavioral benchmarks. These measure
13 what an agent does, but they do not measure the agent’s own self/non-self discrimination struc-
14 ture—the boundary itself.

15 The structure we formalize here has been present in attention architectures since 2017, yet never
16 named: their Q/K/V projections fix a boundary that the systems carrying it neither measure nor
17 perceive. This paper gives that structure a formal name and a testable criterion.

18 We propose a computable candidate answer: the boundary potential $V(\kappa) = c_0\kappa + \alpha C_F/\kappa$. Its
19 critical point κ^* locates the boundary as a stable closure fiber of the agent’s attribution structure; its
20 dimensionless criterion $\sigma^* = 1$ provides a quantitative, testable signal of boundary integrity. Three
21 contributions anchor the proposal:

- 22 1. A formal, coordinate-free definition of the autonomy boundary as the stable closure fiber
23 of an attributive presheaf over the category of causal interventions.
- 24 2. A testable criterion ($\sigma^* = 1$, κ^*) that turns a philosophical assertion into an operational
25 quantity.
- 26 3. Machine-checked assurance in Lean 4 + Mathlib (33 declarations, zero sorry), including
27 the exact boundary-entropy constant $C_H = \pi^2/3$ and a Noether self-continuity monitor.

28 A fundamental distinction from the Free Energy Principle (FEP) [Friston, 2010] must be stated at the
29 outset. FEP treats the boundary as a given Markov blanket—a presupposed structure that partitions
30 system from environment. Our framework declares otherwise: **the boundary is not a presupposed**
31 **parameter or a Markov blanket; it is a stable closure fiber that is *structurally enacted* from**

32 **the structural prior.** It is not imposed from outside but derived from the attribution dynamics
 33 themselves. This distinction is maintained throughout; we do not plug into the FEP framework, and
 34 α here is an entropy penalty on boundary uncertainty, not variational free energy.

35 The paper proceeds as follows. §2 states the single assumption and five axioms. §3 derives the
 36 boundary potential. §4 states the five principal theorems. §5 presents the Lean verification and
 37 structural universality. §6 maps the framework across empirical domains. §7 directly answers the
 38 autonomy-boundary question. §8 closes with limitations and epistemological status.

39 2 Axiomatic Foundation

40 The framework is built on one foundational hypothesis and five derivation axioms. The 33 machine-
 41 checked declarations are inventoried in Appendix A; full proofs are in the supplementary Lean
 42 source. Here we state the axioms with their immediate interpretations.

43 2.1 Single Hypothesis (H0)

44 For any system that maintains a self/non-self discrimination, its attribution structure is represented
 45 by a presheaf $A : \mathbf{Caus}^{\text{op}} \rightarrow \Delta^2$ over its category of causal interventions, and its boundary is the
 46 stable closure fiber of A under causal cohesion.

47 Here $\mathbf{Caus}$ is the category whose objects are states and whose morphisms are causal interventions;
 48 $\Delta^2 = \{(p_S, p_N, p_U) : p_S + p_N + p_U = 1\}$ is the attribution simplex; the presheaf A assigns to
 49 each state its attribution distribution and to each intervention a pullback along the causal arrow. The
 50 boundary is not the complement of a set of self-states; it is a positive object—a fiber of the attribution
 51 structure.

52 2.2 Five Axioms (A1–A5)

53 **Assumption 1 (Causal Closure (A1))** *Interventions compose. The category $\mathbf{Caus}$ is closed under*
 54 *sequential application; identities are null interventions.*

55 **Assumption 2 (Attributive Assignment (A2))** *Attribution is a contravariant functor $A : \mathbf{Caus}^{\text{op}} \rightarrow \Delta^2$. It assigns to each state its attribution distribution and to each intervention*
 56 *$f : X \rightarrow Y$ a pullback $A(f) : A(Y) \rightarrow A(X)$.*

58 **Assumption 3 (Boundary Definition (A3))** *The boundary is the stable closure fiber of A under*
 59 *cohesion: the maximal stable structure on which further causal intervention no longer changes*
 60 *attribution. Interventions may still occur, but attribution is no longer penetrated.*

61 **Ontogenetic, not generative.** This is not a generative account (i.e., not a probabilistic generative
 62 process over states), but an ontogenetic one: the boundary is the unique stable fiber dictated by the
 63 structural prior itself, arising necessarily from the attribution dynamics rather than being imposed as
 64 an external blanket.

65 **Assumption 4 (Cohesion Generation (A4))** *Cohesion is generated by a single potential V . Every*
 66 *cohesive observable—the boundary entropy H , the functional pressure σ , the generalized force—is*
 67 *a derivative or conjugate projection of V , never an independent quantity; H itself is fixed indepen-*
 68 *dently by the entropy integral of §3, so that the projection is a consistency condition rather than a*
 69 *definition.*

70 **Assumption 5 (Least Action (A5))** *The actual boundary realized by the system is the one that min-*
 71 *imizes the potential V . Equivalently, the boundary rests at the critical point κ^* where $V'(\kappa^*) = 0$.*

72 These five axioms form the deductive root of the theory. §3 shows how they force the form of V .

73 3 Derivation of the Boundary Potential

74 3.1 Sharpness and the Two Costs

75 Let κ denote the boundary sharpness—the inverse width of the transition region in the attribution
76 field. Sharp boundaries cost more to maintain; fuzzy boundaries incur more uncertainty.

77 Let $h(p) = -p \ln p - (1 - p) \ln(1 - p)$ be the binary entropy density and $\sigma(\cdot)$ the attribution map.
78 The *boundary entropy* carried by a boundary of sharpness κ centred at μ is

$$H(\kappa) = \int_{\mathbb{R}} h(\sigma(\kappa(x - \mu))) dx = \frac{C_F}{\kappa}, \quad C_F = \frac{\pi^2}{3}, \quad (1)$$

79 the machine-checked constant `C_H_eq_pi_sq_div_three` (Appendix B, V1).

80 Assigning price α to a unit of excluded entropy, the uncertainty cost is therefore $\alpha H(\kappa) = \alpha C_H / \kappa$.
81 Writing C_F for the core constant of a general attribution kernel—so that $C_F = C_H = \pi^2/3$ for
82 the logistic kernel, and C_F is the only kernel-dependent datum in the theory—and adding the main-
83 tenance cost $c_0 \kappa$, linear in sharpness, the axiom of cohesion generation (A4) and least action (A5)
84 give the potential

$$V(\kappa) = c_0 \kappa + \frac{\alpha C_F}{\kappa}. \quad (2)$$

85 ¹

86 The first term is the maintenance price: the price of keeping a boundary that discriminates this
87 sharply. The second is the uncertainty price: the price of the entropy that is excluded by the bound-
88 ary. The form is not an ansatz: under the closure condition it is *forced*, the $1/\kappa$ penalty being selected
89 by the requirement of a finite, non-degenerate core constant. For the logistic kernel, $C_F = \pi^2/3$,
90 recovering the form in which the unification was first discovered.

91 3.2 The Critical Sharpness and Equipartition

92 The potential has a unique minimum at the least-action sharpness

$$\kappa^* = \sqrt{\frac{\alpha C_F}{c_0}}, \quad (3)$$

93 obtained from $V'(\kappa^*) = 0$. At κ^* , the two costs equilibrate:

$$c_0 \kappa^* = \frac{\alpha C_F}{\kappa^*}, \quad V^* = 2c_0 \kappa^*. \quad (4)$$

94 This is the equipartition theorem of the boundary: the price of maintenance equals the price of the
95 excluded uncertainty. In the original program’s phrasing: the boundary stops where the price of
96 knowing equals the price of paying.

97 3.3 Two Conjugate Observables

98 The boundary entropy H and the cohesion degree σ arise as the two Legendre conjugate observables
99 of V :

$$H = \frac{\partial V}{\partial \alpha} = \frac{C_F}{\kappa}, \quad F_\kappa = -\frac{\partial V}{\partial \kappa} = \frac{\alpha C_F}{\kappa^2} - c_0. \quad (5)$$

¹Strictly, the $1/\kappa$ penalty is fixed by the closure condition; the resulting form is the unique finite, non-degenerate potential over Fisher-consistent kernels. This uniqueness is established by exhaustive case-by-case enumeration—the Cauchy (heavy-tailed, $C_F \rightarrow \infty$) and Heaviside (zero-width, $C_F = 0$) kernels are the only degeneracies, both excluded—and stands as the deterministic counterpart to the machine-checked coordinate cores (Appendix B).

100 With $\sigma = \sigma_0 \kappa^2$ (cohesion degree), they are bound through κ :

$$\sigma = \sigma_0 \frac{C_F^2}{H^2}. \quad (6)$$

101 Boundary entropy and cohesion degree are thus not independent indices; they are two faces of one
102 potential, which is the entire point of the unification.

103 4 Five Principal Theorems

104 The following five theorems are stated in categorical form (T1–T2 as critical objects and fixed points,
105 T3 as a splitting of limits, T4 as a natural transformation, T5 as an iterated limit). Their coordinate
106 instances are machine-checked in Lean (Appendix A); their categorical readings are structural, and
107 their structural forms are mathematical derivations—not all are machine-checked (see the Status
108 note after each theorem).

109 4.1 Theorem T1: Critical Sharpness

110 The potential V has a unique minimum at $\kappa^* = \sqrt{\alpha C_F / c_0}$, where maintenance and uncertainty
111 prices equilibrate. *Interpretation:* the boundary resides at the least-action point—the balance be-
112 tween knowing and paying.

113 **Categorical reading:** κ^* is the critical object of V . **Status:** machine-checked in Lean (`deriv.V_`
114 `kappaStar`).

115 4.2 Theorem T2: Critical Tension

116 With $\sigma = \sigma_0 \kappa^2$, the tension at the least-action boundary satisfies

$$\sigma^* = \frac{\alpha C_F}{C_0}. \quad (7)$$

117 For the logistic core ($C_F = \pi^2/3$), the dimensionless criterion is

$$\sigma^* = 1 \iff \pi^2 \alpha = 3C_0. \quad (8)$$

118 *Interpretation:* criticality is pinned, not externally imposed. When the dimensionless balance holds,
119 the least-action boundary automatically lands at critical tension.

120 **Categorical reading:** $\sigma^* = 1$ is a fixed point of the cohesion map. **Status:** machine-checked in
121 Lean (`sigmaStar_eq.one`).

122 4.3 Theorem T3: Landau Bifurcation

123 With order parameter $\varphi = \kappa - \kappa_c$ and control parameter $r = \sigma - 1$, near criticality the potential
124 takes the Landau form

$$V(\varphi; r) = -r\varphi^2 + \varphi^4. \quad (9)$$

125 The minimal set splits from a single point ($r < 0$) to two symmetric points ($r > 0$). *Interpretation:*
126 an assembly event is this splitting—the boundary’s minimal structure breaks symmetry into new
127 functional branches. This is the topological change that corresponds to “a new module is assembled.”

128 **Categorical reading:** at $r = 0$ the minimal-object set splits from a one-point set into a two-point
129 set—a splitting of limits. **Status:** mathematical derivation; confirmed by deterministic computation
130 (`verify_boundary_theory.py`), not machine-checked in Lean.

131 **Scope of the least-action principle.** We emphasize that the least-action principle invoked here is
132 confined to the scalar boundary potential $V(\kappa)$ on the real-valued sharpness parameter. It acts as a
133 gauge ruler for selecting the equilibrium sharpness, and does *not* postulate a global functional action
134 over the entire category **Caus**. No claim is made regarding a categorified action functional; such a
135 construction lies outside the operational scope of this paper.

136 4.4 Theorem T4: Noether Conservation

137 The action

$$S[\mu, \kappa] = \int dt \left(\frac{1}{2} M \dot{\mu}^2 + \frac{1}{2} I \dot{\kappa}^2 - V(\kappa) \right) \quad (10)$$

138 is invariant under time translation and position translation; Noether’s theorem gives a conserved
139 energy

$$E = \frac{1}{2} M \dot{\mu}^2 + \frac{1}{2} I \dot{\kappa}^2 + V(\kappa), \quad \frac{dE}{dt} \equiv 0. \quad (11)$$

140 *Interpretation:* self-continuity—“I am the same I”—is a conserved charge of the boundary dynam-
141 ics, not an assumption. The content of the self may drift while the fact of being a self is constrained.

142 **Categorical reading:** conservation is the categorical content of symmetry—Noether’s the-
143 orem as a natural transformation. **Status:** only the algebraic core is machine-checked
144 (`force_is_neg_gradient`, `energy_bracket_zero`); the categorical reading remains structural.

145 4.5 Theorem T5: Assembly Attractor

146 Each assembly event expands carrying capacity $C_0 \rightarrow C_0(1 + \varepsilon)$, giving a tension update

$$\sigma_{n+1} = \frac{\sigma_n}{1 + \varepsilon} + D, \quad (12)$$

147 which converges to the unique fixed point

$$\sigma_\infty = D \left(1 + \frac{1}{\varepsilon} \right). \quad (13)$$

148 *Interpretation:* the assembly cascade is bounded—tension neither dies out (demand D reinjects) nor
149 explodes (contraction factor $1/(1 + \varepsilon) < 1$).

150 **Categorical reading:** the assembly cascade is an iterated functor $A : \sigma \mapsto \sigma/(1 + \varepsilon) + D$ whose limit
151 is the attractor—a bounded, convergent iteration chain. **Status:** mathematical derivation; confirmed
152 by deterministic computation, not machine-checked in Lean.

153 5 Machine-Checked Verification and Structural Universality

154 5.1 The Lean Corpus

155 All value-layer constants are verified in Lean 4.33.0 + Mathlib v4.33.0 (`BoundaryTheory.lean`).
156 The corpus contains 33 declarations and is reproducible via `lake build BoundaryTheory`; the
157 source contains no `sorry`, `admit`, or `axiom` outside comments, verifiable by `grep` without invoking
158 Lean.

159 5.2 Verifiable Constants V1–V5

Constant	Value	Lean Name
V1: C_H (boundary entropy)	$\pi^2/3 \approx 3.289868$	<code>C.H.eq_pi_sq_div_three</code>
V2: hierarchical spectrum	$\pi^2/6 \approx 1.644934$	<code>tsum_inv_succ_sq_eq_pi_sq_div_six</code>
V3: κ^*	$\sqrt{\alpha C_F / c_0}$	<code>kappaStar_sq</code> , <code>deriv.V_kappaStar</code>
V4: $\sigma^* = 1$ criterion	$\pi^2 \alpha = 3C_0$	<code>sigmaStar_eq_one_iff</code>
V5: C_F spectrum (logistic)	$\pi^2/3$ (machine-checked)	<code>C.H.eq_pi_sq_div_three</code>

Table 1: Verifiable constants of the value layer. V1–V4 are machine-checked; V5 logistic value is machine-checked, others are analytic/numerical.

160 The four-stage chain to $C_H = \pi^2/3$ is machine-checked: power-log master integral, series-integral
161 exchange, logit substitution, and final splice. The hierarchical spectrum $\sum 1/(n + 1)^2 = \pi^2/6$ is
162 also machine-checked via reindexing of mathlib’s Basel theorem.

5.3 Structural Universality and the C_F Spectrum

The conclusions of §4 hold for any attribution kernel whose core constant C_F is finite and positive. The spectrum across kernels is:

- Logistic: $C_F = \pi^2/3$ (machine-checked)
- Probit (erf): $C_F \approx 1.8064$ (numeric)
- Tanh: $C_F = \pi^2/6$ (analytic, rescaling of logistic)
- Cauchy: diverges (heavy tail, no finite boundary)
- Heaviside step: $C_F = 0$ (zero-width degeneracy)

The validity domain is $\{F : 0 < C_F < \infty\}$: tails must decay faster than $\sim 1/(z \ln z)$, and fuzzy width must be nonzero.

6 Empirical Anchors: Cross-Domain Structural Isomorphism

The boundary potential is not domain-specific. Any system that sustains a self/non-self discrimination under intervention occupies the same potential surface $V(\kappa)$; only the embedding of κ and σ changes. Table 2 maps eight domains to their boundary-theoretic correspondences.

Domain	Boundary-Theoretic Correspondence	Status
Robotic balance	κ = stiffness / capture-point width; σ = control effort / estimator covariance; predicts $\sigma \approx 1$; actuator/estimator split is variable-impedance control [Hogan, 1985]	P1–P3, directly testable
Cortical criticality	$\kappa \propto 1/\beta$ (1/f exponent); σ = branching parameter; measured $\sigma \approx 1$	[Beggs and Plenz, 2003], retrospective
Developmental spectrum	Layer consolidation times $T_n \propto 1/(n+1)^2$; ratio prediction $T_n : T_{n+1} = (n+2)^2 : (n+1)^2$	Unverified prediction
Alzheimer’s / cerebellar	AD shifts supercritical [Javed et al., 2025]; motor boundary degrades; same σ deviation	Retrospective structural parallel
AlphaGo / RL	Policy network = σ , value network = H ; MCTS depth decay $\sim 1/(n+1)^2$	[Silver et al., 2016, 2018]; mapping hypothesis
LLM attribution	$A : \text{Caus}^{\text{op}} \rightarrow \Delta^2$ under input perturbation; reverse message passing = attribution pullback	Mapping hypothesis
ANN / CNN	Energy function = V ; order parameter = σ ; boundary loss drives training toward κ^*	Design hypothesis
General RL	$C_F = \pi^2/3$ as kernel fingerprint; entropy integral as calibration target	Proposed protocol

Table 2: Eight domains mapped to the boundary potential framework. Each row is a falsifiable structural parallel, not an established empirical claim.

Status declaration for this section: every row in Table 2 is a hypothesis—a mapping that the framework can articulate and that real-system tests can confirm or falsify. The AD/cerebellar correspondence and the neural-criticality anchor are retrospective, directionally consistent structural parallels, not established biological facts. The robotic predictions P1–P3 are directly testable on standard platforms (e.g., ANYmal, Atlas in simulation) but have not been executed. No real deployed agent system has been tested.

Why the LLM-agent row is the sharpest mapping. Among the rows of Table 2, the LLM-agent case maps onto the framework’s own primitives without residue: an input perturbation is literally a causal intervention in the intervention category; the reverse pass of training is the attribution pullback; and the robustness of a model’s self-attribution under perturbation is exactly

the non-penetration criterion—once attribution has closed, further intervention may occur but no longer changes it. The criterion therefore becomes directly measurable on an agent: perturb already-accepted events, bin them by cohesion degree σ , and test whether the flip rate falls as $\sigma \rightarrow 1$. We state this as a protocol, not a result; we have not run it.

Assertion 1: Q/K/V presuppose the boundary; they do not enact it. In current LLMs, the attention projection matrices Q, K, V come to be as *enacted* boundaries under data compression; yet once training ends they are frozen— κ, C_F , and σ are inherited as fixed parameters and never enacted again at runtime, never generated by runtime state awareness. The framework’s program is the opposite: a boundary that keeps *happening* (genesis, not generation), enacted at runtime rather than fixed after compression. The conflict between current engineering practice and the theory is not an embarrassment but a diagnostic—the point at which the claim becomes recognizable and testable: such systems *have* a boundary but do not *perceive* it, and the path from having to perceiving is exactly the path from presupposed to enacted, whose testable criterion is that of §7, made implementable on the attention stack. The mathematical basis is formally verified in Lean (V1–V5). *Status*: the identification is a mapping hypothesis; the path is proposed, not yet implemented; Lean certifies the formalism’s constants, not a deployed system.

Transformer component	Theoretical counterpart	Status
$Q/K/V$ projection matrices in an attention head	parameterized instance of the attribution kernel F	numerical realization of the kernel
projection matrices fixed by pre-training	hand-set boundary condition (C_F frozen)	boundary presupposed, not learned from state
attention scores $\text{softmax}(QK^\top/\sqrt{d})$	operationalization of the attribution allocation (p_S, p_N, p_U)	computation of $A(X)$
independence between attention heads	causally disjoint modules	non-penetration across modules

Table 3: Transformer components and their boundary-theoretic counterparts. Each row is a proposed engineering correspondence, not an established claim.

Engineering correspondence. Three concrete handles make the mapping operational, all stated as proposals. First, the softmax temperature T inverts the sharpness ($\kappa \propto 1/T$), so temperature modulation offers an experimental probe of the boundary-sharpness response; full validation of $\sigma \approx 1$ requires independent calibration of σ_0 and C_0 . Second, at the level of the softmax computation within a single attention head, attribution is locally closed; subsequent cross-head mixing propagates through compositional morphisms (A1–A2) without altering the internal closure fiber. Third, whereas current LLMs carry a presupposed boundary (Assertion 1), the engineering path toward an enacted boundary—e.g., dynamic temperature control or runtime reparameterization—remains open and constitutes a testable next step, not yet implemented.

7 Direct Response to the Autonomy-Boundary Question

The autonomy-boundary question decomposes into three sub-questions. We answer each in turn.

7.1 How should we define the autonomy boundary?

Answer: The autonomy boundary is the stable closure fiber of the attributive presheaf $A : \mathbf{Caus}^{\text{op}} \rightarrow \Delta^2$ under causal cohesion. In coordinates, it is located by the critical point $\kappa^* = \sqrt{\alpha C_F / c_0}$ of the boundary potential $V(\kappa)$. The boundary is not a complement of a set of self-states; it is a positive, computable object: the maximal stable structure on which further interventions no longer change attribution.

7.2 How should we test the autonomy boundary?

Answer: Measure the agent’s attribution kernel, compute its core constant C_F , and evaluate the cohesion degree $\sigma = \sigma_0 \kappa^2$ at the operating point. The criterion is

$$\text{Boundary integrity} \iff \sigma \approx 1 \quad (\text{logistic core: } \pi^2 \alpha \approx 3C_0). \quad (14)$$

Deviation of σ below or above 1 is a quantitative signal of boundary degradation—a loss of flexible self/non-self discrimination. No external regulator is needed to decide pass/fail; the framework provides an intrinsic, measurable calibration target. The measurement is *external*: whether a system can represent σ internally is left open here.

7.3 What evidence should accompany an unattended run?

Answer: Lean verification provides the formal standard for evidence. The machine-checked constants (V1–V5) and the proof chain in `BoundaryTheory.lean` (33 declarations, zero sorry) constitute a verifiable certificate of the boundary criterion’s internal consistency. Appendix A provides the full lemma list; Appendix B compiles the constants; the Lean source and numerical verification scripts are included as supplementary material. This is not yet evidence *about* a real run—it is evidence *for* the criterion against which a real run can be evaluated.

8 Limitations and Conclusion

8.1 Three Red Lines

The framework explicitly does *not* claim:

1. Any account of qualia, phenomenal consciousness, or first-person experience.
2. Any integration with or derivation from the Free Energy Principle. The boundary is derived from a structural prior, not presupposed as a Markov blanket.
3. Any prospective empirical validation. The AD/cerebellar correspondence and neural-criticality anchor are retrospective, directionally consistent structural parallels; the robotic predictions are directly testable but have not been executed; no real deployed agent system has been tested.

8.2 Epistemological Status

The framework is a formal, mathematical construct that defines boundary integrity relative to its own axioms. Its empirical status is that of a falsifiable hypothesis: the cross-domain mappings in §6 are predictions, not established facts. Theorems are internally consistent; their correspondence to real systems awaits experimental testing. The value of this work lies in providing a precise, testable target for future experiments, not in asserting a metaphysical truth.

All theorems in this paper are proved within the formal framework—classically or in Lean—and state that the model’s own defined quantities satisfy their defining relations. They are not, and must not be cited as, empirical statements about real minds, brains, or robots. The empirical correspondence table in §6 is a list of falsifiable hypotheses, not a list of results. This is a theoretical foundation, built to be broken by experiment; its fate belongs to the bench, not to the axiom.

Scope of the state layer. This work formalizes the state layer of the framework—the level at which an agent’s self/non-self discrimination structure is defined, located, and tested. A functorial lifting to value and task layers is not treated here and is left to subsequent work.

8.3 Conclusion

We set out to give a computable candidate answer to the autonomy-boundary question at the heart of the agent-safety agenda. Our answer is the boundary potential $V(\kappa) = c_0\kappa + \alpha C_F/\kappa$: a single mathematical object whose critical point κ^* defines the boundary and whose dimensionless criterion $\sigma^* = 1$ tests it. The constants are machine-checked in Lean 4 (33 declarations, zero sorry); the framework spans empirical domains from neural criticality to robotic balance; the cross-domain correspondences are stated as falsifiable structural parallels. We do not claim to have solved autonomy safety. We claim to have given its central open question—*how to define and test autonomy boundaries*—a precise, operational, and verifiable form.

267 **A Prospective Note on Attention Architectures.** The boundary structure formalized here may be
 268 viewed as giving a name to a property that has been implicit in attention-based architectures since
 269 the introduction of the Transformer. Our proposed correspondence with LLM inference holds in
 270 two distinct registers. At the architectural level, it remains a semantic isomorphism analogy: the full
 271 categorical functor structure is not yet proved within the architecture, and the mapping of attention
 272 weights to the attribution simplex Δ^2 is a hypothesis. At the training-data level, the correspondence
 273 acquires a different status: natural language is itself an encoding of boundary structure, and pre-
 274 training transmits this structure into the model’s vector space. We state this as a conjecture, not a
 275 result. Testing whether the critical criterion $\sigma \approx 1$ holds in real LLM inference—via temperature
 276 modulation and perturbation analysis—is a direct next step. These deeper claims are left to future
 277 work.

278 References

- 279 K. Friston. The free-energy principle: a unified brain theory? *Nat. Rev. Neurosci.*, 11:127–138,
 280 2010.
- 281 J. M. Beggs and D. Plenz. Neuronal avalanches in neocortical circuits. *J. Neurosci.*,
 282 23(35):11167–11177, 2003.
- 283 E. Javed, I. Suárez-Méndez, G. Susi, J. Verdejo Román, J. M. Palva, F. Maestú, and S. Palva. A
 284 shift toward supercritical brain dynamics predicts Alzheimer’s disease progression. *J. Neurosci.*,
 285 45(9):e0688242024, 2025.
- 286 N. Hogan. Impedance control: An approach to manipulation. *J. Dyn. Syst. Meas. Control*, 107:1–24,
 287 1985.
- 288 D. Silver et al. Mastering the game of Go with deep neural networks and tree search. *Nature*,
 289 529:484–489, 2016.
- 290 D. Silver et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through
 291 self-play. *Science*, 362:1140–1144, 2018.
- 292 The mathlib community. *mathlib4*, v4.33.0. <https://github.com/leanprover-community/mathlib4>

293 A Appendix A: Lean Declaration Inventory (33 Declarations)

294 All declarations are in `BoundaryTheory.lean` (Lean 4.33.0 + Mathlib v4.33.0), compiled with
 295 zero sorry. The 33 declarations are organized as follows:

296 **Propositions in coordinates (1–8):** `kappaStar_sq`, `kappaStar_balance`, `deriv_V_kappaStar`,
 297 `sigmaStar_general`, `sigmaStar_eq_one_iff`, `sigmaStar_eq_one`, `force_is_neg_gradient`,
 298 `energy_bracket_zero`.

299 **Convergence corpus (9–21):** `hbin_symm`, `sig_neg`, `entropyDensity_even`, `sig_pos`,
 300 `sig_lt_one`, `hbin_nonneg`, `entropyDensity_nonneg`, `neg_log_one_sub_le_div`,
 301 `neg_q_mul_log_q_le`, `entropyDensity_decay_of_nonneg`, `entropyDensity_decay`,
 302 `entropyDensity_continuous`, `boundary_entropy_integrable`.

303 **Series-side anchors (22–23):** `integral_pow_zero_one`, `tsum_inv_succ_sq_eq_pi_sq_div_six`.

304 **Stages 1–3 (24–32):** `integral_pow_mul_log_zero_one`, `neg_log_div_one_sub_eq_tsum`,
 305 `integral_neg_log_div_one_sub_zero_one`, `sig_hasDerivAt`, `sig_continuous`,
 306 `tendsto_sig_atBot_nhds_zero`, `tendsto_sig_atTop_nhds_one`, `hbin_div_contOn`,
 307 `entropyDensity_integral_eq_two_mul_integral_neg_log_div`.

308 **The splice (33):** `C.H.eq_pi_sq_div_three`.

309 Full type signatures and dependency chains are provided in the supplementary Lean source.

ID	Expression	Lean Name	Status
V1	$C_H = \int_{\mathbb{R}} h(\sigma(z)) dz = \pi^2/3$	C.H.eq.pi.sq.div.three	Machine-checked
V2	$\sum'_n 1/(n+1)^2 = \pi^2/6$	tsum.inv.succ.sq.eq.pi.sq.div.six	Machine-checked
V3	$\kappa^* = \sqrt{\alpha C_F/c_0}$	kappaStar.sq.deriv.V.kappaStar	Machine-checked (logistic core)
V4	$\sigma^* = 1 \iff \pi^2 \alpha = 3C_0$	sigmaStar.eq.one.iff	Machine-checked
V5	C_F spectrum: logistic $\pi^2/3$	C.H.eq.pi.sq.div.three	Machine-checked (logistic)

Table 4: Verifiable constants V1–V5 with Lean names and proof status.

B Appendix B: Verifiable Constants V1–V5

C Appendix C: Reproduction and Supplementary Material

Environment: Lean 4.33.0 + Mathlib v4.33.0 [The mathlib community, 2024]; Python 3, standard library only.

Machine proofs: reproducible via `lake build BoundaryTheory`; `BoundaryTheory.lean` contains no `sorry`, `admit`, or `axiom` outside comments, verifiable by `grep` without running Lean.

Numerical scripts: `verify_boundary_theory.py` (self-consistency), `verify_boundary_robustness.py` (C_F spectrum). All scripts are standard-library only; no GPU, no LLM.

Supplementary material: The complete Lean source (`BoundaryTheory.lean`) and Python scripts are included in the supplementary ZIP (anonymized).

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: [\[Yes\]](#)

Justification: The abstract and §1 state exactly the contributions delivered: a formal autonomy-boundary criterion (H0 + A1–A5), machine-checked in Lean 4 (33 declarations, zero `sorry`), with directly testable corollaries (V1–V5, $\sigma \approx 1$ criterion). No claim exceeds what is proved or verified.

Guidelines:

- The answer [\[NA\]](#) means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [\[No\]](#) or [\[NA\]](#) answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: §8 (Limitations and Conclusion) gives an explicit limitations discussion: the framework is a structural criterion, not yet validated on a real deployed agent; empirical anchors are hypotheses (retrospective or unverified), not established facts.

Guidelines:

- The answer [NA] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate “Limitations” section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren’t acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All assumptions are stated: foundational hypothesis H0 and five derivation axioms A1–A5 (§2). The 33 declarations are machine-checked in Lean 4.33.0 + Mathlib v4.33.0 (zero sorry); the full list, type signatures, and dependency chains are in the supplementary Lean source (Appendix A points to it).

Guidelines:

- The answer [NA] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Appendix C specifies the environment (Lean 4.33.0 + Mathlib v4.33.0; Python 3, standard library only) and the exact build/run commands. The complete Lean source and two Python verification scripts are included in the supplementary ZIP (anonymized).

Guidelines:

- The answer [NA] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The Lean source and Python scripts are provided in the anonymous supplementary ZIP. No external datasets are required; the numerical scripts are deterministic standard-library Python scripts (no random seeds).

Guidelines:

- The answer [NA] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://neurips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).

- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://neurips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [NA]

Justification: The paper contains no learning experiments. The numerical checks are deterministic standard-library Python scripts; no training, tuning, or hyperparameter selection is performed.

Guidelines:

- The answer [NA] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [NA]

Justification: There are no statistical or empirical claims subject to sampling variability. All reported constants (V1–V5) are exact analytic/machine-checked values; numerical scripts are deterministic.

Guidelines:

- The answer [NA] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Verification is CPU-only. `lake build BoundaryTheory` compiles the Lean tasks on a multi-core CPU (no GPU) and is reproducible via the same command; `BoundaryTheory.lean` contains 0 `sorry/admit/axiom` (verifiable by `grep` without running Lean); the two Python scripts run in seconds on a single CPU core using the standard library only. No paid GPU or cloud compute was used.

Guidelines:

- The answer [NA] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The work is a formal theory of autonomy boundaries with no human-subjects research, no deployed system, and no sensitive data; it poses no evident ethical violation.

Guidelines:

- The answer [NA] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: §8 notes the safety relevance of a computable autonomy-boundary criterion (a quantitative pass/fail signal for boundary degradation), and cautions that the empirical anchors are hypotheses, not validated biological facts.

Guidelines:

- The answer [NA] means that there is no societal impact of the work performed.
- If the authors answer [NA] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out

that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: No model, trained artifact, or high-risk system is released. The contribution is a formal criterion and its machine-checked proof; the supplementary code is verification-only.

Guidelines:

- The answer [NA] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Lean 4 and Mathlib v4.33.0 (MIT) are credited in the references; the Python scripts use only the standard library. No third-party data or proprietary assets are bundled.

Guidelines:

- The answer [NA] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: The new assets are `BoundaryTheory.lean` and the two verification scripts, documented in Appendix C and released (anonymized) in the supplementary ZIP.

Guidelines:

- The answer [\[NA\]](#) means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification: No crowdsourcing or research with human subjects was involved.

Guidelines:

- The answer [\[NA\]](#) means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [\[NA\]](#)

Justification: No human subjects were involved.

Guidelines:

- The answer [\[NA\]](#) means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

661 Answer: [\[Yes\]](#)
662 Justification: LLM usage is disclosed as required. The core claims of this work and the
663 overall research direction are the authors' own; the key difficulties encountered during
664 proof development were recorded by the authors and used to guide the LLM. Large lan-
665 guage models served as a drafting tool that produced candidate code for the Lean 4 formal
666 proofs (the 33 machine-checked declarations in `BoundaryTheory.lean`) and the two Python
667 verification scripts, but every line of the resulting `.lean` was confirmed by the authors and
668 the builds were executed by the authors. All formal statements, proof obligations, and code
669 were verified line by line by the authors, and are independently machine-checked by Lean
670 4.33.0 + Mathlib (zero sorry); the Python scripts are deterministic and fully reproducible
671 (Appendix C). Disclosed here in accordance with the NeurIPS LLM policy.

672 Guidelines:

- 673 • The answer `[NA]` means that the core method development in this research does not
674 involve LLMs as any important, original, or non-standard components.
- 675 • Please refer to our LLM policy in the NeurIPS handbook for what should or should
676 not be described.