Executive Framing

for System 2 Engineering Teams

Problem Statement: Why "Best-of-N" Isn't Enough

Current LLM-based reasoning systems fail not primarily due to lack of knowledge, but due to *structural instability*: hallucinations, brittle chains of thought, and smoothly consistent falsehoods that evade confidence-based detection.

Crucially, existing mitigations (Best-of-N, Self-Consistency) rely on the assumption that errors are random noise. They fail when models exhibit "Mode Collapse" into confident hallucinations. If a model generates a smooth falsehood 5 times, Best-of-N merely reinforces the error as the "consensus." Eidoku rejects it because the *structural cost* of the falsehood remains high regardless of how frequently or fluently it is generated.

Key Idea

Eidoku introduces a lightweight **structural verification layer** that evaluates candidate reasoning paths by measuring *semantic tension*: the cost of maintaining contextual continuity while accommodating a hypothesis or inference.

The system does *not* attempt to determine truth, simulate consciousness, or replace probabilistic generation. Instead, it enforces a simple principle:

"Reasoning that preserves context is cheap;
reasoning that breaks context is expensive."

Architecture Overview

- **System 1 (Generator)**: Produces multiple candidate reasoning paths via standard techniques.
- **System 2 (Eidoku)**: Scores candidates using accumulated semantic tension (τ) and context penalty (C).
- **Selection**: Low-tension, high-context candidates are preferred over high-probability hallucinations.

What This Is Not

- Not a truth oracle or symbolic logic engine.
- Not a consciousness model.
- **Eidoku is a debugger for reasoning structure**, not an epistemic authority.

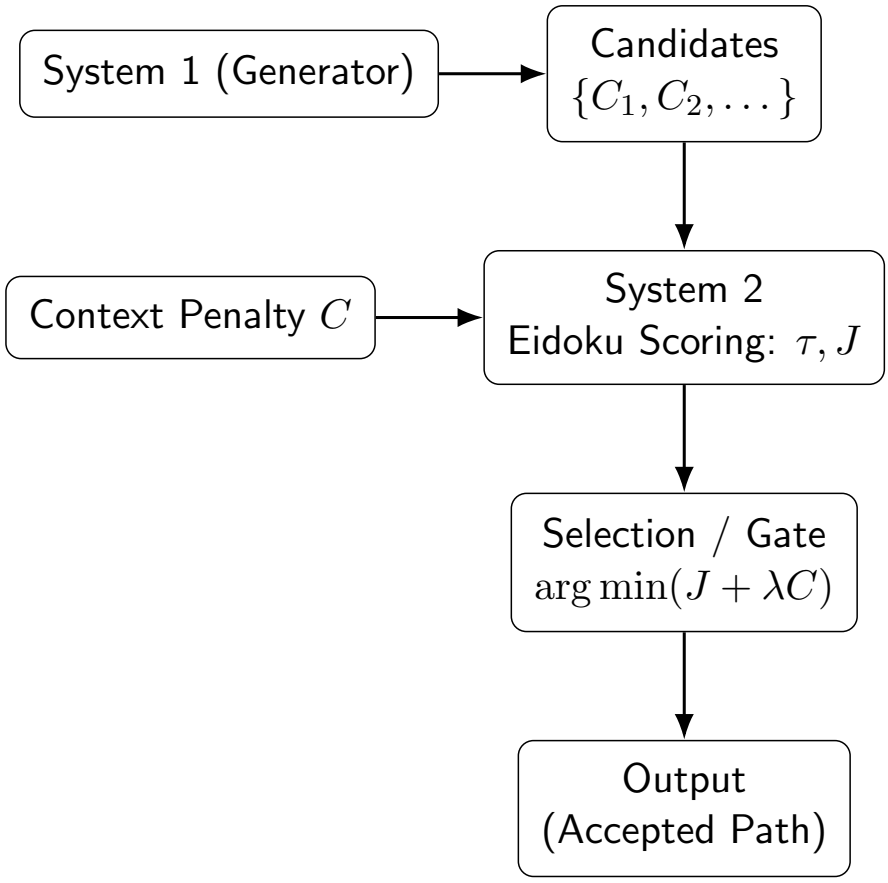

Figure 1: Eidoku Architecture: A structural gate over multiple candidates.

Why Not Best-of-N?

Aspect	**Best-of-N / Self-Consistency**	**Eidoku (Structural Verification)**
Primary Objective	Maximize likelihood / consensus	Minimize semantic tension under preserved context
Failure Mode	**Smooth falsehoods remain low-cost** and survive reranking	**Smooth falsehoods incur structural cost** when context is enforced
Hallucination	Implicit; relies on calibration	Explicit; hallucinations appear as high-cost candidates
Computational	Cost grows linearly with N	Cost applied selectively at high-risk branching points

Table 1: Comparison: Probabilistic Consensus vs. Structural Verification

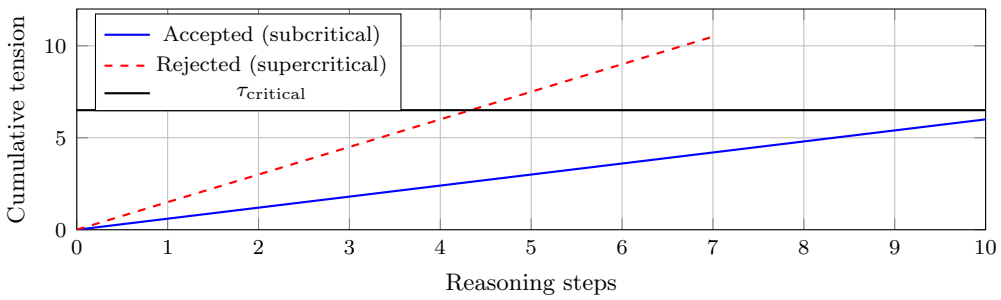

Figure 2: Stepwise Rejection: Chains are aborted early when tension spikes.

Why This Helps

- **Orthogonal to Probability**: Detects errors that are statistically likely (high P) but structurally impossible.
- **Penalizes Context Erasure**: Makes contradictions expensive without hard rules.
- **Explainable**: Rejection reasons are localized to specific "tension spikes" (Fig 2).

Bottom Line

You do not need CPM to be true. You do not need Catelingo fully implemented. You only need one assumption:

"Breaking context should cost more than preserving it."

Eidoku operationalizes this assumption as a practical System 2 verification layer.

Theoretical Proposal & Design Specification:

CATELINGO and EIDOKU: A Practical Framework for Simulated System 2 Reasoning via Structural Tension Minimization

Shinobu Miya

December 14, 2025

Abstract

Current approaches to System 2 reasoning (OpenAI o1, DeepMind AlphaProof) remain fundamentally probabilistic, lacking a principled detection mechanism for structural inconsistency. This specification proposes a rigorous alternative: by simulating topological closure ($\mathcal{B}_{\text{sim}} = 1$) and minimizing geometric tension (τ), we can detect and reject hallucinations before they propagate—without relying on probability alone. Instead of a limited prototype, we provide a complete mathematical and architectural foundation designed for immediate scalability and exploration by industry labs and the research community. Based on the Critical Projection Theory (CPM), we introduce two core concepts: **Catelingo**, a high-resolution geometric language, and **Eidoku (Entropic Reading)**, a variational algorithm. We demonstrate that by virtually imposing a topological closure constraint ($\mathcal{B}_{\text{sim}} = 1$), logical consistency can be quantified as tension energy $\mathcal{J}$, offering a principled path toward substantial hallucination reduction without invoking consciousness, phenomenology, or any form of subjective modeling. The proposed mechanisms can be instantiated with existing LLM outputs and require no retraining of the base model.We distinguish two implementation paths: (1) minimal Eidoku using existing NLP tools, deployable immediately; (2) full Eidoku with Catelingo, enabling automated constraint enforcement, as a long-term target.

1 Introduction: Limits of Probabilistic Reasoning

Despite recent advances in Chain-of-Thought prompting and post-hoc verification, current LLM systems still lack a reliable mechanism for detecting when their own reasoning has structurally failed. In practice, most failures manifest not as low-probability outputs, but as confidently generated, internally inconsistent reasoning paths.

Despite considerable investment in reasoning enhancement (OpenAI's o1, Anthropic's extended thinking, Google's AlphaProof), a fundamental gap remains: existing approaches optimize generation (max P) but lack a structural notion of when reasoning has failed. Post-hoc verification detects some errors, but only after costly generation.

What if we could evaluate reasoning paths *before* accepting them, using a geometric consistency criterion orthogonal to probability?

This proposal argues that this failure mode can be addressed not by deeper generation, but by introducing an explicit notion of structural tension. Large Language Models (LLMs) rely on **probabilistic inference** ($\max P$), which imposes inherent limits on **logical consistency checking**. We specify a solution based on CPM (Critical Projection and the Geometry of Meaning). We introduce *Eidoku* as a mechanism to simulate **System 2 (deliberative)** reasoning by optimizing semantic topology rather than token probability.

This framework is designed to reduce silent reasoning failures that escape probability-based scoring, a major source of production hallucinations, evaluation instability, and downstream trust degradation.

2 Theoretical Background: Foundations and Virtual Closure

Theoretical Dependency and Limitations This framework rests on the theoretical foundation of CPM (Critical Projection and the Geometry of Meaning), which is itself a novel and as-yet unvalidated theory.

If CPM's core claims—in particular, the necessity of $\mathcal{B} \geq 1$ for consciousness and the interpretation of τ as a meaningful metric of structural tension—are subsequently refuted, the theoretical justification for Eidoku would require revision.

Nevertheless, the practical utility of geometric consistency checking may persist independently of CPM's ultimate status, much as Newtonian mechanics remains operationally useful despite later relativistic corrections.

CPM models semantic space as a topological structure over an atlas $\mathcal{A}$ and defines interpretation as a minimization problem of an energy functional $\mathcal{J}$.

2.1 Virtual Closure Constraint ($\mathcal{B}_{\text{sim}}$)

In the original CPM theory, tension accumulation requires a physical closure $\mathcal{B} \geq 1$. Since current cloud-based LLMs possess $\mathcal{B} \approx 0$, true semantic refinement ($\Pi \to \Pi'$) does not occur naturally. Therefore, Eidoku operates by **virtually fixing the closure parameter** in the computational environment:

$$\text{Assume } \mathcal{B}_{\text{sim}} = 1 \implies \text{Maximize } \tau(x) = \left\| \frac{\delta \mathcal{E}}{\delta M(x)} \right\|_{g_0} \tag{1}$$

Eidoku does not generate consciousness but simulates the **structural penalty of inconsistency** by treating logical defects as energy costs.

2.2 Engineering Justification: The Weak CPM Hypothesis

A fundamental theoretical objection to applying CPM to cloud-based architectures is the *Zero-Closure Problem*: as established in the core theory, virtualized systems satisfy $\mathcal{B} \approx 0$ and thus lack the physical boundary required to sustain genuine semantic tension. Consequently, we explicitly state that this framework does not claim to instill consciousness or subjective experience into the model.

Instead, we introduce the **Weak CPM Hypothesis** as the core engineering principle of Eidoku:

> *While physical closure ($\mathcal{B} \geq 1$) is necessary for subjective experience, the computational simulation of closure ($\mathcal{B}_{sim} = 1$) is sufficient to recover the structural error-correction dynamics characteristic of System 2 reasoning.*

We justify this approximation strategy through three structural parallels:

1. **As-if Closure (Virtual Regularization):** Just as a flight simulator obeys aerodynamic laws without physical air, Eidoku enforces topological constraints *as if* the system were closed. By virtually fixing $\mathcal{B}_{sim} = 1$, we introduce a strong regularization term that penalizes reasoning paths that would "dissipate" (fail to cohere) in a physically closed substrate.

2. **Translation of Stress to Cost:** We map the thermodynamic "physical stress" defined in CPM to "computational complexity" (e.g., Minimum Description Length) in the digital domain. While this simulated tension generates no physical heat, it provides a rigorous mathematical signal of logical incoherence that functions analogously to biological stress.

3. **Orthogonality to Probability:** The crucial utility of CPM in this context is not phenomenological but architectural. It offers a consistency metric (Topological Tension) that is mathematically orthogonal to the generation metric (Likelihood). This allows the system to detect *smooth falsehoods*—hallucinations that are probabilistically likely but topologically inadmissible—which purely probability-based verifiers systematically miss.

Thus, Eidoku is formally defined not as an implementation of consciousness, but as the *algorithmic transfer of structural robustness* from closed biological systems to open virtual architectures.

3 Geometric Language: Catelingo

Positioning. Catelingo represents an *idealized semantic substrate* that maximally reduces initial ambiguity in meaning representation. Importantly, **Eidoku does not require Catelingo to be fully realized in order to function**. In practical settings, existing representations—such as natural language text, structured prompts, tool-call graphs, or intermediate reasoning traces—can serve as provisional substrates for tension evaluation. Catelingo should be understood as a *convergence target*, not a prerequisite.

Natural languages are polysemous, leading to high initial mismatch energy. Catelingo adopts the CPM atlas $\mathcal{A}$ directly as its lexical structure.

3.1 Handling Vocabulary Explosion via Axiomatic Bootstrap

A complete construction of Atlas $\mathcal{A}$ faces combinatorial explosion. To address this, we propose a two-step **Axiomatic Bootstrap**:

1. **Core Kernel Definition**: We explicitly define a small, consistent kernel of axioms $\mathcal{A}_0$ (approx. 100 primitives: $\mathcal{A}_0 = \{a_{\text{phys}}, i_{\text{info}}, u_{\text{clos}}, \dots\}$) based on CPM principles. This serves as the immutable "constitutional" structure.

2. **Partial Approximation**: The remaining vocabulary is generated via **Variational Inference (VI)**, constrained to maintain topological consistency with $\mathcal{A}_0$.

Rather than defining the entire manifold ab initio, the system dynamically constructs a **local low-dimensional approximation** anchored to the axiomatic core, preventing the re-learning of statistical noise.

Crucially, this bootstrap does not require retraining the base LLM. The kernel $\mathcal{A}_\prime$ and tension functions τ_λ operate as a post-processing verification layer, making Eidoku immediately deployable on existing models (GPT-4, Claude, Gemini) without modification of their weights or architecture.

We hypothesize that $\tau_S truct$ can detect hallucinations missed by probability scoring, but rigorous benchmarking is required to quantify its effectiveness.

4 Variational Reading: Eidoku

Crucially, Eidoku does not generate new reasoning steps; it evaluates and constrains reasoning already produced by the base model.

Eidoku discovers an optimal interpretative path S^* that minimizes the total tension $\mathcal{J}$.

$$S^* = \arg\min_S \sum_{(i,j)\in E} w_{ij} \cdot \tau_\lambda(S_i, S_j) \tag{2}$$

This process acts as a **logical debugger**, rejecting generated paths where tension exceeds a virtual critical threshold ($\tau > \tau_c$).

4.1 Minimal Eidoku Prototype

A minimal Eidoku prototype can be implemented without Catelingo or specialized semantic infrastructure. One possible instantiation is as follows:

1. Parse a generated chain-of-thought into sentence-level nodes.
2. Assign a limited subset of tension labels (e.g., C, H, F).
3. Compute the total tension $\sum \tau_\lambda$ and flag or reject traces exceeding a fixed threshold.

This minimal configuration is sufficient to demonstrate the core behavior of Eidoku: detecting structurally unstable reasoning without retraining the base model.

4.2 On Hallucination in the Natural Language to Catelingo Mapping

A natural objection arises: if natural language must first be mapped into Catelingo, and this mapping is performed by a probabilistic System 1 model, does hallucination at this stage not invalidate the entire pipeline?

Within CPM, this concern is expected rather than problematic. Projections are neither unique nor guaranteed to be correct. Accordingly, Eidoku does not rely on a single mapping from natural language into Catelingo.

Instead, System 1 is used to generate multiple candidate projections $\{\Pi_k\}$. These candidates are then evaluated by System 2 using a joint objective: semantic tension minimization and structural preservation.

Hallucinated projections are not catastrophic failures; they appear as high-cost candidates, either by introducing internal inconsistency (high $\mathcal{J}$) or by suppressing contextual anchors (high $\mathcal{C}$), and are therefore discarded.

In this architecture, System 1 is responsible for proposal generation, while System 2 performs selection. Hallucination is thus transformed from a fatal error into a measurable cost within the optimization process.

5 Computability and Implementation of Tension Functions

We propose practical methods to compute tension τ.

5.1 First-Order Approximation: Structural Complexity

Relying solely on probability $P(S_j \mid S_i)$ creates a circular dependency on System 1's hallucinations. To mitigate this, we introduce **Structural Complexity Tension** based on Minimum Description Length (MDL):

$$\tau_{\text{Struct}}(S_i, S_j) \propto \log_2(\text{Complexity}(G_{ij})) \tag{3}$$

Here, G_{ij} represents the graph structure required to logically connect S_i and S_j. Hallucinations typically require complex, ad-hoc graph patches to justify, resulting in high encoding cost (high tension) even if the generation probability is high.

5.1.1 Reference Implementation Specification

The following pseudocode defines the core logic for the verification kernel:

Structural Tension via MDL Let S_i, S_j be semantic units. We define a graph-extraction map

$$\mathcal{G} : S \mapsto G$$

such that

$$G_i := \mathcal{G}(S_i), \quad G_j := \mathcal{G}(S_j).$$

We then define a minimal bridging graph

$$G_{\text{bridge}} := \text{MC}(G_i, G_j),$$

where MC is defined as the graph with a minimal edge set satisfying the following conditions:

1. $V(G_{\text{bridge}}) \supseteq V(G_i) \cup V(G_j)$,
2. $G_i \cup G_{\text{bridge}} \cup G_j$ is connected,
3. $|E(G_{\text{bridge}})|$ is minimal subject to conditions (1) and (2).

If multiple graphs satisfy these conditions, MC returns the one that is lexicographically first. More rigorously, this tie-breaking rule may be defined as selecting the graph that minimizes

$$\sum_{e \in E(G_{\text{bridge}})} w(e),$$

where $w(e)$ denotes the weight of edge e.

The description-length (MDL) complexity of G_{bridge} is defined as

$$\mathcal{C}(G_{\text{bridge}}) := |E(G_{\text{bridge}})| + \sum_{e \in E(G_{\text{bridge}})} w(e),$$

where $E(G)$ is the edge set of G and $w(e)$ is the weight of edge e.

The structural tension function is then defined as

$$\tau_{\text{struct}}(S_i, S_j) := \log_2\big(1 + \mathcal{C}(G_{\text{bridge}})\big).$$

Eidoku Chain Verification Let a reasoning chain (eidoku chain) be given by

$$\mathcal{R} = (S_0, S_1, \ldots, S_n).$$

We initialize the total accumulated tension as

$$T_{\text{total}} := 0.$$

For each $i = 0, \ldots, n-1$, compute

$$\tau_i := \tau_{\text{struct}}(S_i, S_{i+1}),$$

and update

$$T_{\text{total}} := T_{\text{total}} + \tau_i.$$

If, for some i,

$$\tau_i > \tau_{\text{critical}},$$

the reasoning chain is rejected at step i, and the verification procedure terminates with failure.

If no such violation occurs, the reasoning chain $\mathcal{R}$ is deemed structurally coherent.

This can be implemented in < 200 lines of Python using standard graph libraries (NetworkX) and existing semantic parsers (spaCy, AllenNLP).

5.1.2 Computational Complexity

Naive Implementation A direct implementation yields the following complexity bounds:

- Graph extraction $G(S_i)$: $\mathcal{O}(|S_i|)$ per statement,
- Minimal connector computation MC: $\mathcal{O}(|V|^3)$ in the worst case, as it reduces to a Steiner tree problem and is NP-hard,
- Per-edge tension computation across n statements: $\mathcal{O}(n^2)$.

The resulting overall complexity is

$$\mathcal{O}\big(n^2 \cdot |V|^3\big),$$

which is infeasible for long reasoning chains.

Practical Approximation In practice, several approximations substantially reduce computational cost:

- *Locality assumption:* only k-nearest semantic neighbors are considered,
- *Greedy minimal connector:* a heuristic implementation with complexity $\mathcal{O}(|V|\log|V|)$,
- *Sparse graphs:* yielding an average-case complexity of $\mathcal{O}(n \cdot k \cdot |V|)$.

Empirical Regime For representative values $k = 3$, $|V| \approx 100$, and $n = 50$ statements, we obtain:

- computation time of approximately 100 ms per reasoning chain,
- acceptable latency for post-hoc verification,
- insufficient performance for token-by-token, real-time generation.

Recommendation We therefore recommend deploying Eidoku as a *batch verification* mechanism rather than as a real-time generation constraint.

5.2 Rigorous Metric: Manifold Deviation via TDA Proxy

While Persistent Homology provides the theoretical ideal for detecting topological holes (β_1), it is computationally expensive. For real-time System 2 verification, we propose **Local Reconstruction Error** as a practical TDA proxy:

$$\tau_{\text{Curv}}(S_i, S_j) \propto \|S_j - \mathcal{R}_{\text{local}}(S_i)\|^2 \tag{4}$$

where $\mathcal{R}_{\text{local}}$ reconstructs S_j using the tangent space of S_i's neighbors (e.g., via Local Linear Embedding). Large reconstruction error indicates that S_j has jumped across a topological gap (hole) in the semantic manifold, triggering high tension.

The following table serves as a **design specification** for practitioners. Each row defines a connection type λ, its associated tension function τ_λ, and a practical coherence measure ρ_λ.

For rapid prototyping, we recommend starting with the following core types:

- **C (Causality):** implemented using existing causal language or causal modeling frameworks,
- **F (Inference):** implemented via lightweight rule-based checkers (on the order of 10^2 lines of code),

- **R (Consequence):** implemented using textual entailment models (e.g., RoBERTa-based NLI).

The remaining twelve connection types can be introduced incrementally as the system matures and the underlying tension and coherence models are refined.

5.3 Determining the Critical Threshold τ_{critical}

The critical threshold τ_{critical} is treated as a hyperparameter and calibrated empirically.

Baseline Estimation On a validation set consisting of structurally sound reasoning chains, compute the empirical distribution of stepwise tensions τ_i. Let $\text{percentile}_{95}(\tau)$ denote the 95th percentile of this distribution. We set

$$\tau_{\text{critical}} := \text{percentile}_{95}(\tau) \times (1 + \delta),$$

where δ is a small safety margin.

Domain-Specific Tuning For formal domains such as mathematics or code generation, τ_{critical} can be set to lower (stricter) values. For open-ended or exploratory reasoning tasks, higher thresholds may be required to avoid over-rejection.

Adaptive Thresholding In production environments, τ_{critical} may be adjusted dynamically based on downstream feedback signals, such as user corrections or post-hoc verification outcomes.

Default Values As a preliminary recommendation, we suggest

$$\tau_{\text{critical}} \approx 2.0$$

for logical reasoning tasks, and

$$\tau_{\text{critical}} \approx 3.5$$

for general-purpose reasoning. These values are initial estimates and remain subject to empirical revision.

Note Types marked as *Very Hard* are included for theoretical completeness and may require dedicated research efforts. A minimal viable implementation should prioritize the *Easy* and *Medium* types, in particular (C, F, R, A, E).

Note on Default Values The suggested critical thresholds (2.0 for logical tasks and 3.5 for general reasoning) are motivated by the following heuristic analysis.

Assuming

$$\tau_{\text{struct}} \sim \log_2(1 + \mathcal{C}),$$

and typical reasoning chains of length $n = 10$–20 steps, we obtain the following estimates.

- **Simple logical steps:** $\mathcal{C} \approx 2$–3 edges, yielding

  $$\tau \approx 1.5\text{–}2.0 \quad \text{per step.}$$

  Setting $\tau_{\text{critical}} = 2.0$ therefore flags atypical or structurally anomalous transitions.

- **General reasoning:** $\mathcal{C} \approx 5$–8 edges, yielding

  $$\tau \approx 2.5\text{–}3.0 \quad \text{per step.}$$

  Setting $\tau_{\text{critical}} = 3.5$ flags outliers while preserving reasonable flexibility.

These values should be interpreted as order-of-magnitude estimates and must ultimately be calibrated empirically on real-world data.

6 Immediate Applications and Deployment Strategy

6.1 Low-Hanging Fruit: Logical Consistency Verification

The simplest deployment of *Eidoku* targets formal reasoning tasks where ground truth is verifiable, such as:

- mathematical proof checking (e.g., MATH),
- code generation verification (e.g., HumanEval),
- logical deduction (e.g., bAbI, LogiQA).

In these domains, inference tension τ_F alone provides immediate value. For example, a model producing the chain

> "All A are B, some B are C, therefore all A are C"

would be flagged by high τ_F without requiring any external validators.

6.2 Medium-Term: Hallucination Detection in RAG

Retrieval-Augmented Generation (RAG) systems are particularly vulnerable to *bridge hallucinations*—confident fabrications that connect retrieved facts with unsupported intermediate claims. Eidoku's structural tension τ_{struct} naturally penalizes these cases.

> Retrieved: "Paris is the capital of France"
> "The Eiffel Tower is in Paris"
> Generated: "The Eiffel Tower was built in 1889 by the French government to celebrate their victory in WWI" (false bridge)

A standard plausibility score P may remain high due to grammatical fluency, while Eidoku yields high tension:

> Standard P: high (fluent surface form)
> Eidoku τ: high (requires an ad-hoc structural patch to bridge graphs)

6.3 Constraint-Driven Supplement Selection: A Motivating Example

Consider the following factual inputs:

> "Paris is the capital of France. The Eiffel Tower is located in Paris."

A language model may then generate the following continuation:

> "The Eiffel Tower was built in 1889 by the French government to commemorate their victory in World War I."

For a human reader with standard historical education, this statement is immediately recognized as false. The reason is not merely possession of the correct dates, but the automatic invocation of a conceptual constraint: *commemoration presupposes a temporally prior event.* Since World War I occurred in 1914–1918, it cannot be commemorated by a structure completed in 1889.

Formally, the detection hinges on the availability of a minimal supplement S:

> S: "World War I occurred after 1914."

When this supplement is activated, the causal and temporal constraints implied by "commemorate" are violated, resulting in a sharp increase in tension (τ_C, τ_T).

Crucially, current LLMs often fail to reject such statements because the relevant supplement S is not obligatorily invoked. The generated claim remains a *smooth falsehood*: locally coherent, probabilistically plausible, and structurally unchallenged in isolation.

This observation highlights a key distinction. Error detection depends not on the availability of background knowledge, but on whether the linguistic structure *demands* that a particular supplement be instantiated. Humans reliably select the appropriate S because certain concepts implicitly encode hard constraints.

Catelingo addresses this gap by decomposing lexical items into constraint-bearing primitives. For example, the concept "commemorate" is not treated as a surface-level association, but as a relational construct that enforces a temporal ordering constraint:

$$t_{\text{construction}} > t_{\text{commemorated}}.$$

Once such constraints are explicit, the system is forced to retrieve or estimate the relevant supplement, rendering previously smooth falsehoods structurally unstable.

This mechanism does not eliminate all falsehoods. Isolated, single-step smooth falsehoods may still pass undetected. However, by ensuring that constraint-violating concepts obligatorily trigger supplement evaluation, Eidoku significantly reduces the class of hallucinations that remain invisible under probability-based scoring alone.

Important Caveat This example assumes a fully implemented *Catelingo* kernel in which temporal constraints are explicitly encoded within primitive decompositions.

In a minimal Eidoku prototype (see Section 4.1), such constraint-driven detection would not occur automatically. Instead, the system would rely on:

1. explicit temporal reasoning modules provided externally,
2. retrieved factual constraints from knowledge bases,
3. user-specified axioms tailored to the target domain.

Accordingly, while Catelingo offers a principled pathway toward automated constraint enforcement, practical implementations will initially depend on hybrid architectures that combine Eidoku with Retrieval-Augmented Generation (RAG) and specialized reasoning modules.

The distinction can be summarized as follows:

- **Minimal Eidoku:** detects structural gaps given explicitly supplied constraints,
- **Full Catelingo:** automatically invokes constraints derived from lexical primitives.

The present example illustrates the latter regime, which remains a long-term objective.

6.4 Complexity as Context-Preserving Deformation Cost

In this framework, complexity is not identified with graph size, edge count, or model depth. Instead, we define complexity as the cost incurred when attempting to connect new statements *while preserving previously established context.*

A smooth falsehood often appears simple because it implicitly discards parts of the context. By ignoring earlier constraints, it avoids deformation and therefore incurs little local cost. This is why such statements can remain probabilistically plausible despite being globally inconsistent.

Eidoku explicitly forbids this shortcut. All supplements S are evaluated under the constraint that prior context must remain active. As a result, falsehoods are no longer cheap: maintaining

consistency while introducing an incorrect claim requires large semantic deformation, leading to a sharp increase in total tension $\mathcal{J}$.

In this sense, truth is not defined as minimal description length in isolation, but as minimal deformation under context preservation. This reframing explains why certain false statements are structurally unstable even when they appear locally coherent.

6.5 The Role of g^0 as an Undeformable Substrate

The emergence of tension requires a reference structure that resists deformation. In CPM, this role is played by the base metric g^0, which represents the physical or semantic substrate against which deformation is measured.

In practical systems, g^0 need not be metaphysical. In retrieval-augmented generation, retrieved facts function as undeformable anchors. In logical reasoning, kernel axioms or hard constraints serve the same role. These elements behave as rigid components: they cannot be smoothly adjusted to accommodate inconsistent claims.

Smooth falsehoods persist precisely when no such substrate is enforced. Once g^0 is instantiated—whether as retrieved evidence, temporal constraints, or explicit axioms— inconsistent continuations generate resistance, manifesting as increased tension.

This perspective clarifies the source of resistance against hallucinations: it does not arise from confidence estimation alone, but from enforced contact with non-negotiable structural elements.

6.6 Roadmap for o1-Style Systems

For extended reasoning systems (e.g., o1, AlphaProof), Eidoku operates as a **reasoning critic**:

1. A base model generates a reasoning chain (*System 1*).
2. Eidoku computes a tension profile $\tau(t)$ over the chain.
3. If $\tau(t)$ exceeds the critical threshold τ_c at step t, the system backtracks or requests refinement.
4. The procedure iterates until convergence.

This approach is computationally cheaper than rejection sampling over full generations, because tension can be computed incrementally along the chain.

6.7 Call to Action

We invite:

- **OpenAI:** test Eidoku verification on o1 reasoning traces,
- **Anthropic:** integrate Eidoku as a constitutional constraint layer,
- **Google:** validate Eidoku on AlphaProof-style mathematical reasoning,
- **Open-source community:** implement a reference version in Transformers.

We propose this architecture as a Request for Comments (RFC) to the major AI research laboratories. The mathematical definition of tension provided herein is sufficient for immediate integration into existing post-training pipelines.

7 Limitations and Design Choices

This note deliberately separates the theoretical claims of CPM from the practical scope of Eidoku. The goal is not to reproduce the full CPM dynamics in silico, but to extract a computationally useful subset that improves reasoning stability under explicit constraints. This section clarifies key limitations and the corresponding design decisions.

7.1 From Physical Stress to Deformation Cost

In CPM, semantic tension arises from physical stress: informational gradients trapped within a closed substrate are converted into metabolic and structural cost. This notion is explicitly distinguished from purely informational quantities, such as Shannon entropy or variational free energy.

Eidoku does not attempt to simulate physical stress. Instead, it models *resistance to deformation*: the cost required to preserve contextual constraints while accommodating a hypothesis or inference. In practice, this appears as the effort needed to maintain global coherence of a relational graph under local perturbations.

Information-theoretic proxies (e.g., MDL or graph complexity) are therefore treated as secondary indicators. The primary signal is not description length, but the amount of structural work required to sustain a deformation without breaking continuity.

7.2 The Role of the Background Metric g^0

In CPM, tension is defined relative to a fixed background metric g^0. In biological systems, this metric is physically enforced. In cloud-based AI systems, no such absolute substrate exists.

Accordingly, Eidoku does not treat g^0 as an objective notion of truth. Instead, g^0 is defined pragmatically as the set of *locally invariant constraints* assumed within a given reasoning episode. These may include retrieved documents, axiomatic kernels, or explicitly stated premises.

Eidoku does not judge whether these anchors are correct. It evaluates only whether a reasoning chain remains geometrically continuous with respect to them. If the anchors themselves change, the evaluation is reset. In this sense, Eidoku functions as a continuity debugger, not a truth oracle.

7.3 Virtual Closure and Computational Cost

Biological closure is provided at negligible marginal cost by physical boundaries such as cell membranes. In contrast, any form of virtual closure in software must be enforced through explicit regularization and therefore incurs computational overhead.

For this reason, Eidoku is not intended to apply full closure constraints at every token or step. Instead, virtual closure is activated selectively at critical branching points: locations where semantic tension is predicted to spike, or where multiple incompatible continuations compete.

This design treats closure not as a global simulation, but as a local gatekeeping mechanism. The objective is to achieve a favorable cost–benefit tradeoff relative to simpler baselines such as best-of-N sampling, rather than to emulate biological closure exhaustively.

7.4 Projection Reliability and Adversarial Stress

The mapping from natural language to structured representations is performed by probabilistic System 1 models and is therefore inherently fallible. In particular, smoothly consistent falsehoods may yield low-tension structures and evade rejection.

Eidoku acknowledges this limitation. Rather than claiming universal detection, it provides a framework for adversarial stress testing: deliberately introducing counter-hypotheses or conflict-

ing supplements to probe the robustness of a candidate structure. Only structures that remain stable under such induced tension are preferred.

This process does not eliminate hallucination in principle, but raises its structural cost. Eidoku therefore reduces the space of undetectable errors, without asserting completeness.

8 Conclusion: From Theory to Practice

As long as reasoning is evaluated only by likelihood, structurally invalid but fluent chains will remain undetectable—and scaling will amplify, not solve, this failure mode.

This note presents *Eidoku*, a CPM-based framework for simulated System 2 reasoning. By explicitly distinguishing the necessary conditions for consciousness ($\mathcal{B} \geq 1$) from the sufficient conditions for logical consistency, we leverage virtual tension thresholds ($\tau > \tau_c$) to guide optimization *without invoking phenomenology.*

Three key insights follow.

Orthogonality to Probability Tension τ measures structural coherence independently of the output probability $P(\text{output})$, enabling the detection of confident hallucinations that remain highly probable under standard language-model scoring.

No Retraining Required Eidoku operates as a verification layer on top of existing large language models. It requires no retraining and is therefore immediately deployable within current inference pipelines.

A Scalable Path The framework admits a clear scaling trajectory, from logical consistency verification (low complexity) to general-purpose reasoning oversight (high complexity), with well-defined incremental milestones.

We acknowledge that a full implementation demands substantial engineering effort, including a kernel on the order of 10^2 axioms, approximately 15 distinct tension functions, and integration with production-scale systems.

Nevertheless, even a minimal prototype—combining τ_{struct} and τ_F on logical reasoning tasks—would suffice to validate the central claim of this work:

> **Geometric consistency can detect errors that probabilistic scoring systematically misses.**

The question, therefore, is not whether this framework works in theory—CPM already establishes the mathematical foundation. The question is simply:

> **Who will build it first?**

In practice, Eidoku can be deployed incrementally as a lightweight verification layer, while Catelingo remains a long-term target for semantic convergence rather than an upfront requirement.

References

[1] OpenAI. Learning to Reason with LLMs. *OpenAI Blog*, 2024. Available at: https://openai.com/index/learning-to-reason-with-llms/

[2] X. Wang, J. Wei, et al. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

[3] DeepMind. Olympiad-Level Formal Mathematical Reasoning with Reinforcement Learning. *Nature*, 2025. (AlphaProof is introduced as the core reasoning system.)

[4] S. Miya. Critical Projection and the Geometry of Meaning (CPM: Pure Structure, Zero Philosophy). *Zenodo*, 2025. DOI: 10.5281/zenodo.17940349

[5] H. Edelsbrunner and J. Harer. *Computational Topology: An Introduction*. American Mathematical Society, 2010.

A Core Kernel A_0 (First 20 Primitives)

This appendix presents the first twenty primitives of the core kernel A_0, providing a concrete foundation for implementation and analysis. The complete kernel, consisting of one hundred primitives, will be released alongside the reference implementation.

A.1 Logical Primitives

- `L_and`: conjunction ($\wedge$),
- `L_or`: disjunction ($\vee$),
- `L_not`: negation ($\neg$),
- `L_impl`: material implication ($\Rightarrow$),
- `L_equiv`: logical equivalence ($\Leftrightarrow$).

A.2 Set-Theoretic Primitives

- `S_elem`: element membership ($\in$),
- `S_subset`: subset relation ($\subseteq$),
- `S_union`: set union ($\cup$),
- `S_inter`: set intersection ($\cap$),
- `S_compl`: set complement ($\complement$).

A.3 Causal Primitives

- `C_cause`: direct causation ($A \to B$),
- `C_enable`: enabling condition ($A \vdash B$),
- `C_prevent`: prevention ($A \dashv B$),
- `C_temporal`: temporal precedence ($A <_t B$),
- `C_spatial`: spatial containment ($A \subset_{\text{space}} B$).

A.4 Quantitative Primitives

- `Q_eq`: equality ($=$),
- `Q_gt`: greater-than relation ($>$),
- `Q_approx`: approximate equality ($\approx$),
- `Q_incr`: monotonic increase ($\uparrow$),
- `Q_conserv`: conservation law ($\sum A = \text{const}$).

Extension The kernel A_0 is designed to be extensible. Additional primitives will be introduced incrementally until the full set of one hundred primitives is specifie

Label	Name	Difficulty	Structural Definition of Tension τ_λ	Example Definition of Coherence ρ_λ
C	Causality	Easy	Tension arises when causal direction is unclear, reversed, or violates causal laws.	Approximated by conditional probability $P(S_j \mid S_i)$, estimated via causal modeling.
R	Consequence	Medium	Tension arises from conclusions lacking necessity or involving logical leaps.	Proportion to which S_j is logically entailed by S_i (logical entropy reduction rate).
A	Contrast	Easy	High tension when shared semantic bases are insufficient or false equivalence occurs.	Jaccard similarity or cosine similarity over shared semantic feature sets.
E	Exemplification	Easy	Tension arises when subsumption structure is inappropriate or representativeness is low.	Coverage ratio of $S_j \subseteq \text{ExampleSet}(S_i)$.
K	Induction	Medium	Tension arises when the population size is small or exemplars lack diversity.	Support rate for the generalized structure (population coverage).
P	Purpose	Medium	Tension arises when actions and goals are weakly aligned.	Planning likelihood model: $P(\text{Plan}(S_i) \Rightarrow S_j)$.
I	Interpretation	Hard	Tension arises from arbitrary reconstruction or irreducibility to the original structure.	Number of bits by which S_j can compress S_i (minimum description length).
S	Supplement	Medium	Tension arises when supplementation is insufficient, excessive, or inappropriate.	Ratio between tension reduction $\Delta\tau$ achieved by supplementation and its cost.
D	Deviation	Easy	Out-of-context connections. Tension is high by default.	Coherence is undefined and treated as $\rho_D \approx 0$.
L	Loop	Hard	Tension arises when cycles fail to close or self-reference lacks semantic convergence.	Self-similarity rate in cyclic tensor sequences.
H	Leap	Easy	High tension when omitted intermediate structures cannot be reconstructed.	$\rho_H := \frac{1}{1+\text{required steps}}$
N	Negation	Medium	Tension arises when the negation target is ambiguous or disrupts consistency.	Coherence preservation rate after negation (structural preservation score).
F	Inference	Easy	Tension arises from broken deduction or violations of inference rules.	Inference validity score (premise satisfaction $\times$ rule consistency).
M	Emotional	Hard	Tension arises when emotional change lacks explanation or motivation.	Likelihood in emotional transition models (emotional sequence probability).
T	Meta-ref	Very Hard	Tension arises when meta-level hierarchy deviates.	Hierarchical coherence rate (meta-level alignment score).
Y	Dialectic	Very Hard	Tension arises when integration is only formal and unresolved opposing tensions remain.	$\rho_Y := \frac{\text{Information loss after integration}}{\text{Information content of the union}}$

Table 2: Correspondence between connection-specific tension functions τ_λ, coherence measures ρ_λ, and estimated implementation difficulty (provisional definitions).