

# DENSN: Dual-Pathway Neuro-Symbolic Optimization via Spectral Structure Learning

Liam O’Boyle

liamoboyle0@gmail.com

September 08, 2025

## Abstract

Conventional neuro-symbolic systems often operate under the constraint of a fixed ontology, where learning is restricted to parameter optimization within a static graph topology. This approach is inherently limited when faced with “out-of-distribution” logical contradictions—scenarios where the existing symbols are fundamentally insufficient to represent the domain. We present the **Dynamic Energy-Based Neuro-Symbolic Network (DENSN)**, a framework that treats logical frustration not as a terminal error, but as a generative signal for topological expansion.

DENSN introduces a **Dual-Pathway Optimization** mechanism: *Pathway A (Coherence)* minimizes free energy in low-tension regimes via spectral diffusion and modularity optimization, while *Pathway B (Frustration)* utilizes persistent logical paradoxes to trigger structural evolution—synthesizing new meta-symbols via Markov Blanket analysis and Exact/Approximate interface synthesis. This architecture replaces rigid static solvers with a **Spectral-MCM Hybrid Engine**, combining global spectral guidance via the Graph Laplacian with local stochastic refinement. We provide a rigorous Lyapunov stability analysis for the spectral relaxation and introduce an **Energetic Verification** protocol that mathematically decouples structural validity from semantic labeling, mitigating LLM-based hallucinations in neuro-symbolic concept formation. Our empirical validation on biomedical corpora and synthetic XOR paradoxes demonstrates that DENSN can autonomously navigate from localized controversies to hierarchical consensus, providing a scalable pathway for automated scientific discovery.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Core Formalism: The Dynamic Energy Landscape</b>	<b>3</b>
2.1	The Global Hamiltonian and Conflict Memory . . . . .	3
2.2	The Continuous Potential Field and Forcing Vector . . . . .	4
<b>3</b>	<b>Spectral-MCM Hybrid Dynamics</b>	<b>5</b>
3.1	Continuous Phase: Spectral Diffusion and Potential Flow . . . . .	5
3.2	Discrete Phase: Stochastic Collapse and WalkSAT Heuristic . . . . .	5
<b>4</b>	<b>Topological Structure Learning (TSL)</b>	<b>6</b>
4.1	Dual-Pathway Optimization: Modularity vs. Resolution . . . . .	6
4.2	The Abstraction Operator and Interface Synthesis . . . . .	7
<b>5</b>	<b>The Neuro-Symbolic Bridge: Semantic Verification</b>	<b>8</b>
5.1	Formalizing the Grounding Problem . . . . .	8
5.2	The Energetic Verification Protocol . . . . .	8
5.3	The Bridge Decision Rule . . . . .	8
5.4	Biomedical Case Study: Consensus vs. Controversy . . . . .	9
5.5	Synthetic Benchmark: The 3-Symbol XOR Paradox . . . . .	9
5.5.1	Section 6.2.1 Convergence Metrics . . . . .	10
<b>6</b>	<b>Conclusion</b>	<b>11</b>
<b>A</b>	<b>Proof of Spectral Stability</b>	<b>13</b>
A.1	A.1 Notation and Preliminaries . . . . .	13
A.2	A.2 Continuous Phase Stability (Lemma 1) . . . . .	13
A.3	A.3 Theorem 1: Lyapunov-Style Boundedness for the Hybrid Cycle . . . . .	13

# 1 Introduction

The central promise of Neuro-Symbolic Artificial Intelligence (NSA) is the integration of the robust, generalizable reasoning of symbolic logic with the adaptive, data-driven learning of neural architectures. Current paradigms—ranging from Differentiable Logics [2, 8] and Neural Probabilistic Logic [10] to the Neural-Symbolic Concept Learner [11]—have made significant strides in this direction. However, these systems often share a common fundamental vulnerability: they assume a predefined and fixed ontology or graph topology [14]. Whether the system is adjusting truth values in a fuzzy logic or probability masses in a factor graph, it is constrained by its initial ontological vocabulary.

In any complex, evolving knowledge domain—such as theoretical physics, molecular biology, or environmental science—new data frequently introduces contradictions that cannot be resolved within the existing symbolic framework. Standard computational logic treats such contradictions as failure states ( $\perp$ ) to be eliminated via backtracking or pruning. We hypothesize, however, that the capacity for genuine machine intelligence is bounded by the system’s ability to internalize and resolve contradiction through **Topological Expansion**.

We propose that a persistent structural paradox is not an error in calculation, but an *energetic signal* that the parameter space of the system is insufficient to describe reality. To address this, we present the **Dynamic Energy-Based Neuro-Symbolic Network (DENSN)**, which formalizes the transition from parameter optimization to structure learning through three core contributions:

1. **Dual-Pathway Optimization:** We propose that optimization is not a monolithic process but a dual-pathway mechanism. In high-consensus regimes, the system optimizes for *Coherence* (Modularity); in high-conflict regimes, it optimizes for *Resolution* (Structural Evolution).
2. **A Physics-Based Hybrid Engine:** We move beyond discrete heuristics by implementing a **Spectral-MCM Engine**. This engine utilizes the eigenvalues of the Graph Laplacian to define a global potential field, steering binary states through a spectrally-stable diffusion process before collapsing into discrete truth assignments.
3. **Interface Fidelity & Energetic Verification:** We formulate a rigorous test for the "Symbol Grounding Problem." By measuring the Hamiltonian response ( $\Delta\Psi$ ) to injected semantic labels, we provide a mathematical safeguard that ensures endogenously derived structural logic is consistent with exogenously retrieved human semantics.

In the following sections, we derive the mathematical foundations of the DENSN Hamiltonian, prove the stability of its hybrid dynamics, and demonstrate its efficacy in resolving complex medical controversies and synthetic SAT paradoxes.

## 2 Core Formalism: The Dynamic Energy Landscape

The DENSN framework operates on a dynamic bipartite graph  $\mathcal{G}(t) = (\mathcal{S}, \mathcal{C}, E)$  that mediates the relationship between atomic logical propositions and the constraints that govern them. Unlike static graphical models, the DENSN graph is an evolving manifold where both node states and edge topologies respond to logical pressure.

### 2.1 The Global Hamiltonian and Conflict Memory

The central quantity of the DENSN system is the Global Hamiltonian  $\Psi(t)$ , which serves as a macroscopic measure of logical frustration (total tension). We define the system state as a vector  $\mathbf{s} \in \{0, 1\}^n$ , where each component  $s_i$  represents the truth value of symbol  $S_i$ .

**Definition 1** (Weighted Violation Mass). *The Global Hamiltonian is defined as the sum of weighted binary violation states:*

$$\Psi(t) = \sum_{j=1}^m w_j(t) \cdot T_j(\mathbf{s}) \quad (1)$$

where  $T_j(\mathbf{s}) \in \{0, 1\}$  is a predicate function that evaluates to 1 if constraint  $C_j$  is violated under state  $\mathbf{s}$ , and 0 otherwise. This formulation aligns with the broader class of Energy-Based Models (EBM) as formalized by LeCun et al. [9].

In traditional SAT solvers or LTNetworks, the weights  $w_j$  are often treated as static priors. In DENSIN, we introduce **Conflict Cache Dynamics** ( $\mathcal{K}$ ), where constraint weights evolve temporally based on the persistence of their violation. This prevents the system from stagnating in low-energy limit cycles and effectively "pinches" paradoxes until they are forced into structural resolution.

$$w_j(t) = w_{\text{base}} \cdot (1 + \eta \cdot \text{persistence}_j(t)) \quad (2)$$

where  $w_{\text{base}}$  is the default constraint importance,  $\eta$  is the escalation rate (sensitivity to frustration), and  $\text{persistence}_j$  is a monotonic counter of consecutive epochs where  $C_j$  has remained violated.

**Remark 1** (The Saturation Plateau and  $\Psi_{\text{crit}}$ ). *To ensure structural bounded-behavior, we define  $w_{\text{max}}$  as the **Saturation Plateau** (internal parameter `mcm_max_weight_multiplier`). Escalation follows Equation (2) until one of two conditions is met:*

1.  $w_j \geq w_{\text{max}}$ : *The constraint reaches its maximum possible importance, becoming a "hard" boundary condition for the local cluster.*
2.  $\Psi_{\text{cluster}} \geq \Psi_{\text{crit}}$ : *The aggregate tension of the connected subgraph exceeds the critical threshold for **Frustration-Driven Learning (Pathway B)**, triggering immediate TSL abstraction.*

## 2.2 The Continuous Potential Field and Forcing Vector

While Equation (1) defines the discrete energy landscape, the inference engine requires a continuous representation to steer the state towards feasibility. We define the **Local Potential**  $\Phi(S_i)$  for each symbol as the aggregate pressure incident on that node:

$$\Phi(S_i) = \sum_{C_k \in \mathcal{N}(S_i)} w_k(t) \cdot T_k(\mathbf{s}) \quad (3)$$

This potential field maps the logical tension to a "Force Field" in  $\mathbb{R}^n$ . The optimization is then driven by the **Forcing Vector**  $\mathbf{b}_t$ , which represents the instantaneous gradient of the Hamiltonian:

$$\mathbf{b}_t = -\nabla_{\mathbf{s}} \Psi(\mathbf{s}, \mathbf{w}) \quad (4)$$

Note that because  $\mathbf{s}$  is discrete,  $\Psi$  is strictly piecewise differentiable; in implementation,  $b_{i,t}$  serves as a *Net Logical Torque*—the magnitude of the incentive to flip symbol  $S_i$  based on the local sensitivity of incident violated constraints. A minimal 3-symbol example of this machinery is detailed in Section 6.2.

### 3 Spectral-MCM Hybrid Dynamics

Optimization in a high-dimensional logical space is traditionally plagued by the presence of vast, low-utility plateaus and sharp, local energetic wells. Standalone SAT solvers, while efficient at discrete exploration, lack a global sense of the graph’s topological "flow." Conversely, continuous gradient methods (such as those in LTNs) often struggle with the rigid non-convexity of XOR-like constraints. DENSIN resolves this by decoupling the optimization into two distinct phases: a **Global Spectral Diffusion** phase for landscape smoothing, and a **Local Stochastic Collapse** phase for discrete satisfiability.

#### 3.1 Continuous Phase: Spectral Diffusion and Potential Flow

We define the weighted incidence matrix  $A \in \mathbb{R}^{m \times n}$  where  $A_{ji}$  represents the participation of  $S_i$  in  $C_j$ . The Graph Laplacian, which captures the topological connectivity and tension flow of the symbol-symbol projection, is defined as  $\mathcal{L} = A^\top W A$ , where  $W = \text{diag}(w_1, \dots, w_m)$  is the dynamic weight matrix from Section 2. This spectral representation is foundational in spectral graph theory [5] and ensures that the potential flow respects the algebraic connectivity of the knowledge graph [7].

The potential field  $\Phi$  evolves via a discrete-time diffusion process, analogous to Laplacian Eigenmaps in dimensionality reduction [3]. This process allows "logical potential" to flow from high-tension clusters to lower-tension regions, effectively smoothing the energy landscape and revealing the global structure of the problem.

$$\Phi_{t+1} = (I - \kappa \mathcal{L})\Phi_t + \mathbf{b}_t \quad (5)$$

The diffusion rate  $\kappa$  is a critical hyperparameter. If  $\kappa$  is too small, the system fails to propagate tension information globally; if too large, the system oscillates and violates the Lyapunov stability criterion. In the DENSIN implementation,  $\kappa$  is bounded by the reciprocal of the largest eigenvalue of  $\mathcal{L}$ , ensuring that the linear operator  $(I - \kappa \mathcal{L})$  remains a contraction for all non-zero frequency components.

**Implementation: Spectral Initialization.** The following Python implementation (Listing 1) demonstrates the dynamic calculation of  $\kappa$  using Arnoldi iterations to find the spectral radius of the graph.

#### 3.2 Discrete Phase: Stochastic Collapse and WalkSAT Heuristic

While spectral diffusion aligns the potential field with the global topology, the actual truth values  $s_i$  must remain discrete. The **Collapse Operator**  $\mathcal{O}_{collapse}$  periodically projects the continuous potential back into the boolean space.

This operator follows a modified WalkSAT logic [16]: it prioritizes flipping symbols with high local potential  $\Phi(S_i)$ , but introduces controlled stochastic noise to escape local minima. This ensures that the system does not get trapped in "logical vortices"—subgraphs where local optimization cannot reach a zero-tension state. Modern CDCL-based solvers like MiniSAT [6] utilize conflict-clause learning, but DENSIN relies on spectral smoothing to achieve global awareness before entering the discrete search phase.

The synergy between the continuous diffusion and discrete collapse constitutes the **Spectral-MCM Engine**. Diffusion provides the "physics-based" guidance, while collapse provides the "algorithmic-descent" into feasible states.

```

1 # densn/dynamics/relaxation.py
2 import scipy.sparse.linalg
3 import numpy as np
4
5 def initialize_dynamics(self, graph):
6     """
7     Initializes the diffusion rate kappa based on the spectral
8     properties of the graph Laplacian to ensure stability.
9     """
10    # 1. Construct the Laplacian L = A^T * W * A
11    L = self.build_laplacian(graph)
12
13    # 2. Compute the largest eigenvalue (Lambda_max)
14    # Theory: kappa < 2 / lambda_max for L2 stability.
15    eigenvals = scipy.sparse.linalg.eigsh(L, k=1, which='LM',
16                                          return_eigenvectors=False)
17    lambda_max = eigenvals[0]
18
19    # 3. Set diffusion rate kappa with safety factor
20    self.kappa = (2.0 / lambda_max) * 0.95
21
22    return self.kappa

```

Listing 1: Spectral Initialization Logic

---

**Algorithm 1** Hybrid Collapse Operator (WalkSAT-Inspired)

---

```

1: Input: Potential field  $\Phi$ , Noise probability  $p$ , Threshold  $\theta$ 
2: for each symbol  $S_i \in \mathcal{S}$  do
3:   if  $\Phi(S_i) > \theta$  then ▷ Symbol is under significant logical pressure
4:     Calculate  $\Delta\Psi_i = \Psi(\mathbf{s}$  with  $s_i$  flipped)  $- \Psi(\mathbf{s})$ 
5:     if Random(0, 1) <  $p$  then
6:        $s_i \leftarrow 1 - s_i$  ▷ Explore via noise (Heat injection)
7:     else if  $\Delta\Psi_i < 0$  then
8:        $s_i \leftarrow 1 - s_i$  ▷ Greedy Descent (Newton-like bit flip)
9:     end if
10:  end if
11: end for

```

---

## 4 Topological Structure Learning (TSL)

The most distinctive feature of the DENS architecture is its ability to transition from parameter optimization to **Topological Structure Learning (TSL)**. While the Spectral-MCM engine resolves tension within a fixed graph manifold, the TSL module is responsible for evolving that manifold itself. This evolution is not a random search but a targeted response to the system’s internal energy state.

### 4.1 Dual-Pathway Optimization: Modularity vs. Resolution

We propose that hierarchical learning follows a **Dual-Pathway** mechanism, where the trigger for structural change is based on the residual Hamiltonian  $\Psi$ .

- **Pathway A: Structure-Driven (Coherence Consolidation).**

Triggered when  $\Psi < \Psi_{\text{crit}}$  and  $\dot{\Psi} \approx 0$ . In this regime, the system is globally satisfiable but topologically "messy." DENS optimizes for **Model Parsimony** by identifying clusters of highly-correlated symbols that exhibit no internal conflict. This aligns with the Minimum

Description Length (MDL) principle [15], where the system seeks the most compressed representation of its symbolic state.

- **Pathway B: Frustration-Driven (Paradox Resolution).**

Triggered when  $\Psi \geq \Psi_{\text{crit}}$  and  $\dot{\Psi} \approx 0$ . This represents a **Frustrated Equilibrium**—an ontological failure where no bit-flipping combination can satisfy the existing constraints. DENSIN identifies the "Tension Hotspot" and abstracts it, a process analogous to non-monotonic Belief Revision in the AGM framework [1], where the system must expand its ontology to maintain consistency.

**Implementation: Dual-Pathway Selection.** The engine determines the pathway by examining the global and local tension maps before performing community detection (see Listing 2 for implementation details).

```

1 # densn/tsl/community.py
2
3 def find_clusters(self, graph, tension_map, total_tension):
4     """
5     Switches between Pathway A (Coherence) and Pathway B (Frustration)
6     based on the critical tension threshold PSI_CRIT.
7     """
8     PSI_CRIT = self.config.tsl_frustration_threshold
9
10    # PATHWAY B: Frustration-Driven (Resolve Paradoxes)
11    if total_tension > PSI_CRIT:
12        # Use tension density (w * T) as edge weights for clustering
13        return self.find_louvain(graph, edge_weights=tension_map)
14
15    # PATHWAY A: Structure-Driven (Consolidate Knowledge)
16    else:
17        # Use structural adjacency (binary connectivity) as weights
18        return self.find_louvain(graph, edge_weights="TOPOLOGY")

```

Listing 2: Dual-Pathway Community Detection

## 4.2 The Abstraction Operator and Interface Synthesis

When a cluster  $G_{\text{sub}}$  is identified for abstraction, DENSIN must synthesize an **Interface Function**  $f_{\text{meta}}$ . This function defines how the new meta-symbol  $S_{\text{meta}}$  interacts with the rest of the graph (its Markov Blanket).

The synthesis task is a search for the minimal logical representation that preserves the truth-table of the underlying cluster pins. To maintain scalability, DENSIN employs an **Adaptive Synthesis Pipeline**:

1. **Exact Synthesis (Quine-McCluskey):** For Markov Blankets with  $|I| \leq 6$ , the system performs a full prime-implicant reduction [4]. This guarantees the most compressed possible logical representation, minimizing the "Symbolic Tax" on the engine.
2. **Approximate Synthesis (Neural/Statistical):** For  $|I| > 6$ , the state space ( $2^{|I|}$ ) becomes intractable for exact methods. DENSIN instead trains a shallow Neural Network (or Decision Tree) to approximate the interface logic, trading absolute formal precision for computational tractability.

This ability to trade precision for speed allows DENSIN to scale linearly with problem size, a capability missing in traditional purely-symbolic logic engines.

## 5 The Neuro-Symbolic Bridge: Semantic Verification

A recurring challenge in autonomous neuro-symbolic systems is the "Semantic Hallucination" problem: the generation of symbolic abstractions that are structurally useful but semantically incoherent or factually incorrect. DENSIN mitigates this by maintaining a strict separation between **Structural Truth** (endogenous) and **Semantic Meaning** (exogenous), using the system’s own energy function as a verification gate.

### 5.1 Formalizing the Grounding Problem

For any synthesized meta-symbol  $S_{\text{meta}}$ , DENSIN possesses two independent definitions:

1. **Structural Realization** ( $f_{\text{meta}}$ ): The logic minimized from the Markov Blanket during TSL. This represents the \*functional\* reality of the cluster.
2. **Semantic Hypothesis** ( $L_{\text{meta}}$ ): A human-readable label or description proposed by a Large Language Model (LLM). This represents the \*intended\* meaning based on the conceptual centroid of the abstracted symbols.

Traditional concept-learning systems accept  $L_{\text{meta}}$  implicitly. In DENSIN,  $L_{\text{meta}}$  must survive an energetic audit.

### 5.2 The Energetic Verification Protocol

To verify a label, the system tentatively injects a bidirectional implication constraint—the **Verification Bridge**—between the structural function and the semantic label:

$$C_v : L_{\text{meta}} \longleftrightarrow f_{\text{meta}} \quad (6)$$

The system then measures the **Energetic Impact**  $\Delta\Psi$  of this new constraint on the Global Hamiltonian.

**Definition 2** (Verification Energy). *The Energetic Response of the system to a semantic hypothesis is defined as:*

$$\Delta\Psi = \Psi(\mathcal{G} \cup \{C_v\}) - \Psi(\mathcal{G}) \quad (7)$$

We define the **Energetic Verification Threshold**  $\epsilon_v$  (internal parameter `semantic_verification_threshold`, typically 0.1) as the maximum permissible tension increase.

### 5.3 The Bridge Decision Rule

The decision to accept a semantic label is governed by the following rule:

1. **Acceptance** ( $\Delta\Psi \leq \epsilon_v$ ): The semantic hypothesis introduces no significant contradiction into the system. The label is considered "Grounded" and is finalized.
2. **Rejection** ( $\Delta\Psi > \epsilon_v$ ): The label is inconsistent with the structural physics of the cluster (e.g., labeling a causal chain as a conflict). The hypothesis is rejected as a "Hallucination," and the system fallbacks to a generic structural identifier (`CLUSTER_ID`) until a valid hypothesis is provided.

This protocol ensures that DENSIN’s symbolic representation remains "truthful" to its underlying logical state, even when using black-box neural models.

We evaluated the DENSIN framework across two distinct experimental paradigms: a large-scale real-world biomedical knowledge extraction task and a controlled synthetic benchmark designed to probe the limits of structural paradox resolution. The current implementation is a single-machine prototype developed in Python, utilizing NumPy and SciPy for spectral operations.



## 5.4 Biomedical Case Study: Consensus vs. Controversy

The first validation phase focused on the system’s ability to navigate high-dimensional concept spaces derived from raw scientific literature. We selected the clinical debate regarding the survival efficacy of *Cisplatin* versus *Carboplatin* in ovarian cancer treatment as our primary test case.

**Graph Complexity.** We queried PubMed with the string “cisplatin carboplatin ovarian cancer survival” and ingested the top 10 primary research abstracts. The resulting graph contained **579 atomic symbols** and **922 logical constraints**. A subset of **116 constraints** corresponding to canonical oncological facts (e.g., basic pharmacological and survival relations) was locked as a Tier-1 “Sensory Boundary” to prevent semantically implausible flips of established medical knowledge.

**Inference Summary.** In this biomedical run, the Spectral-MCM engine executed three full relaxation cycles. The recorded global tension remained numerically constant ( $\Psi_{\text{init}} = \Psi_{\text{final}} = 11.22$ ), while the residual ratio emitted by the engine was 0.0388 (approximately 3.9%). Internally, the system identified exactly one XOR-class conflict core corresponding to mutually exclusive superiority claims between Carboplatin and Cisplatin regimens. The engine reported:

- **High-consensus background:** The system identified no persistent unsatisfied cores at the configured thresholds outside of the primary conflict cluster. Spectral diffusion reorganized the continuous potential field into coherent clusters corresponding to clinical trial cohorts, chemotherapy regimens, and toxicity profiles.
- **Localized controversy:** A single high-tension subgraph around comparative efficacy (Carboplatin vs. Cisplatin) persisted as a “singular contradiction” despite multiple hybrid cycles, marking it as a candidate for Pathway B (Frustration-Driven) abstraction.

The system computed a final implementation-defined **Confidence Score of 90.0** and categorized the overall **Controversy Level** as **Consensus**. These summary metrics are derived from the run artifact (JSON log) and should be interpreted as diagnostics: the low residual ratio and high confidence signal that the knowledge base is predominantly coherent, with a single encapsulated debate rather than pervasive inconsistency.

## 5.5 Synthetic Benchmark: The 3-Symbol XOR Paradox

To rigorously test the Conflict Cache escalation (Equation (2)), the spectral diffusion stability, and the TSL resolution logic in a fully controlled environment, we constructed a minimal ontology-insufficient benchmark with three symbols.

**Paradox Topology.** We define three symbols  $S_1, S_2, S_3$  with the following constraints:

- $C_1 : S_1 \rightarrow S_2$  (implication);
- $C_2 : S_2 \rightarrow S_3$  (implication);
- $C_3 : S_1 \oplus S_3$  (XOR:  $S_1$  and  $S_3$  cannot share the same state).

To enforce a structurally irreducible contradiction, we fix  $S_1 = 1$  as an immutable observation (Sensory Boundary). Under this configuration, no assignment to  $(S_2, S_3)$  can satisfy all three constraints simultaneously: any attempt to enforce  $C_1$  and  $C_2$  will ultimately violate  $C_3$ .

**Engine Log and Conflict Escalation.** The DENS engine is initialized with  $(S_1, S_2, S_3) = (1, 1, 1)$  and initial tension  $\Psi_{\text{init}} = 1.0$ . The run proceeds for 22 cycles, while the Conflict Cache repeatedly identifies a persistent singularity (one structurally irreducible violation) and escalates its weight according to Equation (2). The log records:

- **Persistent singularities:** At epochs 3, 6, 10, 13, 17, and 21 the engine reports “1 persistent singularities detected” followed by “Eq 2 Escalation: Increased weights of 1 constraints”, indicating repeated weight amplification of the same violated constraint.
- **Hybrid MCM attempts:** Fifteen MCM attempts are executed, alternating between `stochastic_flip`, `constraint_reweight`, `weak_constraint_prune`, and `random_restart`. Several attempts temporarily escape local minima (`escaped=True`) but none eliminate the singular violation.

This behavior is the intended “pinching” effect: paradoxical constraints become progressively heavier in the Hamiltonian until they force a structural response rather than being absorbed by further bit flips.

### 5.5.1 Section 6.2.1 Convergence Metrics

The synthetic run emits an explicit convergence summary:

- **Result Converged:** True;
- **Total Cycles:** 22;
- **Initial / Final Tension:**  $\Psi_{\text{init}} = 1.0000$ ,  $\Psi_{\text{final}} = 0.0000$ ;
- **TSL Events:** 1 (creation of `META_1`);
- **Flips:** 604; **Constraint evaluations:** 2,554 (2,415 served from the Conflict Cache, implying an 85.6% cache rate);
- **Wall-clock time:** 1,063.3 ms on the test machine.

Table 1 reports the trajectory of global tension  $\Psi$  and quadratic smoothness  $Q$  (topological mass) across cycles. Until the TSL event,  $\Psi$  increases in a staircase pattern as the violated constraint is repeatedly up-weighted, while  $Q$  reflects the reshaping of the potential field under escalating local tension.

We observe three distinct phases:

1. **Baseline stagnation (cycles 0–5):**  $\Psi$  remains at 1.0 and  $Q$  at 6.0, indicating that Spectral-MCM is unable to find a tension-free configuration within the fixed ontology.
2. **Escalation regime (cycles 6–20):** As persistence accumulates in the Conflict Cache, the weight of the singular violated constraint increases, driving  $\Psi$  from 1.0 up to 16.0 and inflating  $Q$  from 6.0 to 96.0. The diffusion process propagates this localized tension through the graph, raising the overall topological mass.
3. **Resolution via TSL (cycle 21):** Once the local cluster energy exceeds the frustration threshold  $\Psi_{\text{crit}}$ , the engine reports a Level-3 TSL event and synthesizes a meta-symbol `META_1` to encapsulate the XOR paradox. A “hormonal reset” then reinitializes the escalated constraint’s weight to baseline in the new topology, resulting in  $\Psi_{\text{final}} = 0.0$  and a non-zero  $Q$  (25.0) corresponding to the reconfigured potential field on the expanded graph.

This synthetic benchmark confirms the intended semantics of the DENS architecture: in genuinely ontology-insufficient settings, the Conflict Cache does not merely produce exploding energies, but instead drives the system into a controlled topological expansion that restores a zero-tension ground state at the cost of an enriched ontology.

Table 1: Synthetic benchmark trajectory: tension  $\Psi$  and smoothness  $Q$ .

Cycle	$\Psi$ (Tension)	$Q$ (Topo Mass)	Status
0	1.0000	6.0000	
1	1.0000	6.0000	
2	1.0000	6.0000	
3	1.6000	6.0000	
4	1.6000	6.0000	
5	1.6000	6.0000	
6	3.0000	6.0000	
7	3.0000	6.0000	
8	3.0000	6.0000	
9	3.0000	6.0000	
10	6.6000	15.3600	
11	6.6000	15.3600	
12	6.6000	15.3600	
13	10.8000	34.5600	
14	10.8000	34.5600	
15	10.8000	34.5600	
16	10.8000	34.5600	
17	16.0000	96.0000	
18	16.0000	96.0000	
19	16.0000	96.0000	
20	16.0000	96.0000	
21	0.0000	25.0000	TSL event

## 6 Conclusion

DENSN reframes logical contradiction from a failure mode to a generative resource. By coupling spectral dynamics with a dual-pathway structural learning mechanism, we provide a mathematically grounded framework for neuro-symbolic evolution.

*Limitations & Scope:* While DENSN provides a rigorous framework for continuous-discrete optimization, its performance is subject to the **Epistemic Horizon** of the initial extraction. If a paradox is fundamentally underspecified in the base corpus, TSL will generate a valid meta-symbol that remains "semantically thin" until external data foraging (PDEF) is activated. Furthermore, the  $O(N^3)$  complexity of spectral initialization for large dense graphs necessitates the use of localized spectral clustering (as implemented in `spectral_mode="LOCAL_ONLY"`), which may skip global topological features in favor of neighborhood-level resolution.

DENSN demonstrates that Intelligence is the minimization of a hybrid potential: the weighted sum of **Logical Frustration** (Error) and **Topological Complexity** (Structure). By defining this potential spectrally, we allow the system to self-organize not just to solve paradoxes, but to build the most efficient representation of its reality—whether that reality is a messy environmental debate or a settled medical treatment.

## References

- [1] Alchourrón, C. E., Gärdenfors, P., & Makinson, D. (1985). On the Logic of Theory Change: Partial Meet Contraction and Revision Functions. *The Journal of Symbolic Logic*, 50(2), 510–530.
- [2] Badreddine, S., Garcez, A. d., Serafini, L., & Spranger, M. (2022). Logic Tensor Networks. *Artificial Intelligence*, 303, 103649.
- [3] Belkin, M., & Niyogi, P. (2003). Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, 15(6), 1373–1396.
- [4] Brayton, R. K., et al. (1984). *Logic Minimization Algorithms for VLSI Synthesis*. Springer Science & Business Media.
- [5] Chung, F. R. K. (1997). *Spectral Graph Theory*. CBMS Regional Conference Series in Mathematics, No. 92.
- [6] Een, N., & Sörensson, N. (2003). An Extensible SAT-solver. In *International Conference on Theory and Applications of Satisfiability Testing* (pp. 502–518). Springer.
- [7] Fiedler, M. (1973). Algebraic Connectivity of Graphs. *Czechoslovak Mathematical Journal*, 23(2), 298–305.
- [8] Garcez, A. d., Besold, T. R., & de Raedt, L. (2019). Neural-symbolic computing: An effective methodology for principled integration of machine learning and reasoning. *Journal of Applied Logic*, 35, 1–32.
- [9] LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., & Huang, F. (2006). A Tutorial on Energy-Based Learning. *Predicting Structured Data*. MIT Press.
- [10] Manhaeve, R., et al. (2018). DeepProbLog: Neural Probabilistic Logic Programming. In *Advances in Neural Information Processing Systems* (pp. 3749–3759).
- [11] Mao, J., et al. (2019). The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences from Natural Supervision. In *International Conference on Learning Representations*.
- [12] Ozols, R. F., et al. (2003). Phase III Trial of Carboplatin and Paclitaxel Compared with Cisplatin and Paclitaxel in Patients with Optimally Resected Stage III Ovarian Cancer. *Journal of Clinical Oncology*, 21(17), 3194–3200.
- [13] Priest, G. (2006). *In Contradiction: A Study of the Transconsistent*. Oxford University Press.
- [14] Richardson, M., & Domingos, P. (2006). Markov Logic Networks. *Machine Learning*, 62(1-2), 107–136.
- [15] Rissanen, J. (1978). Modeling by Shortest Data Description. *Automatica*, 14(5), 465–471.
- [16] Selman, B., Kautz, H. A., & Cohen, B. (1994). Noise Strategies for Improving Local Search. In *AAAI* (Vol. 94, pp. 337–343).

## A Proof of Spectral Stability

In this appendix, we provide a derivation of a sufficient stability condition for the spectral diffusion component, and a Lyapunov-style boundedness argument for the hybrid cycle.

### A.1 A.1 Notation and Preliminaries

Let  $L$  be the graph Laplacian, a real symmetric positive semidefinite matrix. By the Spectral Theorem,  $L$  has an orthonormal eigenbasis  $\{u_1, \dots, u_n\}$  with corresponding eigenvalues  $0 = \lambda_1 < \lambda_2 \leq \dots \leq \lambda_{\max}$ .

The potential field  $\Phi_t$  can be expanded in this basis as  $\Phi_t = \sum_{i=1}^n c_{i,t} u_i$ . The **Quadratic Field Energy** (smoothness) is defined as  $Q(\Phi_t) = \Phi_t^\top L \Phi_t = \sum_{i=1}^n \lambda_i c_{i,t}^2$ .

### A.2 A.2 Continuous Phase Stability (Lemma 1)

**Lemma 1.** *For the diffusion update  $\Phi_{t+1} = (I - \kappa \mathcal{L}) \Phi_t$ , if  $0 < \kappa < 2/\lambda_{\max}$ , then  $Q(\Phi_{t+1}) \leq Q(\Phi_t)$ , with strict inequality if  $\Phi_t$  contains any non-zero frequency components.*

*Proof.* Consider the update in the spectral domain. Applying  $(I - \kappa \mathcal{L})$  to the basis expansion:

$$\Phi_{t+1} = \sum_{i=1}^n c_{i,t} (I - \kappa \mathcal{L}) u_i = \sum_{i=1}^n c_{i,t} (1 - \kappa \lambda_i) u_i \quad (8)$$

The new spectral coefficients are  $c_{i,t+1} = (1 - \kappa \lambda_i) c_{i,t}$ . The ratio of energy in each mode  $i$  is:

$$\frac{\lambda_i (1 - \kappa \lambda_i)^2 c_{i,t}^2}{\lambda_i c_{i,t}^2} = (1 - \kappa \lambda_i)^2 \quad (9)$$

For monotonic decrease, we require  $(1 - \kappa \lambda_i)^2 < 1$  for all  $\lambda_i > 0$ . This implies  $-1 < 1 - \kappa \lambda_i < 1$ , or  $0 < \kappa \lambda_i < 2$ . Since  $\lambda_i \leq \lambda_{\max}$  for all  $i$ , the condition  $0 < \kappa < \frac{2}{\lambda_{\max}}$  guarantees that the contraction factor is strictly less than 1 for all non-zero modes. Thus,  $Q(\Phi_{t+1}) < Q(\Phi_t)$ .  $\square$

### A.3 A.3 Theorem 1: Lyapunov-Style Boundedness for the Hybrid Cycle

**Theorem 1.** *Define the hybrid Lyapunov candidate  $V(\Phi, T) = \Psi(T) + \alpha Q(\Phi)$ . Under the bounded forcing assumption ( $\mathbf{b}_t^\top \mathcal{L} \mathbf{b}_t \leq B$ ) and a bounded flip-impact assumption, there exists a weighting factor  $\alpha$  such that the full hybrid cycle yields a negative expected drift in  $V$  outside a bounded set.*

*Proof.* Let the cycle consist of one diffusion step followed by one collapse step. The change in energy is  $\Delta V = \Delta \Psi + \alpha \Delta Q$ .

1. **Collapse Step:** As per Algorithm 1, the collapse operator accepts a flip only if  $\Delta \Psi < 0$  (Greedy) or via noise. In the deterministic limit,  $\Delta \Psi \leq -\delta$  for some minimum discrete energy quantum  $\delta$ . The change in  $Q$  due to a single bit flip is bounded by a constant  $M$  for finite graphs.

2. **Diffusion Step:** By Lemma 1, diffusion reduces  $Q$  by a factor  $\epsilon > 0$ .<sup>1</sup>  $\Psi$  is invariant under diffusion as symbol states  $T$  are fixed.

Total expected change is bounded by:  $\Delta V \leq -\delta + \alpha(M - \epsilon Q)$ . By choosing  $\alpha < \frac{\delta}{M}$ , we ensure  $\Delta V < 0$  for all  $Q$  above a finite threshold. Thus, the system exhibits Lyapunov-style boundedness.  $\square$

<sup>1</sup>While Lemma 1 guarantees a reduction, the specific magnitude of  $\epsilon$  is dependent on the spectral decomposition of  $\Phi_t$ ; smooth fields dominated by low-frequency components exhibit slower decay than highly non-smooth residual fields.