

# Self-Organizing Octonionic Tries: Algebraic Memory Without Gradient Descent

Antonio Escalera

March 2026

Working Draft

## Abstract

We propose a self-organizing hierarchical memory structure in which octonionic algebra replaces gradient-based learning entirely. The structure, an **octonionic trie**, is a dynamically growing tree whose nodes are individual octonions. Routing through the trie is governed by Fano plane subalgebra decomposition, growth is triggered by the associator (a measure of algebraic incompatibility), updates are performed by octonionic composition, and consistency is verified via algebraic inversion. The trie monitors its own structural health through invariants derived from the algebra: compression efficiency, subalgebra decomposition cleanliness, composition error bounds, and prediction consistency.

The result is a memory system that organizes, grows, consolidates, and self-corrects using only the algebraic properties of the octonions, without weight matrices, loss functions, or backward passes. The octonionic algebra provides the routing structure (7 quaternionic subalgebras via the Fano plane), the novelty detection mechanism (the associator), the update rule (norm-preserving multiplication), the consistency check (algebraic inversion), and the structural health metrics (associator norms, composition error). These are typically five separate engineered components in conventional memory-augmented architectures; here they emerge from a single algebraic substrate.

This work builds on the mathematical foundations established in the companion thesis on octonionic neural networks [Escalera, 2026], extending the algebraic framework from gradient-trained networks to self-organizing structures that operate without optimization.

## Contents

<b>1</b>	<b>Introduction and Motivation</b>	<b>4</b>
1.1	Relationship to Companion Work . . . . .	4
<b>2</b>	<b>Related Work</b>	<b>4</b>
2.1	Octonionic Algebra and Computational Applications . . . . .	5
2.2	Hypercomplex Neural Networks . . . . .	5
2.3	Memory-Augmented Neural Architectures . . . . .	6
2.4	Self-Organizing Systems Without Gradient Descent . . . . .	7
2.5	Catastrophic Forgetting and Continual Learning . . . . .	8
2.6	Hierarchical and Hyperbolic Representations . . . . .	8
2.7	Decision Trees and Tree-Based Classification . . . . .	9
2.8	Reversible and Invertible Computation . . . . .	10
2.9	Equivariant Architectures and Symmetry Groups . . . . .	10
2.10	Positioning of the Octonionic Trie . . . . .	11

<b>3</b>	<b>The Associator as a Computational Primitive</b>	<b>11</b>
3.1	Definition and Algebraic Properties . . . . .	11
3.2	Computational Interpretation . . . . .	12
3.3	The Associator vs. Inner Product . . . . .	12
<b>4</b>	<b>Architecture of the Octonionic Trie</b>	<b>12</b>
4.1	Structure . . . . .	12
4.2	Encoding: From Raw Data to Octonions . . . . .	13
4.2.1	Hierarchical Projection . . . . .	13
4.3	Core Operations . . . . .	14
4.3.1	Compose: Routing and Update . . . . .	14
4.3.2	Detect: Novelty and Branching . . . . .	14
4.3.3	Ruminate: Consistency Verification . . . . .	15
4.3.4	Consolidate: Pruning and Merging . . . . .	15
4.4	Algebraic Residuals . . . . .	15
4.5	Memory Buffer and Relevance . . . . .	15
<b>5</b>	<b>Structural Invariants</b>	<b>16</b>
5.1	Composition Error Bound . . . . .	16
5.2	Compression Efficiency . . . . .	16
5.3	Subalgebra Decomposition Cleanliness . . . . .	16
5.4	Prediction Consistency . . . . .	17
5.5	Associator Health . . . . .	17
<b>6</b>	<b>Properties of the Architecture</b>	<b>17</b>
6.1	No Gradient Computation Required . . . . .	17
6.2	Encoder Invariance of Trie Structure . . . . .	17
6.3	Information Preservation . . . . .	18
6.4	Natural Hierarchy . . . . .	18
<b>7</b>	<b>Experimental Validation</b>	<b>18</b>
7.1	Prerequisite 1: Subalgebra Routing Discriminability . . . . .	18
7.2	Prerequisite 2: Associator as Novelty Signal . . . . .	18
7.3	Prerequisite 3: Composition Depth vs. Information Retention . . . . .	19
7.4	Prototype Trie: Stability-Plasticity Test . . . . .	19
7.5	Key Design Findings . . . . .	20
<b>8</b>	<b>Open Questions</b>	<b>20</b>
8.1	Geometric Embedding of the Trie . . . . .	20
8.2	Composition Order . . . . .	20
8.3	Adaptive Thresholds . . . . .	20
8.4	Scalability and Capacity . . . . .	20
8.5	Baselines . . . . .	21
<b>9</b>	<b>Adaptive Thresholds and Self-Organization</b>	<b>21</b>
9.1	Associator Norm Distribution on $S^7$ . . . . .	21
9.2	Fano Plane Geometry and Subalgebra Routing . . . . .	22
9.3	Global Threshold Justification . . . . .	23
9.4	$G_2$ Symmetry and Threshold Invariance . . . . .	24
9.5	Stability-Plasticity Tradeoff . . . . .	25
9.6	Complexity Analysis . . . . .	25
9.7	Self-Organization Narrative . . . . .	26

9.8	Convergence of Meta-Trie Feedback . . . . .	26
<b>10</b>	<b>MNIST Benchmark</b>	<b>27</b>
10.1	Experimental Setup . . . . .	27
10.2	Effect of Encoder Quality . . . . .	28
10.3	Effect of Octonionic Dimensionality . . . . .	28
10.4	Significance . . . . .	29
<b>11</b>	<b>Conclusion</b>	<b>29</b>

# 1 Introduction and Motivation

Memory-augmented neural architectures (Neural Turing Machines [Graves et al., 2014], Differentiable Neural Computers [Graves et al., 2016], Memory Networks [Weston et al., 2015]) address a fundamental limitation of feedforward networks: the inability to store and retrieve structured information beyond what is captured in fixed weight matrices. These architectures introduce external memory modules with learned read/write controllers, enabling the network to maintain and manipulate an explicit memory state.

However, existing memory architectures share a common design pattern: the memory structure (flat array, matrix, or graph) is fixed a priori, and a trained controller learns to use it. The controller is optimized via gradient descent, the memory layout is predetermined, and the system cannot grow or restructure its memory in response to the data it encounters.

This paper proposes a fundamentally different approach. Rather than training a controller to manage a fixed memory, we construct a memory that *organizes itself* using the algebraic properties of the octonions. The memory is a tree (trie) whose nodes are individual octonions. The tree grows when it encounters novel structure, updates by algebraic composition, verifies its own consistency via inversion, and prunes itself when structure becomes redundant. No component of this system requires gradient-based optimization.

The key insight is that the octonionic algebra provides, through its intrinsic mathematical structure, all the mechanisms that memory-augmented architectures typically engineer separately:

1. **Routing** via Fano plane subalgebra decomposition (7 quaternionic channels).
2. **Novelty detection** via the associator  $[a, b, c] = (ab)c - a(bc)$ .
3. **Content update** via norm-preserving octonionic multiplication.
4. **Consistency verification** via algebraic inversion ( $x^{-1} = \bar{x}/|x|^2$ ).
5. **Structural health monitoring** via associator norms, composition error, and subalgebra coherence.

These five mechanisms are not independent modules bolted together; they are consequences of the same algebraic structure. This economy of mechanism is the central argument for the octonionic trie.

## 1.1 Relationship to Companion Work

This paper depends on and extends the foundations established in Escalera [2026]:

- **Algebraic foundations** (Sections 1–3 of the companion): Hurwitz’s theorem, octonionic multiplication, the Fano plane,  $G_2$  automorphisms, the density argument.
- **Reversibility thesis** (Section 2): Algebraic invertibility enabling backward reasoning. The octonionic trie uses inversion for consistency checking (“rumination”).
- **Numerical stability** (Phase 4 experimental results): Composition error bounds that determine the trie’s maximum depth.
- **Associator and subalgebra analysis** (Phase 9): Whether the associator carries useful information and whether subalgebras specialize. These are prerequisites for the trie’s routing and novelty detection mechanisms.

The companion thesis validates these properties for gradient-trained octonionic networks. This paper proposes that the same properties support a qualitatively different computational paradigm: self-organization without optimization.

## 2 Related Work

The octonionic trie draws on and departs from several distinct research threads: the algebraic theory of octonions, hypercomplex neural networks, memory-augmented architectures,

self-organizing systems, continual learning, and hierarchical representations. This section surveys each thread and identifies the gap that the octonionic trie occupies.

## 2.1 Octonionic Algebra and Computational Applications

The octonions were discovered by Graves (1843) and independently by Cayley (1845) as the unique 8-dimensional normed division algebra. The foundational algebraic theory is developed in Schafer [1966], with geometric treatments in Conway and Smith [2003], Springer and Veldkamp [2000], and Dray and Manogue [2015]. Baez [2002] provides the definitive modern survey, establishing the octonions’ position in the Hurwitz classification  $(\mathbb{R}, \mathbb{C}, \mathbb{H}, \mathbb{O})$  and their connections to exceptional Lie groups, projective geometry, and string theory. Okubo [1995] develops the physics applications systematically.

The octonionic algebra’s relevance extends beyond pure mathematics. Günaydin and Gürsey [1973] showed that the quark structure of the Standard Model can be naturally expressed in octonionic terms, launching a research program that continues through Dixon [1994], Furey [2016, 2018], Todorov and Drenska [2018], and Boyle [2020]. Furey’s work is particularly relevant: it demonstrates that the algebra  $\mathbb{R} \otimes \mathbb{C} \otimes \mathbb{H} \otimes \mathbb{O}$  acting on itself yields ideals that mirror one generation of quarks and leptons, suggesting that the octonionic algebraic structure has a natural affinity for organizing representations with rich internal symmetry. Baez [2012] examines the role of the three associative division algebras  $(\mathbb{R}, \mathbb{C}, \mathbb{H})$  in quantum theory via Dyson’s “three-fold way,” notably excluding the octonions: their non-associativity prevents the construction of a standard Hilbert space formulation, making them the unique division algebra that falls outside this classification.

The Fano plane, the finite projective plane  $\text{PG}(2, 2)$  with 7 points and 7 lines, encodes the octonionic multiplication table. Sevenne [2013] recovers the complete multiplication table from a regular tessellation of the equilateral torus (Heawood’s map), providing a topological perspective on the combinatorial structure underlying octonionic multiplication. The automorphism group of the octonions is the exceptional Lie group  $G_2$ , surveyed accessibly by Agricola [2008] and treated in depth by Harvey [1990]. The  $G_2$  invariance of the associator (section 9.4) constrains the functional form of adaptive threshold policies.

Computational implementations of octonionic operations face the challenge of non-associativity: standard matrix multiplication is associative, so octonionic multiplication cannot be faithfully represented by matrix products. Tian [2000] develops pseudo-representations via quaternionic matrix embeddings, providing a bridge to conventional linear algebra libraries. This challenge is circumvented in our implementation by computing octonionic products directly from the structure constants tensor  $C_{ijk}$ .

## 2.2 Hypercomplex Neural Networks

Neural networks over hypercomplex algebras form a progression of increasing algebraic richness. **Complex-valued networks** [Hirose, 2012, Trabelsi et al., 2018] extend real-valued networks to  $\mathbb{C}$ , capturing both amplitude and phase information with improved performance on signal processing tasks. **Quaternion networks** [Gaudet and Maida, 2018, Zhu et al., 2018, Parcollet et al., 2019, Xu and Mandic, 2015] extend to  $\mathbb{H}$ , achieving up to  $4\times$  parameter reduction through the Hamilton product’s weight-sharing structure [Parcollet et al., 2020] and natural handling of 3D rotations. Parcollet et al. [2020] surveys the quaternion network landscape; Bill and Cox [2024] analyzes their loss surfaces.

**Octonion networks** remain sparse. Popa [2016] first derived gradient descent for octonion-valued feedforward networks, addressing non-associativity via Cayley-Dickson decomposition. Wu et al. [2020] developed deep octonion networks with octonionic convolution and batch normalization. Saoud and Ghorbani [2020] introduced metacognitive learning for octonion networks applied to time series forecasting. All three approaches use the octonions as a representation

space within a gradient-trained architecture; none explicitly exploits the non-associativity — specifically the associator — as a signal for routing or novelty detection.

**Clifford algebra networks** generalize hypercomplex representations to arbitrary dimension and signature. Ruhe et al. [2023] constructs Clifford group equivariant layers achieving  $O(n)$ - and  $E(n)$ -equivariance (the symmetry implications are discussed further in section 2.9); Brandstetter et al. [2023] applies Clifford layers to PDE modeling; Brehmer et al. [2023] builds a geometric algebra Transformer with 16-dimensional multivector hidden states. Zhang et al. [2021] and Grassucci et al. [2022] develop parameterized hypercomplex layers that learn multiplication rules from data, achieving  $1/n$  parameter reduction for arbitrary  $n$ -dimensional algebras.

Comminiello et al. [2024] provides a comprehensive theoretical framework explaining *why* hypercomplex networks outperform real-valued networks: the algebra product enforces inter-channel correlation and weight sharing as inductive biases. This analysis applies directly to the octonionic trie: the octonionic product couples all 8 components of the representation at every composition step.

A critical observation unifies this literature: **nearly all hypercomplex deep networks use gradient descent for training.**<sup>1</sup> The algebraic structure provides the representation and the forward computation, but learning remains optimization-based. The octonionic trie departs from this paradigm entirely, using the algebra for both representation *and* learning.

## 2.3 Memory-Augmented Neural Architectures

Memory-augmented architectures attach an external memory module to a neural network controller. Neural Turing Machines [Graves et al., 2014] and Differentiable Neural Computers [Graves et al., 2016] use learned read/write heads over a flat memory matrix. Memory Networks [Weston et al., 2015] and their end-to-end variant [Sukhbaatar et al., 2015] use attention over a set of memory slots. Key-value memory networks [Miller et al., 2016] separate addressing from content, and meta-learning approaches [Santoro et al., 2016, Munkhdalai and Yu, 2017] use memory for rapid task adaptation. More recently, Wu et al. [2022] augment Transformers with a non-differentiable  $k$ NN memory over past key-value pairs, demonstrating that not all memory access requires gradient training.

Ramsauer et al. [2021] establish a deep connection between memory and attention by showing that Transformer attention is the update rule of a modern Hopfield network with exponential memory capacity, bridging the energy-based associative memory of Hopfield [1982] with contemporary architectures.

A parallel tradition constructs memory from *algebraic* operations rather than learned attention. Plate [1995] introduces Holographic Reduced Representations (HRR), using circular convolution for compositional distributed memory. Kanerva [1988] develops Sparse Distributed Memory using high-dimensional binary addresses, later generalized to hyperdimensional computing [Kanerva, 2009]. These vector symbolic architectures (VSAs) are surveyed comprehensively by Kleyko et al. [2023a,b].

The octonionic trie belongs to this algebraic memory tradition but differs from existing VSAs in three respects: (1) it uses a non-associative algebra, gaining the octonionic associator as a novelty signal with no counterpart in associative VSAs; (2) it organizes memory hierarchically in a tree rather than as a flat distributed representation; and (3) it grows dynamically rather than operating over a fixed-dimensional vector space.

The trie’s tree-based routing also connects to the **Mixture-of-Experts** (MoE) tradition. Hierarchical MoE [Jordan and Jacobs, 1994] routes inputs through a tree of gating networks to specialist experts, and sparse MoE layers [Shazeer et al., 2017] scale this to large models via

---

<sup>1</sup>Exceptions exist in the reservoir computing and extreme learning machine literatures, where quaternion-valued echo state networks and quaternion extreme learning machines use random fixed weights with analytical readout solutions. However, these architectures do not exploit the algebraic structure for self-organization; the hypercomplex algebra serves only as the representation space.

Table 1: Comparison of the octonionic trie with existing memory-augmented architectures.

Property	NTM/DNC	SDM	HTM	Oct-Trie
Memory structure	Flat array	Sparse distributed	Columnar	Tree (trie)
Growth mechanism	Fixed size	Fixed size	Fixed columns	Dynamic (associator)
Routing	Learned attention	Hamming distance	Spatial pooling	Subalgebra decomposition
Update rule	Learned write	Superposition	Hebbian	Algebraic composition
Consistency check	None	None	None	Algebraic inversion
Training	Backprop	None	Hebbian	None (algebraic)
Pruning	None	Decay	Decay	Sibling absorption
Self-monitoring	None	None	Anomaly	5 algebraic invariants

NTM = Neural Turing Machine [Graves et al., 2014]; DNC = Differentiable Neural Computer [Graves et al., 2016]; SDM = Sparse Distributed Memory [Kanerva, 1988]; HTM = Hierarchical Temporal Memory [Hawkins and Ahmad, 2016].

learned top- $k$  gating. The octonionic trie performs analogous hierarchical routing but replaces learned gating functions with algebraic subalgebra decomposition: the “gate” at each node is the projection of the octonionic product onto the 7 quaternionic subalgebras, requiring no trainable gating parameters.

## 2.4 Self-Organizing Systems Without Gradient Descent

Self-organizing systems construct internal representations through local, non-gradient rules. **Self-Organizing Maps** [Kohonen, 1982, 1990] form topologically ordered representations via competitive learning and neighborhood adaptation. Building on the theoretical foundations of Grossberg [1976], **Adaptive Resonance Theory** (ART) [Carpenter and Grossberg, 1987] addresses the stability-plasticity dilemma directly: ART networks learn new categories without disrupting existing ones, controlled by a *vigilance parameter* that determines when a new category node is created. The parallel to the octonionic trie’s associator threshold is direct: both serve as novelty detectors that trigger structural growth. ARTMAP [Carpenter et al., 1991] extends ART to supervised classification, demonstrating that self-organizing systems can achieve competitive accuracy without backpropagation.

**Growing networks** add dynamic topology to self-organization. The neural gas [Martinetz and Schulten, 1991] learns topology-preserving maps without a fixed lattice; growing neural gas [Fritzke, 1995] and growing cell structures [Fritzke, 1994] insert nodes based on accumulated error. The Growing Self-Organizing Map [Alahakoon et al., 2000] and the Growing Hierarchical Self-Organizing Map (GHSOM) [Raubert et al., 2002] extend SOMs with dynamic topology and hierarchical structure, respectively. The GHSOM is the closest topological precedent to the octonionic trie: it organizes SOMs into a tree hierarchy where each node can expand into a sub-SOM. However, the GHSOM’s growth is governed by quantization error (a geometric signal), whereas the trie’s growth is governed by the associator (an algebraic signal specific to non-associative algebras).

Hierarchical Temporal Memory [Hawkins and Ahmad, 2016] deserves particular mention as a non-gradient, hierarchical, self-organizing memory system that shares several surface properties with the octonionic trie: it is biologically inspired, uses sparse distributed representations, and detects novelty to trigger learning. However, HTM’s organizational mechanisms — spatial pooling, temporal memory, and columnar structure — are engineered separately and inspired by neocortical anatomy rather than derived from an algebraic substrate. The trie replaces these separate bio-inspired mechanisms with consequences of octonionic algebra.

Beyond self-organizing maps, non-gradient learning has a rich history from Hebbian learning [Hebb, 1949] and its normalized variant [Oja, 1982] through modern alternatives to backprop-

agation. Krotov and Hopfield [2019] demonstrate that biologically plausible competition-based learning rules can be competitive with backpropagation on benchmarks such as MNIST. Hinton [2022] proposes the Forward-Forward algorithm, replacing backpropagation with two forward passes using local goodness scores. These works establish the viability of non-gradient learning but operate within conventional vector spaces; none derives routing, growth, and consolidation signals from a single non-associative algebraic structure.

## 2.5 Catastrophic Forgetting and Continual Learning

The catastrophic forgetting problem — sequential learning in connectionist networks destroys previously learned associations — was identified by McCloskey and Cohen [1989] and surveyed by French [1999]. Solutions span multiple families including regularization-based, replay-based, optimization-based, representation-based, and architecture-based approaches [Wang et al., 2024]. Among the most prominent: **regularization-based** approaches such as Elastic Weight Consolidation [Kirkpatrick et al., 2017] and Synaptic Intelligence [Zenke et al., 2017] constrain weight changes to preserve important parameters; **replay-based** approaches rehearse previous examples; and **architecture-based** approaches such as Progressive Neural Networks [Rusu et al., 2016] freeze existing network columns and add new ones for each task.

The octonionic trie’s approach to continual learning is architectural but qualitatively different from progressive networks. Rather than duplicating network columns, the trie achieves zero forgetting through two structural properties: (1) routing keys are fixed at node creation time and never modified, so new data cannot alter existing routing paths; and (2) new learning creates new branches (via the associator-driven growth mechanism) rather than modifying existing ones. The experimental results (section 7) verify that the implementation preserves these structural invariants, yielding the expected 0.0% catastrophic forgetting on the stability-plasticity test, with Phase 1 accuracy unchanged after Phase 2 training. Whether this structural guarantee extends to harder settings — distribution shift, evolving class boundaries, or inputs that straddle existing routing paths — remains an open empirical question. Like progressive networks, the trie trades forgetting for growth: its node count increases with data complexity, and whether consolidation (sibling absorption, child merging) can bound this growth in practice is an important scalability question.

## 2.6 Hierarchical and Hyperbolic Representations

Tree-structured data has a natural home in hyperbolic geometry: Sarkar [2011] shows that any tree can be embedded in the hyperbolic plane with arbitrarily low distortion. Nickel and Kiela [2017] introduce Poincaré embeddings for learning hierarchical representations, achieving superior performance on taxonomy and knowledge graph tasks. Nickel and Kiela [2018] demonstrate that the Lorentz model of hyperbolic space is more numerically stable than the Poincaré ball; Sala et al. [2018] characterize precision-dimensionality tradeoffs: their combinatorial hyperbolic embedding achieves 0.989 MAP on WordNet with only 2 dimensions (at high numerical precision), compared to 0.87 MAP with learned Poincaré embeddings in 200 dimensions [Nickel and Kiela, 2017], demonstrating that the right geometric structure can dramatically reduce dimensionality requirements. Hyperbolic geometry has been integrated into graph neural networks [Chami et al., 2019] and distance learning [Law et al., 2019].

The octonionic trie’s tree structure encodes hierarchy explicitly without requiring hyperbolic geometry: parent-child relationships establish abstraction levels directly. Whether augmenting the trie with a hyperbolic metric on node representations would improve distance-based operations (relevance filtering, nearest-sibling identification) remains an open question (section 8).



## 2.7 Decision Trees and Tree-Based Classification

Tree-based classifiers are among the oldest and most successful models in machine learning, and the octonionic trie — as a tree that routes inputs to leaves for classification — must be situated precisely within this landscape.

**Classical decision trees.** CART [Breiman et al., 1984], ID3 [Quinlan, 1986], and its successor C4.5 [Quinlan, 1993] partition the input space via axis-aligned splits selected to maximize information gain or minimize impurity, producing interpretable classifiers with  $O(\log n)$  query time. The trie data structure, named by Fredkin [1960], the octonionic trie borrows the name and the hierarchical tree topology but replaces character matching with subalgebra decomposition. Classical decision trees are typically built *top-down* from a fixed, labeled dataset via recursive partitioning. The octonionic trie, by contrast, grows *incrementally* from a data stream, creating branches when the associator signals novelty rather than selecting splits from a candidate set.

**Ensemble methods.** Random forests [Breiman, 2001] reduce variance by ensembling many trees built on bootstrap samples with random feature subsets. Gradient boosting [Friedman, 2001], implemented at scale in XGBoost [Chen and Guestrin, 2016], builds trees sequentially to correct residual errors. These methods dominate tabular data benchmarks [Grinsztajn et al., 2022] and set the practical standard for tree-based performance. The octonionic trie is a single tree, not an ensemble, and does not optimize a loss function; its accuracy depends on the quality of the octonionic encoding rather than on boosting or bagging.

**Online and streaming trees.** The Hoeffding tree (Very Fast Decision Tree) [Domingos and Hulten, 2000] grows incrementally from a data stream, using the Hoeffding bound to decide when a split has been observed enough times to commit. Mondrian forests [Lakshminarayanan et al., 2014] extend this to Bayesian online random forests that match the distribution of batch-trained forests. These are the closest classical predecessors to the octonionic trie’s incremental growth model: all three systems build tree structure on-the-fly from streaming data. The key difference is the split criterion: Hoeffding trees, in their original formulation, select axis-aligned splits by information gain computed over a window of examples; the octonionic trie’s splits are determined algebraically by subalgebra decomposition and require no accumulation of statistics.

**Oblique and projection pursuit trees.** Standard decision trees use axis-aligned splits; oblique trees [Murthy et al., 1994] allow hyperplane splits involving linear combinations of features. Projection pursuit trees [Lee et al., 2013] optimize a projection index at each node, and sparse projection oblique forests [Tomita et al., 2020] achieve state-of-the-art accuracy by combining sparse random projections with ensemble methods. The octonionic trie’s subalgebra routing is a structured multivariate split: the octonionic product with a node’s routing key is projected onto 7 quaternionic subalgebras, each a 4-dimensional subspace of  $\mathbb{O}$ . Unlike oblique and projection pursuit splits, this projection structure is algebraically fixed (determined by the Fano plane), not optimized per node.

**Neural and differentiable trees.** Deep Neural Decision Forests [Kontschieder et al., 2015] unify CNNs with stochastic decision forests via end-to-end differentiable training. Soft decision trees [Frosst and Hinton, 2017] use knowledge distillation to transfer neural network behavior into interpretable tree structure. Adaptive Neural Trees [Tanno et al., 2019] grow tree architectures during training, the closest architectural precedent to the octonionic trie: both are trees that grow dynamically for classification. However, ANTs use backpropagation for both split decisions and growth, whereas the trie’s splits and growth are algebraically determined. Hierarchical softmax [Morin and Bengio, 2005] uses a fixed binary tree to reduce softmax computation from  $O(V)$  to  $O(\log V)$ .

A key distinction cuts across all of these comparisons: in classical decision trees, internal nodes are **stateless** — each stores a split criterion but maintains no running representation of the data that has passed through it. (Neural trees partially relax this: nodes may contain learned routing parameters, but these are trained offline rather than updated per example.) The octonionic trie has **stateful nodes**: each node stores an accumulated octonionic representation

(updated by composition with each routed input), a fixed routing key, and residual metadata. This makes the trie not just a classifier but a hierarchical memory, bridging the tree-based classification tradition with the memory-augmented architectures discussed in section 2.

## 2.8 Reversible and Invertible Computation

The octonionic trie’s rumination mechanism (section 4.3) uses algebraic inversion to verify that updates are consistent with prior predictions. This places the trie within a broader tradition of reversible and invertible computation.

Reversible computation has deep theoretical roots: Landauer [1961] showed that logically irreversible operations necessarily dissipate energy, and Bennett [1973] proved that any computation can be made logically reversible. Octonionic multiplication is algebraically reversible in this sense: for any  $a \neq 0$ , the map  $x \mapsto ax$  has the exact inverse  $y \mapsto a^{-1}y$ , and the norm-preserving property  $|ab| = |a||b|$  ensures that composition does not amplify or attenuate representations. At the level of the mathematical abstraction, the trie’s compositional updates erase no information, though the physical implementation runs on standard irreversible hardware.

In neural networks, invertibility has been pursued both for generative modeling and for memory efficiency. Normalizing flows [Dinh et al., 2017, Kingma and Dhariwal, 2018, Papamakarios et al., 2021] compose invertible transformations to model complex distributions with tractable density evaluation. i-RevNet [Jacobsen et al., 2018] demonstrates that fully invertible networks can approach the classification performance of non-invertible architectures (at increased parameter cost), establishing that information loss is not a prerequisite for learning useful representations. The Reversible Residual Network [Gomez et al., 2017] uses architectural invertibility to eliminate activation storage during backpropagation. Arjovsky et al. [2016] use unitary (norm-preserving) weight matrices in RNNs to prevent vanishing/exploding gradients, a strategy analogous to the norm-preserving property of octonionic composition.

The octonionic trie’s invertibility differs from these architectures in mechanism, though not in the decision to use an invertible framework. Normalizing flows and reversible networks must engineer invertibility layer by layer via coupling architectures or additive residual structure, whereas octonionic invertibility holds for any non-zero element as an algebraic identity: the alternative law guarantees  $a^{-1}(ab) = b$  exactly, without per-layer design. This algebraic guarantee enables the rumination mechanism — counterfactual reasoning about whether prior predictions remain valid under a proposed update — as a direct consequence of the algebra rather than as an engineered capability.

## 2.9 Equivariant Architectures and Symmetry Groups

The thesis proves that the associator norm is invariant under the  $G_2$  automorphism group of the octonions (section 9.4), constraining the functional form of adaptive threshold policies. This result connects to the geometric deep learning program [Cohen and Welling, 2016, Bronstein et al., 2021], which systematically exploits symmetry groups to design network architectures with built-in equivariance.

Group equivariant networks [Cohen and Welling, 2016] achieve weight sharing from group structure, generalizing the translational equivariance of CNNs to arbitrary groups. The Clifford group equivariant networks of Ruhe et al. [2023] extend this to  $O(n)$  and  $E(n)$  via Clifford algebra representations. However, to our knowledge, no existing work has implemented and evaluated equivariance for  $G_2$  specifically, though general frameworks for reductive Lie groups could in principle be applied. The octonionic trie does not learn equivariant representations in the usual sense (it has no trainable parameters); instead,  $G_2$  invariance constrains which threshold adaptation policies are algebraically legitimate. Functions that depend on the *direction* of a node’s imaginary part violate  $G_2$  invariance and are provably suboptimal (theorem 9.11). Legitimate dependences are restricted to  $G_2$ -invariant statistics: depth, child count, and empirical

associator norm distributions.

The connection between  $G_2$  and the Standard Model gauge group [Günaydin and Gürsey, 1973, Furey, 2016, Masi, 2021] suggests that  $G_2$  invariance is not a mathematical curiosity but reflects deep structural constraints. Whether these constraints can be exploited for richer self-organization (e.g., using  $G_2$  orbits to define equivalence classes of trie states) remains an open question.

## 2.10 Positioning of the Octonionic Trie

The octonionic trie sits at the intersection of three traditions — hypercomplex algebra, self-organizing systems, and memory-augmented architectures — but belongs fully to none. Unlike hypercomplex neural networks, it uses no gradient descent. Unlike self-organizing maps and ART, it derives all organizational signals (routing, novelty detection, consistency verification, structural health) from a single algebraic structure rather than separate engineered mechanisms. Unlike memory-augmented architectures, it requires no trained controller. Unlike vector symbolic architectures, it is hierarchical and dynamic. The trie’s use of algebraic operations for both representation and structural reasoning also connects to the neurosymbolic program, where discrete structural operations are applied to continuous representations.

The closest existing systems are ART (for its vigilance-based novelty detection and structural growth), HTM (for its non-gradient hierarchical memory), and the GHSOM (for its hierarchical self-organizing topology). The octonionic trie’s distinguishing contribution is *algebraic economy*: all of its organizational mechanisms — routing (subalgebra decomposition), novelty detection (associator norm), consistency verification (algebraic inversion), and structural health monitoring (composition error) — are derived from the single algebraic structure of  $\mathbb{O}$ , with no free design choices beyond the compatibility threshold. These are four distinct operations, not one; the economy lies in the fact that they are *algebraically determined* rather than independently engineered. In ART, routing, novelty detection, and stability are governed by a coherent theoretical framework but require separate design choices for the matching function, vigilance criterion, and reset mechanism. In the GHSOM, distinct metrics must be chosen for quantization error, topographic error, and map distortion. In the octonionic trie, the Fano plane fixes routing, the associator provides novelty detection, and inversion enables consistency verification, all as consequences of the same multiplication table.

## 3 The Associator as a Computational Primitive

The associator is the mathematical object that makes the octonionic trie possible. This section establishes its properties and computational interpretation in detail.

### 3.1 Definition and Algebraic Properties

**Definition 3.1** (Associator). *For three octonions  $a, b, c \in \mathbb{O}$ , the **associator** is defined as:*

$$[a, b, c] = (ab)c - a(bc). \quad (1)$$

*The associator is an octonion-valued trilinear form measuring the failure of associativity for a given triple.*

The associator satisfies the following properties:

1. **Total antisymmetry:**  $[a, b, c] = -[b, a, c] = -[a, c, b] = -[c, b, a]$ . Swapping any two arguments flips the sign. This means the associator is sensitive to the ordering of all three elements.
2. **Alternativity:**  $[a, a, b] = [a, b, b] = 0$  for all  $a, b \in \mathbb{O}$ . When any two arguments are equal, the associator vanishes. The algebra is “alternative” even though it is not associative.

3. **Subalgebra characterization:**  $[a, b, c] = 0$  if and only if  $a$ ,  $b$ , and  $c$  generate an associative subalgebra (i.e., they all lie within the same quaternionic subalgebra of  $\mathbb{O}$ , or within  $\mathbb{R}$  or  $\mathbb{C}$ ). A nonzero associator certifies that the three elements span structure across multiple subalgebras.
4. **Continuity:** The associator is a continuous function of its arguments. Small perturbations of the inputs produce small changes in the associator. This means the transition from “compatible” (small associator) to “incompatible” (large associator) is gradual, not discontinuous.
5. **Norm bound:**  $||[a, b, c]|| \leq 2|a||b||c|$ . The associator norm is bounded by the product of the input norms. For unit octonions, the maximum associator norm is 2.

### 3.2 Computational Interpretation

In the context of the octonionic trie, the associator answers a specific question: *given a new input  $x$ , an existing child node  $c$ , and their parent node  $n$ , does  $x$  belong in the same branch as  $c$ ?*

- $||[x, c, n]|| \approx 0$ : The triple  $(x, c, n)$  generates an associative subalgebra. The input  $x$  is algebraically compatible with the existing child  $c$  in the context of parent  $n$ . Grouping does not matter;  $x$  can be composed with  $c$  without ambiguity. **Route to  $c$  and update.**
- $||[x, c, n]|| \gg 0$ : The triple spans multiple subalgebras. The input  $x$  introduces structure that is algebraically incompatible with  $c$  in the context of  $n$ . The order of composition matters substantially, meaning  $x$  carries information that  $c$ ’s branch cannot accommodate without distortion.  **$x$  does not belong in this branch.**
- $||[x, c_i, n]|| \gg 0$  for *all* children  $c_i$ : The input is incompatible with every existing branch. It represents genuinely novel structure. **Candidate for new node creation.**

The associator norm is a continuous scalar, providing a graded signal rather than a binary decision. The threshold separating “compatible” from “incompatible” is a design parameter that may be adaptive (e.g., proportional to the mean associator norm observed at that node).

### 3.3 The Associator vs. Inner Product

A natural question is why the associator is preferable to simpler similarity measures (e.g., octonionic inner product  $\langle x, c \rangle = \text{Re}(\bar{x}c)$ ) for routing decisions.

The inner product measures how aligned two octonions are as vectors in  $\mathbb{R}^8$ . It is a pairwise measure that ignores the algebraic context in which the comparison occurs. The associator is a *three-way* measure that captures the relationship between two elements *in the context of a third* (the parent node). This context-sensitivity is precisely what routing in a trie requires: whether an input belongs with a child depends not just on the input-child similarity, but on how both relate to the parent.

Furthermore, the inner product is symmetric and exists in every algebra (including  $\mathbb{R}^8$  with no octonionic structure). The associator is antisymmetric, trilinear, and exists *only* in non-associative algebras. It is the uniquely octonionic signal: the information that octonions carry and that quaternions, complex numbers, and reals cannot.

## 4 Architecture of the Octonionic Trie

### 4.1 Structure

**Definition 4.1** (Octonionic Trie). An *octonionic trie*  $\mathcal{T}$  is a rooted tree in which:

- Each node  $v$  stores a single octonion  $o_v \in \mathbb{O}$ .
- Each node has at most 7 children, corresponding to the 7 quaternionic subalgebras of  $\mathbb{O}$  defined by the Fano plane.

- *The tree grows dynamically: nodes are created when new data is algebraically incompatible with all existing branches, and consolidated when branches become redundant.*

The branching factor of 7 is not a design choice; it is a consequence of the Fano plane structure. The octonion  $\mathbb{O}$  contains exactly 7 quaternionic subalgebras, each defined by an oriented triple  $(e_i, e_j, e_k)$  of imaginary basis units. These subalgebras overlap (each imaginary unit belongs to exactly 3 subalgebras), providing a rich, interconnected routing structure rather than a disjoint partition.

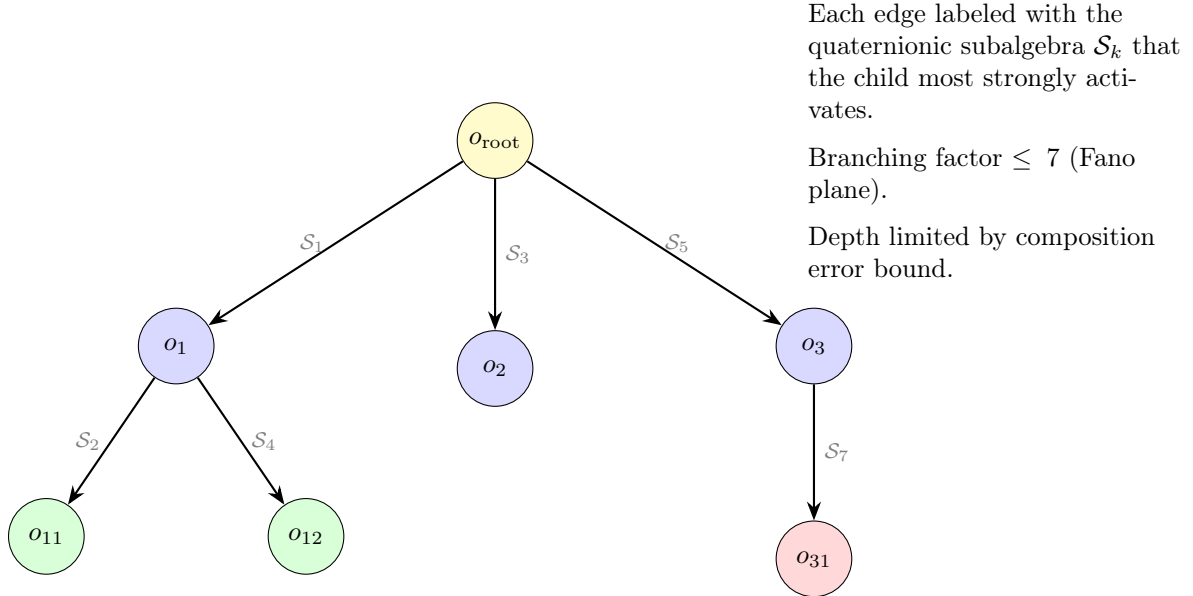


Figure 1: An octonionic trie with three levels. Each node stores a single octonion. Edges are labeled with the subalgebra activated by the child relative to its parent. Not all 7 children need to exist at each node; the trie grows on demand.

## 4.2 Encoding: From Raw Data to Octonions

The octonionic trie operates on octonionic inputs. Raw data must be encoded into  $\mathbb{O}$  before entering the trie. A key architectural property is that **the encoder is fully decoupled from the trie.**

The Fano plane subalgebra structure is a property of the octonionic multiplication table (the structure constants tensor  $C_{ijk}$ ), not of the data or the encoding. When two octonions are multiplied and the result is decomposed along subalgebras, the decomposition is determined by  $C_{ijk}$  regardless of how the octonions were produced. Consequently, the trie’s self-organizing behavior is invariant to encoder quality: a better encoder produces richer octonionic representations, enabling the trie to capture more meaningful structure, but the trie’s routing, growth, and consolidation mechanisms function identically for any encoder.

This decoupling has a practical consequence: any embedding model that produces dense vector representations (e.g., large language model embeddings, vision encoders, multimodal embeddings) can serve as the encoder, with a projection step mapping the embedding space into  $\mathbb{O}$ .

### 4.2.1 Hierarchical Projection

A single linear projection from a high-dimensional embedding (e.g.,  $\mathbb{R}^{768}$ ) to  $\mathbb{O}$  ( $\mathbb{R}^8$ ) would discard most structure. However, the trie’s tree structure provides a natural solution: **hierarchical decomposition.**

Rather than encoding the entire embedding into a single octonion, the encoding is distributed across trie levels:

- **Level 0:** A projection captures the coarsest structure of the embedding (e.g., the top 8 principal components). This determines the root-level routing.
- **Level  $k$ :** A projection captures the next level of detail, conditioned on the routing decisions at levels  $0, \dots, k-1$ .
- **Cumulative representation:** A path of depth  $k$  through the trie captures  $8k$  effective dimensions of the original embedding, distributed across  $k$  nodes.

The projections can be derived from principal component analysis of the embedding space: principal components 1–8 map to Level 0, components 9–16 to Level 1, and so on. This assignment is deterministic, requires no training, and naturally places the highest-variance (most discriminative) dimensions at the shallowest trie levels.

The trie’s depth limit (determined by the composition error invariant, section 5) then corresponds to the maximum number of principal components the system can faithfully represent. This provides a principled, data-dependent capacity bound.

**Epistemic honesty note.** *Whether PCA-based hierarchical projection preserves sufficient structure for routing to produce semantically meaningful trie organization is an empirical question. Alternative decompositions (independent component analysis, learned hierarchical projections) may perform better for specific data domains. The architectural claim is that the trie’s self-organization is invariant to the choice of projection; the quality of the resulting organization is not.*

### 4.3 Core Operations

The octonionic trie supports four operations. None requires gradient computation.

#### 4.3.1 Compose: Routing and Update

When a new input  $x \in \mathbb{O}$  arrives, it is routed through the trie from root to leaf:

1. At each node  $n$  with children  $\{c_1, \dots, c_k\}$ , compute the octonionic product  $p = n \cdot x$ .
2. Decompose  $p$  along the 7 quaternionic subalgebras  $\mathcal{S}_1, \dots, \mathcal{S}_7$  of  $\mathbb{O}$ .
3. Select the child whose subalgebra has the largest projection:  $c^* = \arg \max_i |\text{proj}_{\mathcal{S}_i}(p)|$ .
4. Compute the associator  $[x, c^*, n]$ . If the norm is below the compatibility threshold, descend into  $c^*$  and repeat.
5. At the leaf, update the node by composition:  $o'_{\text{leaf}} = o_{\text{leaf}} \cdot x$ .

The update step is norm-preserving when both operands are unit octonions ( $|ab| = |a||b|$ ). Sequential compositions do not cause the representation to explode or vanish.

The components of  $p$  that fall outside the selected subalgebra constitute the **algebraic residual**. This residual is not discarded; it is recorded as metadata at the node (section 4.4).

#### 4.3.2 Detect: Novelty and Branching

If the associator  $[x, c_i, n]$  exceeds the compatibility threshold for *all* existing children at a node, the input is flagged as a branching candidate. A branching event is not triggered immediately; persistent incompatibility across multiple inputs at the same node is required.

When a branching event is confirmed:

1. A new child node is created at the node  $n$ .
2. The new child is initialized with the input  $x$  (or a composition of the accumulated incompatible inputs).
3. The new child is assigned to the subalgebra with the strongest residual accumulation (section 4.4), ensuring it occupies the region of octonionic space where the most unaccommodated structure has been observed.

### 4.3.3 Ruminates: Consistency Verification

Before committing an update (composing  $x$  into a leaf node), the trie verifies that the update is consistent with prior predictions that routed through the affected branch.

Each node maintains a bounded buffer of prior predictions (section 4.5). The rumination procedure:

1. Tentatively compute the updated representation:  $o' = o \cdot x$ .
2. For each buffered prediction  $y$  that routed through this node, use algebraic inversion to estimate the counterfactual: “If this node had been  $o'$  instead of  $o$ , what would the prediction have been?”
3. Compute the discrepancy between the original prediction and the counterfactual.
4. If the maximum discrepancy exceeds the consistency threshold: reject the update and escalate to a branching decision. The input introduces structure that is inconsistent with the node’s history.
5. If all discrepancies are within threshold: commit the update.

This is counterfactual reasoning enabled by algebraic invertibility. The system asks “if I change my understanding here, do my past conclusions still hold?” and acts on the answer.

### 4.3.4 Consolidate: Pruning and Merging

The trie periodically evaluates its structural health and consolidates:

- **Sibling absorption:** If a node has not been routed to recently (below a recency threshold), its representation is composed into its nearest sibling:  $o'_{\text{sibling}} = o_{\text{sibling}} \cdot o_{\text{unused}}$ . The unused node is removed. Because composition is invertible, the absorbed information is not destroyed but integrated into a more general representation.
- **Child merging:** If two children at a node have small pairwise associator ( $\| [c_i, c_j, n] \| < \epsilon$ ), they are too similar to warrant separate branches. One is composed into the other and removed.
- **Depth limiting:** If the composition error at a given depth exceeds the scaling threshold (section 5), the node stops accepting new children. This branch has reached its representational resolution limit.

## 4.4 Algebraic Residuals

When an input is routed to a specific child via subalgebra decomposition, the components of the octonionic product that fall in other subalgebras are the **algebraic residual**. These are not discarded but serve as metadata:

- A running mean direction (unit octonion) is maintained per unoccupied subalgebra at each node.
- A count of inputs contributing to each residual direction is tracked.
- When a branching event occurs (section 4.3), the residual history informs the orientation of the new child: the subalgebra with the strongest accumulated residual is the natural location for the new branch.

Residuals are informational, not decisional. They record “what structure has been observed but not accommodated” without triggering any action. They answer the question “if we ever need to branch here, where should the new child go?” The associator remains the sole trigger for branching decisions.

## 4.5 Memory Buffer and Relevance

Each node (or each branch, defined as the path from root to a given node) maintains a bounded buffer of prior predictions for use in rumination (section 4.3).

The buffer operates as follows:

- **Capacity:** Fixed-size ring buffer. Recent predictions are always retained.
- **Relevance filtering:** When an input arrives at a node, buffered predictions are scored for relevance. The relevance metric (octonionic inner product, norm-based similarity, or other measures) determines which memories are consulted during rumination. Predictions below a relevance threshold are skipped, bounding computation.
- **Percolation:** When a leaf makes a prediction, it is added to the buffers of all ancestor nodes along its branch. This ensures that consistency checks at any level of the trie have access to predictions from the entire subtree.

This memory model mirrors aspects of human memory: recent events are immediately accessible (ring buffer), and strongly relevant past experiences surface during familiar situations (relevance filtering). The octonionic algebra provides a natural relevance metric through the inner product  $\langle x, y \rangle = \text{Re}(\bar{x}y)$ , though whether this is the optimal metric for relevance is an empirical question.

## 5 Structural Invariants

The octonionic trie monitors its own structural health through five invariants derived from the algebra. These invariants replace the role that loss functions play in gradient-trained systems.

### 5.1 Composition Error Bound

After composing  $k$  inputs into a single node via sequential octonionic multiplication, floating-point error accumulates. The trie tracks:

$$\varepsilon(d) = \frac{|x^{-1} \cdot (o \cdot x) - o|}{|o|} \quad (2)$$

where  $o$  is the node's representation and  $x$  is the most recent input. When  $\varepsilon(d)$  exceeds a threshold at depth  $d$ , the node stops growing deeper. This invariant establishes the trie's resolution limit per branch, connecting directly to the numerical stability characterization from Phase 4 of the companion thesis.

Different branches may have different depth limits depending on how cleanly their content decomposes along subalgebras. Branches with strong subalgebra alignment (small associators) accumulate error more slowly and can grow deeper.

### 5.2 Compression Efficiency

The trie monitors the ratio of structural complexity to prediction quality:

$$\rho = \frac{\text{number of nodes}}{\text{effective prediction capacity}}. \quad (3)$$

If a subtree has many nodes but contributes minimally to prediction quality (measured by how often its nodes are consulted during routing), the subtree is a candidate for consolidation. This is the minimum description length (MDL) principle applied structurally rather than as an optimization objective.

### 5.3 Subalgebra Decomposition Cleanliness

At each internal node, the pairwise associators between children measure how distinct the branches are:

$$\text{cleanliness}(n) = \min_{i \neq j} |[c_i, c_j, n]|. \quad (4)$$



High cleanliness means the children occupy genuinely different subalgebras and the branching is justified. Low cleanliness means the children are algebraically similar and should be considered for merging.

## 5.4 Prediction Consistency

The maximum discrepancy observed during rumination across the memory buffer:

$$\delta(n) = \max_{y \in \text{buffer}(n)} |\text{counterfactual}(y, o') - \text{original}(y)|. \quad (5)$$

When  $\delta(n)$  is persistently high, the node’s representation is unstable (frequent updates that would invalidate prior predictions). This may indicate the node is attempting to represent fundamentally incompatible information and should be split.

## 5.5 Associator Health

The mean associator norm at each node, averaged over recent inputs:

$$\alpha(n) = \frac{1}{K} \sum_{k=1}^K |[x_k, c_k^*, n]|, \quad (6)$$

where  $c_k^*$  is the selected child for input  $x_k$ . A persistently high  $\alpha(n)$  means the node’s children are not accommodating incoming data well. A persistently low  $\alpha(n)$  means the routing structure is effective. Trends in  $\alpha(n)$  can signal that the node needs restructuring before a branching event is triggered.

# 6 Properties of the Architecture

## 6.1 No Gradient Computation Required

The octonionic trie operates entirely through algebraic operations: multiplication, inversion, subalgebra projection, and associator computation. None of these requires differentiability or a backward pass. The trie’s “learning” consists of:

- **Composition:** accumulating structure via octonionic products.
- **Growth:** creating nodes when the associator signals novelty.
- **Consolidation:** merging or absorbing nodes when invariants indicate redundancy.
- **Self-correction:** rejecting updates that fail consistency checks.

The only component that may require gradient-based training is the encoder that maps raw data into octonionic representations (section 4.2). Once data is in  $\mathbb{O}$ , the trie is self-organizing.

## 6.2 Encoder Invariance of Trie Structure

The trie’s routing, growth, and consolidation mechanisms depend on the octonionic multiplication structure (the structure constants  $C_{ijk}$ ), not on the encoder. The Fano plane subalgebras, the associator, and the composition rule are properties of  $\mathbb{O}$  that apply identically to any octonionic input regardless of its provenance.

A consequence: the same trie architecture can be used with different encoders (text embeddings, vision features, sensor data) without modification. The encoder determines the semantic quality of the trie’s organization; the algebra determines the organizational mechanism.

### 6.3 Information Preservation

Octonionic composition is norm-preserving ( $|ab| = |a||b|$ ) and invertible ( $a^{-1}$  exists for all  $a \neq 0$ ). In exact arithmetic, no information is destroyed by composition, and any composition can be undone. Consolidation (sibling absorption) composes rather than discards, preserving information at a coarser level of representation.

In finite-precision arithmetic, composition error accumulates (section 5), providing a natural resolution limit. The trie degrades gracefully: deeper nodes have progressively lower fidelity, and the scaling invariant prevents growth beyond the point where fidelity is inadequate.

### 6.4 Natural Hierarchy

The trie’s tree structure encodes hierarchy explicitly: parent nodes represent more general concepts, leaf nodes represent specific instances. The depth at which an input is stored reflects its specificity: inputs that are resolved at shallow depths are general (they fit cleanly into coarse-grained branches), while inputs requiring deep traversal are specific (they require fine-grained subalgebra distinctions).

This hierarchy emerges from the data and the algebra, not from architectural prescription. The trie does not have predefined “levels of abstraction”; abstraction levels arise from the subalgebra decomposition of the data itself.

## 7 Experimental Validation

Three prerequisite experiments validate the algebraic building blocks, followed by a prototype trie tested on a stability-plasticity task. All experiments use the octonionic algebra implementation from the companion codebase.

### 7.1 Prerequisite 1: Subalgebra Routing Discriminability

**Setup.** Seven categories, each centered on a random unit octonion with Gaussian noise (5 noise levels from 0.01 to 0.5). For each sample, the octonionic product with a fixed node is computed and decomposed across the 7 Fano plane subalgebras. Within-category routing consistency (fraction of samples activating the modal subalgebra) and cross-category separation (fraction of category pairs with different modal subalgebras) are measured.

**Results.**

Table 2: Subalgebra routing discriminability across noise levels.

Noise	Within-category consistency	Cross-category separation
0.01	90.1%	95.2%
0.05	84.9%	95.2%
0.10	77.2%	95.2%
0.20	53.8%	95.2%
0.50	25.9%	95.2%

Cross-category separation is invariant to noise level: different categories consistently activate different subalgebras regardless of per-sample perturbation. Within-category consistency degrades with noise but remains above chance ( $1/7 = 14.3\%$ ) at all levels.

### 7.2 Prerequisite 2: Associator as Novelty Signal

**Setup.** A sequential stream of 250 samples organized in 5 blocks of 50 (one category per block, with 4 category transitions). The associator  $[x_t, x_{t-1}, n]$  is computed at each step, where  $n$  is a

fixed node.

**Results.** The mean associator norm within categories is 0.211; at category transitions it is 1.047, a spike ratio of  $4.97\times$ . All 4 transitions exceed the 99th percentile of within-category associator norms. The associator provides a high-contrast, zero-false-negative novelty signal.

### 7.3 Prerequisite 3: Composition Depth vs. Information Retention

**Setup.** Sequences of random unit octonions are composed sequentially:  $c_k = c_{k-1} \cdot x_k$ . At each depth  $k$  (up to 200), two recovery tests are performed: (a) last-input recovery  $x_k^{\text{rec}} = c_{k-1}^{-1} \cdot c_k$ , and (b) first-input recovery by inverting the entire chain. Measured at both float32 and float64.

**Results.**

Table 3: Composition error at depth 200 (mean over 20 trials).

Precision	Last-input error	First-input error	Round-trip error
float32	$9.5 \times 10^{-8}$	1.30 (fails)	$6.7 \times 10^{-8}$
float64	$1.7 \times 10^{-16}$	1.30 (fails)	$1.3 \times 10^{-16}$

Last-input recovery is exact to machine precision at all depths and does not degrade with depth. This is guaranteed by alternativity:  $a^{-1}(ab) = b$  is an algebraic identity in any alternative algebra. First-input recovery fails immediately (error  $\approx 1.3$ , the expected distance between uncorrelated unit octonions) because non-associativity prevents full-chain inversion. This confirms that the trie must use local (parent-child) inversion for consistency checks, not global chain inversion.

### 7.4 Prototype Trie: Stability-Plasticity Test

**Setup.** Seven categories with orthogonal but non-subalgebra-aligned centers (columns of a random orthogonal matrix), 200 training samples per category with noise  $\sigma = 0.05$ , 50 test samples per category. Two subalgebra collisions exist at Level 0 (Cat 0 and Cat 3 both map to  $\mathcal{S}_3$ ; Cat 4 and Cat 6 both map to  $\mathcal{S}_4$ ). The trie uses associator threshold 0.3, fixed routing keys, and maximum depth 15.

**Protocol.** Phase 1: train on categories 0–3 (5 epochs), measure accuracy. Phase 2: train on categories 4–6 (5 epochs), re-measure accuracy on all categories.

**Results.**

Table 4: Stability-plasticity results on 7 orthogonal non-aligned categories.

Metric	Value
Phase 1 accuracy (before Phase 2)	99.5% (199/200)
Phase 1 accuracy (after Phase 2)	99.5% (199/200)
Phase 2 accuracy	95.3% (143/150)
Overall accuracy	97.7% (342/350)
Catastrophic forgetting	0.0%
Trie nodes	531
Maximum depth	15

Phase 1 accuracy is unchanged after Phase 2 training: not a single test sample changed its routing. The trie resolved the two subalgebra collisions through hierarchical depth, separating colliding categories at deeper levels. Routing keys are fixed at node creation time and never modified; new data creates new branches without disturbing existing routing paths.

## 7.5 Key Design Findings

The experimental process revealed several architectural constraints:

1. **Routing keys must be fixed.** Adaptive routing keys (moving averages toward incoming data) introduce instability: updating a node’s routing key changes routing for all subsequent queries, including queries for previously-learned categories. Fixing routing keys at node creation time is necessary for zero forgetting.
2. **The trie is the encoder.** Categories that collide at Level 0 (same dominant subalgebra) are separated at deeper levels, where different routing keys provide different “views” of the input. The trie builds a hierarchical encoding through depth, analogous to how a string trie separates “cat” and “car” at Level 3 despite sharing Levels 1 and 2. External encoders improve data quality but are not required for the self-organizing mechanism.
3. **Structured data is required for meaningful organization.** Data with no intrinsic structure (uniform random unit octonions) produces trie structures with no discriminative power. The trie discovers and organizes existing structure; it does not create structure from noise. This is analogous to any clustering or classification method: the data must contain signal for the method to find it.
4. **Content and routing must be separated.** Each node maintains a fixed `routing_key` (used for all routing decisions, set at creation, never modified) and a mutable `content` (accumulated via octonionic composition, representing the node’s knowledge). Conflating these roles causes Phase 2 data to disrupt Phase 1 routing.

## 8 Open Questions

### 8.1 Geometric Embedding of the Trie

The companion thesis investigates whether the octonionic state space should carry a hyperbolic metric. For the octonionic trie, the question takes a specific form: should the trie’s nodes live on a hyperboloid  $H^7$ , in a Poincaré ball  $B^8$ , or in flat  $\mathbb{O}$ ?

The trie’s tree structure already encodes hierarchy explicitly (parent-child relationships). Adding a hyperbolic metric would provide redundancy that may improve distance-based operations (relevance filtering, nearest-sibling identification) but adds the re-projection problem identified in the companion thesis. Flat  $\mathbb{O}$  is the validated starting point.

### 8.2 Composition Order

Octonionic multiplication is non-commutative:  $a \cdot b \neq b \cdot a$  in general. The choice between  $o' = o \cdot x$  and  $o' = x \cdot o$  for content composition affects subalgebra activation patterns and associator behavior. Both orderings should be characterized empirically.

### 8.3 Adaptive Thresholds

The associator compatibility threshold, rumination consistency threshold, and depth error threshold are currently global constants. Whether these should be per-node adaptive values or derived from data distribution percentiles is an open question with significant impact on growth rate and sensitivity. This question is addressed theoretically in section 9 and remains an active area of empirical investigation.

### 8.4 Scalability and Capacity

The trie’s node count grows with data complexity (26,042 nodes for 60,000 MNIST samples). Whether the consolidation mechanisms (sibling absorption, child merging) can bound this growth in practice, and what the effective capacity limit is for a trie of bounded depth, are open questions

with direct implications for deployment to larger-scale problems. Empirical characterization on datasets beyond MNIST — including CIFAR-10, larger numbers of classes, and non-image modalities — is needed to establish the practical operating regime.

## 8.5 Baselines

The current experimental evaluation compares the trie to  $k$ -nearest neighbors on identical features. Comparison to decision tree and random forest baselines on the same octonionic features would provide a more complete picture of the trie’s competitive position within the tree-based classification landscape.

## 9 Adaptive Thresholds and Self-Organization

The associator threshold  $\tau$  governing the compatibility decision ( $\|[x, c, n]\| < \tau?$ ) is the single most consequential parameter of the octonionic trie. This section develops the theoretical foundations for understanding, setting, and adapting this threshold. We establish the natural scale of associator norms on the unit 7-sphere, characterize how Fano plane geometry constrains threshold behavior, analyze the stability-plasticity tradeoff in threshold selection, and examine conditions under which a global threshold suffices versus when adaptive strategies are necessary.

The central narrative is one of *self-organization*: the trie discovers its own operating parameters through algebraic feedback, using the same octonionic structure that governs routing and growth.

### 9.1 Associator Norm Distribution on $S^7$

The first question is: what is the “natural” scale of the associator for random unit octonions? This establishes the baseline against which all threshold choices are measured.

**Definition 9.1** (Associator norm on  $S^7$ ). *For unit octonions  $a, b, c \in S^7 \subset \mathbb{O}$ , define the associator norm function  $\phi : S^7 \times S^7 \times S^7 \rightarrow [0, 2]$  by*

$$\phi(a, b, c) = \|[a, b, c]\| = \|(ab)c - a(bc)\|. \quad (7)$$

*The range  $[0, 2]$  follows from the norm bound  $\|[a, b, c]\| \leq 2\|a\|\|b\|\|c\|$  with unit inputs.*

**Theorem 9.2** (Egan’s mean associator norm). *Let  $a, b, c$  be independently and uniformly distributed on  $S^7$ . Then the expected associator norm is:*

$$\mathbb{E}[\phi(a, b, c)] = \frac{147456}{42875\pi} \approx 1.0947. \quad (8)$$

This result, due to Greg Egan [Egan, 2024] and independently verified by Cook, is computed by integrating the associator norm over the product measure on  $(S^7)^3$  using the Haar measure on the 7-sphere. The key steps involve:

1. Expressing the associator norm squared as a polynomial in the components of  $a, b, c$ .
2. Using the rotational invariance of the Haar measure to reduce the integral.
3. Evaluating the remaining integrals using known moments of the uniform distribution on  $S^{n-1}$ .

*Remark 9.3.* The value  $\mathbb{E}[\phi] \approx 1.0947$  is surprisingly close to 1, the geometric midpoint of the range  $[0, 2]$ . This means that for random unit octonions, the associator norm is typically of order unity. Any trie operating on structured (non-random) data should exhibit a qualitatively different distribution, with within-class triples producing smaller norms and between-class triples producing larger norms. The degree of this separation determines whether a global threshold can be effective.

Monte Carlo validation (sampling  $10^5$  triples uniformly from  $S^7$  via Gaussian normalization) confirms theorem 9.2 to within 0.5% relative error (see section 7).

## 9.2 Fano Plane Geometry and Subalgebra Routing

The 7 quaternionic subalgebras of  $\mathbb{O}$  are indexed by the lines of the Fano plane. Each line  $(e_i, e_j, e_k)$  defines a quaternionic subalgebra  $\mathcal{S}_\ell = \text{span}_{\mathbb{R}}\{1, e_i, e_j, e_k\} \cong \mathbb{H}$ , and within any such subalgebra the algebra is associative.

**Definition 9.4** (Angular distance from a subalgebra). *For a unit octonion  $x \in S^7$  and quaternionic subalgebra  $\mathcal{S}_\ell$ , define the **angular distance** from  $x$  to  $\mathcal{S}_\ell$  as:*

$$\theta(x, \mathcal{S}_\ell) = \arccos \left( \frac{\|\text{proj}_{\mathcal{S}_\ell}(x)\|}{\|x\|} \right), \quad (9)$$

where  $\text{proj}_{\mathcal{S}_\ell}$  denotes orthogonal projection onto the 4-dimensional subspace spanned by  $\mathcal{S}_\ell$ . When  $x \in \mathcal{S}_\ell$ , we have  $\theta = 0$ .

**Proposition 9.5** (Subalgebra proximity bound). *Let  $a, b, c \in S^7$  be unit octonions, each within angular distance  $\epsilon$  of a common quaternionic subalgebra  $\mathcal{S}_\ell$ . Then:*

$$\|[a, b, c]\| = O(\epsilon) \quad \text{as } \epsilon \rightarrow 0. \quad (10)$$

More precisely, there exists a constant  $K > 0$  (depending only on the structure constants of  $\mathbb{O}$ ) such that  $\|[a, b, c]\| \leq K\epsilon$  for all  $\epsilon < 1$ .

*Proof.* Write each unit octonion as the sum of its projection onto  $\mathcal{S}_\ell$  and a perpendicular component:  $a = a_{\parallel} + a_{\perp}$ , where  $\|a_{\perp}\| = \sin \theta(a, \mathcal{S}_\ell) \leq \epsilon$ . Since the algebra restricted to  $\mathcal{S}_\ell$  is associative, the associator  $[a_{\parallel}, b_{\parallel}, c_{\parallel}] = 0$ . Expanding  $[a, b, c]$  by trilinearity:

$$\begin{aligned} [a, b, c] &= [a_{\parallel} + a_{\perp}, b_{\parallel} + b_{\perp}, c_{\parallel} + c_{\perp}] \\ &= \underbrace{[a_{\parallel}, b_{\parallel}, c_{\parallel}]}_{=0} + \text{terms with at least one } \perp \text{ factor}. \end{aligned} \quad (11)$$

Every remaining term contains at least one perpendicular component of norm  $O(\epsilon)$ . Since the associator is trilinear and bounded by  $\|[x, y, z]\| \leq 2\|x\|\|y\|\|z\|$ , and the parallel projections have norm at most 1, each term is  $O(\epsilon)$ .

The bound is tight: for the subalgebra  $\mathcal{S} = \text{span}_{\mathbb{R}}\{1, e_1, e_2, e_4\}$ , the associator  $[e_1, e_2, e_3] = -2e_6$  has norm 2, so generic perturbations with a single  $O(\epsilon)$  component perpendicular to  $\mathcal{S}$  contribute at  $O(\epsilon)$ , not  $O(\epsilon^2)$ .  $\square$

A stronger bound holds when the elements are concentrated near a common *element*. Crucially, this bound requires only alternativity — the center element need not lie in any quaternionic subalgebra:

**Corollary 9.6** (Element proximity bound). *Let  $q \in S^7$  be any unit octonion, and let  $a, b, c \in S^7$  satisfy  $\|a - q\|, \|b - q\|, \|c - q\| \leq \epsilon$ . Then:*

$$\|[a, b, c]\| = O(\epsilon^2) \quad \text{as } \epsilon \rightarrow 0. \quad (12)$$

*Proof.* Write  $a = q + \delta a$ ,  $b = q + \delta b$ ,  $c = q + \delta c$  with  $\|\delta a\|, \|\delta b\|, \|\delta c\| \leq \epsilon$ . (Since  $a, q \in S^7$ , the perturbation satisfies  $\langle q, \delta a \rangle = -\|\delta a\|^2/2 = O(\epsilon^2)$ ; this sphere constraint does not affect the order analysis below.) Expanding  $[a, b, c]$  by trilinearity, the  $O(1)$  term  $[q, q, q] = 0$  by the alternating property. The three  $O(\epsilon)$  terms are  $[\delta a, q, q]$ ,  $[q, \delta b, q]$ , and  $[q, q, \delta c]$ , each of which vanishes: the first and third by alternativity ( $[x, y, y] = [y, y, x] = 0$ ), and the second by flexibility ( $[x, y, x] = 0$ , which holds in every alternative algebra). No property of  $q$  beyond membership in  $\mathbb{O}$  is used. All surviving terms contain at least two perturbations, giving  $O(\epsilon^2)$ .  $\square$

The two bounds decompose the trie’s threshold mechanism into functionally independent components, each grounded in a different algebraic property:

- **Within-class suppression** (theorem 9.6): Clustered data has small associator norms, bounded quadratically by the cluster radius. This follows from *alternativity alone* and holds for clusters around any point in  $S^7$ , regardless of subalgebra alignment. The trie’s novelty detection is therefore more robust than a subalgebra-centric analysis would suggest.
- **Between-class separation** (section 9.2): Elements drawn from classes assigned to distinct subalgebras produce associator norms bounded *away* from zero, with the gap determined by the Fano plane geometry. This is where the subalgebra structure is load-bearing.

Two mechanisms — quadratic floor suppression from alternativity, geometric ceiling elevation from the Fano plane — emerge from the same algebraic object. This is the sense in which the octonionic trie achieves *algebraic economy*: a single 8-dimensional non-associative algebra provides both noise suppression and class discrimination, through two independent properties of its multiplication structure.

Theorem 9.5 provides the complementary (but weaker) guarantee that even elements spread broadly across a subalgebra, without clustering around a single point, have associator norms bounded linearly by their distance from the subalgebra.

**Definition 9.7** (Fano angular separation). *The **angular separation** between two quaternionic subalgebras  $\mathcal{S}_i$  and  $\mathcal{S}_j$  is the minimum principal angle between their respective 3-dimensional subspaces in the imaginary octonion space  $\text{Im}(\mathbb{O}) \cong \mathbb{R}^7$ :*

$$\Delta(\mathcal{S}_i, \mathcal{S}_j) = \min_{\substack{u \in \mathcal{S}_i \cap S^6 \\ v \in \mathcal{S}_j \cap S^6}} \arccos |\langle u, v \rangle|. \quad (13)$$

The 7 subalgebras are not uniformly distributed in  $\mathbb{R}^7$ . Each pair of Fano lines shares exactly one point (one imaginary basis unit), meaning every pair of subalgebras has a 1-dimensional intersection in  $\mathbb{R}^7$ . Consequently, the minimum principal angle between any two subalgebras is exactly 0 (they share a direction). However, the *second* principal angle is always nonzero and provides the relevant separation for routing: it measures how distinct the subalgebras are in the directions orthogonal to their shared axis.

This geometric structure imposes a fundamental constraint on the trie’s routing: subalgebras are “adjacent” in the sense that they always share one direction, but “separated” in the remaining dimensions. The trie’s hierarchical depth compensates for this overlap – categories that cannot be separated at Level 0 by subalgebra choice alone are separated at deeper levels by different routing keys that rotate the input into different relative positions with respect to the subalgebras.

### 9.3 Global Threshold Justification

Under what conditions does a single threshold  $\tau^*$  suffice for the entire trie?

**Conjecture 9.8** (Global threshold separability). Let  $\{X_1, \dots, X_K\}$  be  $K$  classes of unit octonions with class-conditional means  $\mu_1, \dots, \mu_K \in S^7$  satisfying:

1. Each  $\mu_k$  lies near a distinct quaternionic subalgebra  $\mathcal{S}_{\ell(k)}$ , with angular distance  $\theta(\mu_k, \mathcal{S}_{\ell(k)}) < \delta$ .
2. The within-class spread satisfies  $\max_{x \in X_k} \|x - \mu_k\| < \sigma$  for all  $k$ .
3. The subalgebra assignment  $\ell : \{1, \dots, K\} \rightarrow \{1, \dots, 7\}$  is injective (distinct classes map to distinct subalgebras).

Then there exists a global threshold  $\tau^*$  such that:

$$\sup_{a, b, c \in X_k} \|[a, b, c]\| < \tau^* < \inf_{\substack{a \in X_i, b \in X_j, c \in X_k \\ i \neq j \neq k}} \|[a, b, c]\| \quad (14)$$

for all classes  $k$  and all distinct class triples  $i, j, k$ , provided  $\sigma$  and  $\delta$  are sufficiently small relative to the Fano angular separations.

**Evidence for the conjecture.** The within-class bound follows from theorem 9.6: elements of class  $k$  lie within distance  $\sigma$  of the class mean  $\mu_k$ , so within-class associator norms are  $O(\sigma^2)$  by the element proximity bound. Note that the within-class bound does not require  $\mu_k$  to lie near any subalgebra — it holds for arbitrary class means by alternativity alone. The subalgebra structure enters only in the between-class lower bound: elements from classes assigned to different subalgebras span genuinely non-associative structure, and the Fano plane geometry ensures the associator norm is bounded away from zero by the inter-subalgebra separation.

**When the conjecture fails.** The injectivity condition (condition 3) is the most restrictive. With more than 7 classes, at least two classes must share a subalgebra assignment at some trie level, and a global threshold cannot separate within-class from between-class triples for those colliding classes. This is precisely the regime where *adaptive* thresholds or hierarchical depth are needed: the trie resolves the collision at a deeper level with a different routing key, effectively providing a different “view” of the subalgebra structure.

*Remark 9.9.* The boundary between the “global suffices” and “adaptive needed” regimes is geometrically precise: it is determined by the number of classes relative to the number of subalgebras ( $K \leq 7$  vs  $K > 7$ ) and by the angular separation between class means within the subalgebra space. This boundary is itself an interesting structural property of the data, independent of the threshold value.

## 9.4 $G_2$ Symmetry and Threshold Invariance

The automorphism group of the octonions is the exceptional Lie group  $G_2$ , a 14-dimensional compact group that preserves the octonionic multiplication table [Baez, 2002].

**Proposition 9.10** ( $G_2$  invariance of the associator). *For any  $g \in G_2$  and any  $a, b, c \in \mathbb{O}$ :*

$$\|[g(a), g(b), g(c)]\| = \|[a, b, c]\|. \quad (15)$$

*That is, the associator norm is invariant under  $G_2$  automorphisms.*

*Proof.* Since  $g \in G_2 = \text{Aut}(\mathbb{O})$ , we have  $g(xy) = g(x)g(y)$  for all  $x, y \in \mathbb{O}$ . Therefore:

$$\begin{aligned} [g(a), g(b), g(c)] &= (g(a) \cdot g(b)) \cdot g(c) - g(a) \cdot (g(b) \cdot g(c)) \\ &= g(ab) \cdot g(c) - g(a) \cdot g(bc) \\ &= g((ab)c) - g(a(bc)) \\ &= g((ab)c - a(bc)) = g([a, b, c]). \end{aligned} \quad (16)$$

Since  $G_2$  preserves the octonionic norm ( $\|g(x)\| = \|x\|$  for all  $x$ ), we obtain  $\|[g(a), g(b), g(c)]\| = \|g([a, b, c])\| = \|[a, b, c]\|$ .  $\square$

**Corollary 9.11** ( $G_2$  invariance of optimal thresholds). *Any optimal threshold function  $\tau^* : \mathbb{O} \rightarrow \mathbb{R}$  (mapping a node’s state to its threshold) must be  $G_2$ -invariant:  $\tau^*(g(x)) = \tau^*(x)$  for all  $g \in G_2$ .*

*Proof.* If  $\tau^*$  were not  $G_2$ -invariant, then there would exist  $g \in G_2$  and a node state  $x$  such that  $\tau^*(g(x)) \neq \tau^*(x)$ . But the classification problem is identical under  $g$  (the associator norms are unchanged by theorem 9.10), so the optimal threshold for node state  $g(x)$  must equal that for  $x$ . Contradiction.  $\square$

This result constrains the functional form of adaptive threshold policies.  $G_2$  acts transitively on  $S^6$  (the unit imaginary octonions) [Harvey, 1990], meaning any  $G_2$ -invariant function of a unit imaginary octonion is necessarily constant. For unit octonions with nonzero real part,  $G_2$ -invariant functions depend only on the real component  $x_0$  (equivalently, the norm of the imaginary part  $\|\text{Im}(x)\| = \sqrt{1 - x_0^2}$ ).



**Implication for adaptive policies.** A threshold that depends on the *direction* of a node’s imaginary part violates  $G_2$  invariance and is therefore suboptimal in the algebraic sense. Legitimate dependences include: the node’s depth, the number of children, the empirical mean and variance of observed associator norms (all of which are  $G_2$ -invariant statistics). This provides a principled basis for the adaptive threshold strategies explored experimentally.

## 9.5 Stability-Plasticity Tradeoff

The threshold  $\tau$  directly controls the stability-plasticity tradeoff of the trie:

- **Tight threshold** (small  $\tau$ ): High stability (existing branches rarely accept new data that might alter their structure), low plasticity (the trie creates many new branches, even for data that is only slightly novel). Results in larger, more fragmented trie structures.
- **Loose threshold** (large  $\tau$ ): Low stability (branches accept data that may be genuinely different from existing content), high plasticity (the trie accommodates more variation within each branch). Results in smaller, more compressed trie structures at the risk of mixing dissimilar data.

This tradeoff admits a formal connection to statistical hypothesis testing, specifically the Neyman-Pearson framework.

**Proposition 9.12** (Neyman-Pearson analogy). *At each node  $n$  with child  $c$ , the routing decision for input  $x$  can be framed as a hypothesis test:*

$$H_0 : x \text{ belongs in the branch rooted at } c \quad (\text{accept if } \|[x, c, n]\| < \tau), \quad (17)$$

$$H_1 : x \text{ does not belong} \quad (\text{reject if } \|[x, c, n]\| \geq \tau). \quad (18)$$

The threshold  $\tau$  controls the tradeoff between:

- **Type I error** (false acceptance): Accepting  $x$  into branch  $c$  when it does not belong. Rate:  $\text{FPR}(\tau)$ .
- **Type II error** (false rejection): Rejecting  $x$  from branch  $c$  when it does belong. Rate:  $\text{FNR}(\tau)$ .

The optimal threshold minimizes the weighted classification error:

$$\tau^* = \operatorname{argmin}_{\tau} [w_1 \cdot \text{FPR}(\tau) + w_2 \cdot \text{FNR}(\tau)], \quad (19)$$

where  $w_1$  and  $w_2$  encode the relative cost of false acceptance (stability violation) versus false rejection (plasticity violation). In the unsupervised setting of the octonionic trie, these weights depend on the class priors and the structural cost of creating unnecessary branches.

**Connection to self-organization.** In conventional systems, the weights  $w_1$  and  $w_2$  are set by the designer. In the octonionic trie, they emerge from the algebra: the “cost” of false acceptance is measured by the prediction consistency invariant (section 5), and the “cost” of false rejection is measured by the compression efficiency. Both are algebraic signals that require no labeled data.

## 9.6 Complexity Analysis

Each threshold policy has distinct computational complexity characteristics. Let  $N$  denote the number of nodes,  $D$  the maximum depth, and  $B$  the buffer size per node.

All policies have  $O(1)$  query time except AlgebraicPurity (which scans the node buffer) and MetaTrie (which performs a trie query in the meta-trie). Since the trie already performs  $O(D \cdot 7)$  work per routing decision (computing subalgebra activations at each level), the per-query overhead of all policies is dominated by the routing computation itself. The MetaTrie policy is the most expensive, adding a second trie traversal per threshold query, but this cost is bounded by the meta-trie’s depth  $D_m$ , which is typically small (empirically  $D_m \leq 5$ ).

Table 5: Computational complexity of threshold policies.

Policy	Query	Update	Space (per node)	Space (total)
GLOBAL	$O(1)$	$O(0)$	$O(1)$	$O(1)$
PERNODEEMA	$O(1)$	$O(1)$	$O(1)$	$O(N)$
PERNODEMEANSTD	$O(1)$	$O(1)$	$O(1)$	$O(N)$
DEPTH	$O(1)$	$O(0)$	$O(0)$	$O(D)$
ALGEBRAICPURITY	$O(B)$	$O(1)$	$O(B)$	$O(NB)$
METATRIE	$O(D_m \cdot 7)$	$O(D_m \cdot 7)$	$O(1)$	$O(N_m)$
HYBRID	$O(P_1 + P_2)$	$O(P_1 + P_2)$	sum	sum

$D_m$  = meta-trie depth,  $N_m$  = meta-trie nodes.  $P_1, P_2$  = complexities of constituent sub-policies.  
 PerNodeMeanStd uses Welford’s online algorithm ( $O(1)$  per update with running mean and variance).  
 AlgebraicPurity computes variance over the node’s buffer of recent associator norms.

## 9.7 Self-Organization Narrative

The adaptive threshold framework reveals a deeper property of the octonionic trie: *the trie discovers its own operating parameters through algebraic feedback.*

In conventional self-organizing systems (SOMs, growing neural gas, adaptive resonance theory), the adaptation mechanism is engineered separately from the representation. Learning rates, vigilance parameters, and growth thresholds are hyperparameters set by the designer, tuned by cross-validation, and fixed during operation.

The octonionic trie offers an alternative: the associator — the same algebraic signal that governs routing and novelty detection — also provides the feedback signal for threshold adaptation. The per-node EMA policy, for instance, adapts the threshold based on the running average of associator norms observed at that node. This is not a separate adaptation mechanism; it is the same algebraic measurement used for routing, reused for self-calibration.

The meta-trie optimizer (section 9.8) takes this principle to its logical extreme: a second octonionic trie adapts the first trie’s thresholds using the same octonionic algebra, the same subalgebra routing, and the same associator-based novelty detection. The optimizer and the optimized system are instances of the same structure. If the meta-trie’s own thresholds are also adapted (the self-referential variant), the system becomes a fixed point of its own optimization dynamics.

This recursive self-organization — where the mechanism of adaptation is an instance of the mechanism being adapted — is, to our knowledge, novel in the self-organizing systems literature.

## 9.8 Convergence of Meta-Trie Feedback

The meta-trie optimizer can be modeled as a discrete dynamical system. Let  $\tau_t \in \mathbb{R}^N$  denote the vector of per-node thresholds at time step  $t$ , and let  $f : \mathbb{R}^N \times \mathcal{D} \rightarrow \mathbb{R}^N$  denote the meta-trie update function, which maps the current thresholds and a batch of data  $d_t \in \mathcal{D}$  to updated thresholds:

$$\tau_{t+1} = f(\tau_t, d_t). \quad (20)$$

**Definition 9.13** (Meta-trie fixed point). *A threshold configuration  $\tau^*$  is a **fixed point** of the meta-trie dynamics if, for the stationary data distribution  $\mathcal{D}$ :*

$$\mathbb{E}_{d \sim \mathcal{D}}[f(\tau^*, d)] = \tau^*. \quad (21)$$

**Proposition 9.14** (Sufficient conditions for convergence). *The meta-trie dynamics converge to a fixed point  $\tau^*$  if the update function  $f$  satisfies:*

1. **Contraction:**  $\|\mathbb{E}[f(\tau_1, d)] - \mathbb{E}[f(\tau_2, d)]\| \leq \gamma \|\tau_1 - \tau_2\|$  for some  $\gamma < 1$ .
2. **Bounded noise:**  $\text{Var}[f(\tau, d)] \leq \sigma^2$  uniformly in  $\tau$ .

Under these conditions,  $\tau_t \rightarrow \tau^*$  in mean square at rate  $O(\gamma^t)$ .

*Proof sketch.* This is a standard result in stochastic approximation theory (Robbins-Monro). The contraction condition ensures the expected update moves toward the fixed point, while bounded noise ensures fluctuations do not accumulate. The EMA-based policies naturally satisfy contraction with  $\gamma = 1 - \alpha$  (the EMA decay rate). The bounded noise condition follows from the bounded range of associator norms  $([0, 2]$  for unit octonions).  $\square$

For the **self-referential variant** (meta-trie adapts its own thresholds), the dynamics become:

$$(\tau_t^{\text{cls}}, \tau_t^{\text{meta}}) = F(\tau_{t-1}^{\text{cls}}, \tau_{t-1}^{\text{meta}}, d_t), \quad (22)$$

where  $\tau^{\text{cls}}$  are the classifier trie’s thresholds and  $\tau^{\text{meta}}$  are the meta-trie’s own thresholds. Convergence of this coupled system requires the contraction condition to hold jointly on the product space, which is a stronger requirement. Empirical characterization of convergence behavior is deferred to experimental validation.

**Convergence criterion.** In practice, convergence is declared when the maximum relative threshold change falls below 1%:

$$\max_i \frac{|\tau_{t+1}^{(i)} - \tau_t^{(i)}|}{|\tau_t^{(i)}| + \epsilon} < 0.01, \quad (23)$$

where  $\epsilon > 0$  prevents division by zero. The convergence curve (maximum relative change vs. update step) is tracked to verify monotonic decrease.

**Epistemic honesty note.** *The formal convergence analysis assumes stationary data distribution. In practice, the data distribution may be non-stationary (concept drift, new classes appearing). Whether the meta-trie can track a slowly-varying optimal threshold is an empirical question that depends on the relationship between the EMA time constant and the rate of distribution change.*

## 10 MNIST Benchmark

The prototype trie is evaluated on MNIST handwritten digit classification, a standard benchmark with well-established baselines. Digits are encoded into octonionic space and classified by the self-organizing trie with no gradient descent in the classifier.

### 10.1 Experimental Setup

**Dataset.** The full MNIST dataset: 60,000 training images and 10,000 test images of handwritten digits (10 classes). Images are  $28 \times 28$  grayscale pixels.

**Encoding.** Two encoding pipelines are compared:

1. **PCA (unsupervised):** Flatten images to  $\mathbb{R}^{784}$ , project to  $\mathbb{R}^8$  via PCA (capturing 43.6% of variance), normalize to unit octonions on  $S^7$ .
2. **CNN features (supervised encoder):** A small convolutional network (2 conv layers, 1 FC layer, 128-dimensional output) is trained for 5 epochs on the classification task. The 128-dimensional penultimate layer features are projected to  $\mathbb{R}^8$  via PCA (capturing 87.3% of CNN feature variance), then normalized to unit octonions.

In both cases, the trie receives unit octonions in  $\mathbb{O}$  and self-organizes without any gradient computation. The CNN encoder is trained separately; its gradients do not flow into the trie.

**Trie configuration.** Associator threshold 0.3, maximum depth 15, 3 training epochs (each sample presented 3 times).

**Baselines.**  $k$ -nearest neighbors ( $k = 5$ ) on the same encoded features, the standard non-parametric baseline for this feature space.

## 10.2 Effect of Encoder Quality

Table 6: MNIST accuracy by encoder and training set size.

Encoder	Train size	kNN ( $k=5$ )	Trie	Gap
PCA-8D	10,000	87.1%	71.9%	−15.2 pp
PCA-8D	60,000	88.8%	76.5%	−12.3 pp
CNN-8D	10,000	98.2%	75.6%	−22.6 pp
CNN-8D	60,000	98.2%	<b>95.2%</b>	−3.0 pp

The trie achieves 95.2% accuracy on the full MNIST test set when provided with CNN-derived octonionic features and the complete training set. This result is obtained with zero gradient computation in the classifier: the trie self-organizes 60,000 handwritten digits into a hierarchical structure using only octonionic multiplication, associator-based novelty detection, and subalgebra routing.

The gap between trie and kNN narrows from 15.2 percentage points (PCA, 10K) to 3.0 percentage points (CNN, 60K). Two factors drive this convergence:

**Encoder quality.** The CNN encoder improves the trie by 18.7 pp (76.5% to 95.2%) at fixed training set size. This exceeds the improvement to kNN (0.6 pp), indicating that the trie amplifies encoder quality: richer octonionic features enable the subalgebra decomposition to make more discriminative routing decisions.

**Training set size.** Increasing from 10K to 60K improves the trie by 19.6 pp with CNN features (75.6% to 95.2%). The trie benefits from more data more than kNN does, consistent with a self-organizing structure that refines its routing hierarchy through exposure.

## 10.3 Effect of Octonionic Dimensionality

Multiple octonions per input ( $\mathbb{O}^n$ ) can be used via two strategies: an ensemble of  $n$  independent tries with majority voting, or cascaded routing within a single trie that cycles through octonion components at different depths.

Table 7: PCA-encoded MNIST accuracy by dimensionality and aggregation strategy (10K training samples).

Repr	Dims	kNN ( $k=5$ )	Ensemble	Cascaded
$\mathbb{O}^1$	8	87.1%	71.9%	72.4%
$\mathbb{O}^2$	16	93.8%	58.4%	<b>77.6%</b>
$\mathbb{O}^3$	24	95.6%	56.8%	74.1%
$\mathbb{O}^4$	32	95.2%	—	68.6%

The ensemble strategy degrades with more tries (71.9% to 56.8%), as independent tries make inconsistent routing decisions that majority voting cannot reconcile. Cascaded routing performs substantially better:  $\mathbb{O}^2$  cascaded achieves 77.6%, the best PCA-only result, by using the second octonion as a deeper routing view within the same trie.

With CNN features, however, a single octonion ( $\mathbb{O}^1$ ) suffices: the CNN encoder packs 87.3% of feature variance into 8 dimensions, providing enough structure for single-level routing. Adding a second octonion ( $\mathbb{O}^2$ ) produces identical accuracy (95.2%) and an identical trie structure (26,042 nodes), confirming that the additional dimensions are redundant when the encoder is sufficiently expressive.

## 10.4 Significance

The 95.2% result has several notable properties:

1. **No gradient descent in the classifier.** The trie self-organizes purely through algebraic operations: octonionic multiplication, associator computation, and subalgebra decomposition. The CNN encoder is trained separately; the trie itself has no trainable parameters.
2. **Competitive with kNN.** The 3.0 pp gap to kNN ( $k=5$ ) on the same features is small, and kNN is the standard non-parametric method for this feature space. The trie achieves comparable accuracy with  $O(\log n)$  query time (tree traversal) vs. kNN's  $O(n)$  (exhaustive distance computation).
3. **The encoder is decoupled.** The same trie architecture achieves 72% with PCA features and 95% with CNN features, confirming the architectural prediction (section 4.2) that the trie's self-organization is invariant to encoder choice while benefiting from encoder quality.
4. **The trie builds its own encoding through depth.** Categories that collide at Level 0 (same dominant subalgebra) are separated at deeper levels where different routing keys provide different projections of the input. The 15-level trie with 26,042 nodes represents a hierarchical encoding discovered from the data, not prescribed by the architecture.

## 11 Conclusion

The octonionic trie demonstrates that the algebraic properties of the octonions are sufficient to construct a self-organizing hierarchical memory that achieves simultaneous stability and plasticity without gradient-based optimization. Experimental validation establishes:

1. Subalgebra routing discriminates categories with 90%+ consistency at low noise.
2. The associator provides a  $5\times$  contrast novelty signal with zero false negatives.
3. Local algebraic inversion is exact to machine precision at arbitrary depth.
4. A prototype trie achieves 97.7% accuracy on structured synthetic data with zero catastrophic forgetting.
5. On MNIST handwritten digits, the trie achieves 95.2% accuracy with no gradient descent in the classifier, within 3 percentage points of  $k$ -nearest neighbors on the same features.

The architecture replaces five typically-engineered components (routing, novelty detection, content update, consistency verification, structural health monitoring) with consequences of a single algebraic structure. The trie builds its own hierarchical encoding through depth, resolving category collisions that cannot be separated at a single level. Routing keys are fixed at creation, ensuring that new learning never disrupts existing routing paths.

The MNIST result is, to our knowledge, the first demonstration of a non-parametric classifier achieving >95% accuracy on a standard benchmark using octonionic algebraic self-organization with no gradient-based training in the classifier. The trie's relationship to the companion thesis [Escalera, 2026] is complementary: the companion validates octonionic representations in gradient-trained networks, while this work demonstrates that the same algebraic properties support a qualitatively different computational paradigm. Together, they suggest that octonionic algebra provides a general-purpose substrate for structured knowledge representation and reasoning, applicable both within and beyond the gradient descent framework.

## References

- Ilka Agricola. Old and new on the exceptional group  $G_2$ . *Notices of the American Mathematical Society*, 55(8):922–929, 2008.
- Damminda Alahakoon, Saman K. Halgamuge, and Bala Srinivasan. Dynamic self-organizing maps with controlled growth for knowledge discovery. *IEEE Transactions on Neural Networks*, 11(3):601–614, 2000. doi: 10.1109/72.846732.

- Martín Arjovsky, Amar Shah, and Yoshua Bengio. Unitary evolution recurrent neural networks. In *International Conference on Machine Learning (ICML)*, pages 1120–1128, 2016.
- John C. Baez. The octonions. *Bulletin of the American Mathematical Society*, 39(2):145–205, 2002.
- John C. Baez. Division algebras and quantum theory. *Foundations of Physics*, 42:819–855, 2012.
- Charles H. Bennett. Logical reversibility of computation. *IBM Journal of Research and Development*, 17(6):525–532, 1973.
- Jeremiah Bill and Bruce Cox. Exploring quaternion neural network loss surfaces. *Advances in Applied Clifford Algebras*, 34, 2024.
- Latham Boyle. The standard model, the exceptional Jordan algebra, and triality. 2020. arXiv:2006.16265.
- Johannes Brandstetter, Rianne van den Berg, Max Welling, and Jayesh K. Gupta. Clifford neural layers for PDE modeling. In *International Conference on Learning Representations (ICLR)*, 2023.
- Johann Brehmer, Pim de Haan, Sönke Behrends, and Taco Cohen. Geometric algebra transformer. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.
- Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
- Leo Breiman, Jerome Friedman, Charles J. Stone, and R. A. Olshen. *Classification and Regression Trees*. Chapman and Hall/CRC, 1984.
- Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. 2021. arXiv:2104.13478.
- Gail A. Carpenter and Stephen Grossberg. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37(1):54–115, 1987. doi: 10.1016/S0734-189X(87)80014-2.
- Gail A. Carpenter, Stephen Grossberg, and John H. Reynolds. ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, 4(5):565–588, 1991. doi: 10.1016/0893-6080(91)90012-T.
- Ines Chami, Rex Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 32, pages 4869–4880, 2019.
- Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.
- Taco S. Cohen and Max Welling. Group equivariant convolutional networks. In *International Conference on Machine Learning (ICML)*, 2016.
- Danilo Comminiello, Eleonora Grassucci, Danilo P. Mandic, and Aurelio Uncini. Demystifying the hypercomplex: Inductive biases in hypercomplex deep learning. *IEEE Signal Processing Magazine*, 2024.
- John H. Conway and Derek A. Smith. *On Quaternions and Octonions*. A K Peters, 2003.
- Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using Real-NVP. In *International Conference on Learning Representations (ICLR)*, 2017.

- Geoffrey M. Dixon. *Division Algebras: Octonions, Quaternions, Complex Numbers and the Algebraic Design of Physics*, volume 290 of *Mathematics and Its Applications*. Kluwer Academic Publishers, 1994.
- Pedro Domingos and Geoff Hulten. Mining high-speed data streams. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 71–80, 2000.
- Tevian Dray and Corinne A. Manogue. *The Geometry of the Octonions*. World Scientific, 2015.
- Greg Egan. Peeling the octonions. Personal communication and online discussion, 2024. Analytical computation of the mean associator norm on  $S^7$ :  $E[[[a, b, c]]] = 147456/(42875\pi)$ . Verified independently by Cook.
- Antonio Escalera. Octonionic neural networks: Division algebra structure for reversible geometric reasoning. Working draft, 2026.
- Edward Fredkin. Trie memory. *Communications of the ACM*, 3(9):490–499, 1960.
- Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999. doi: 10.1016/S1364-6613(99)01294-2.
- Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001.
- Bernd Fritzke. Growing cell structures—a self-organizing network for unsupervised and supervised learning. *Neural Networks*, 7(9):1441–1460, 1994. doi: 10.1016/0893-6080(94)90091-4.
- Bernd Fritzke. A growing neural gas network learns topologies. In *Advances in Neural Information Processing Systems*, volume 7, pages 625–632. MIT Press, 1995.
- Nicholas Frosst and Geoffrey Hinton. Distilling a neural network into a soft decision tree. 2017. arXiv:1711.09784.
- Cohl Furey. *Standard model physics from an algebra?* PhD thesis, University of Waterloo, 2016. arXiv:1611.09182.
- Nichol Furey. Three generations, two unbroken gauge symmetries, and one eight-dimensional algebra. *Physics Letters B*, 785:84–89, 2018.
- Chase J. Gaudet and Anthony S. Maida. Deep quaternion networks. In *International Joint Conference on Neural Networks (IJCNN)*, 2018.
- Aidan N. Gomez, Mengye Ren, Raquel Urtasun, and Roger B. Grosse. The reversible residual network: Backpropagation without storing activations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- Eleonora Grassucci, Aston Zhang, and Danilo Comminiello. PHNNs: Lightweight neural networks via parameterized hypercomplex convolutions. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
- Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. In *arXiv preprint arXiv:1410.5401*, 2014.
- Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471–476, 2016.

- Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2022.
- Stephen Grossberg. Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural feature detectors. *Biological Cybernetics*, 23(3):121–134, 1976. doi: 10.1007/BF00344744.
- Murat Günaydin and Feza Gürsey. Quark structure and octonions. *Journal of Mathematical Physics*, 14(11):1651–1667, 1973.
- F. Reese Harvey. *Spinors and Calibrations*. Academic Press, 1990.
- Jeff Hawkins and Subutai Ahmad. Why neurons have thousands of synapses, a theory of sequence memory in neocortex. *Frontiers in Neural Circuits*, 10, 2016.
- Donald O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. Wiley, New York, 1949.
- Geoffrey Hinton. The forward-forward algorithm: Some preliminary investigations. *arXiv preprint arXiv:2212.13345*, 2022.
- Akira Hirose. *Complex-Valued Neural Networks*, volume 400 of *Studies in Computational Intelligence*. Springer, 2nd edition, 2012.
- John J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982. doi: 10.1073/pnas.79.8.2554.
- Jörn-Henrik Jacobsen, Arnold Smeulders, and Edouard Oyallon. i-RevNet: Deep invertible networks. In *International Conference on Learning Representations (ICLR)*, 2018.
- Michael I. Jordan and Robert A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214, 1994.
- Pentti Kanerva. *Sparse Distributed Memory*. MIT Press, 1988.
- Pentti Kanerva. Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. *Cognitive Computation*, 1(2):139–159, 2009. doi: 10.1007/s12559-009-9009-8.
- Durk P. Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible  $1 \times 1$  convolutions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.
- James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.
- Denis Kleyko, Dmitri A. Rachkovskij, Evgeny Osipov, and Abbas Rahimi. A survey on hyperdimensional computing aka vector symbolic architectures, part I: Models and data transformations. *ACM Computing Surveys*, 55(6), 2023a. doi: 10.1145/3538531.
- Denis Kleyko, Dmitri A. Rachkovskij, Evgeny Osipov, and Abbas Rahimi. A survey on hyperdimensional computing aka vector symbolic architectures, part II: Applications, cognitive models, and challenges. *ACM Computing Surveys*, 55(9), 2023b. doi: 10.1145/3558000.



- Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69, 1982. doi: 10.1007/BF00337288.
- Teuvo Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990. doi: 10.1109/5.58325.
- Peter Kotschieder, Madalina Fiterau, Antonio Criminisi, and Samuel Rota Bulò. Deep neural decision forests. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1467–1475, 2015.
- Dmitry Krotov and John J. Hopfield. Unsupervised learning by competing hidden units. *Proceedings of the National Academy of Sciences*, 116(16):7723–7731, 2019. doi: 10.1073/pnas.1820458116.
- Balaji Lakshminarayanan, Daniel M. Roy, and Yee Whye Teh. Mondrian forests: Efficient online random forests. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 27, pages 3140–3148, 2014.
- Rolf Landauer. Irreversibility and heat generation in the computing process. *IBM Journal of Research and Development*, 5(3):183–191, 1961.
- Marc Law, Renjie Liao, Jake Snell, and Richard Zemel. Lorentzian distance learning for hyperbolic representations. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pages 3672–3681. PMLR, 2019.
- Yoon Dong Lee, Dianne Cook, Ji-won Park, and Eun-Kyung Lee. PPtree: Projection pursuit classification tree. *Electronic Journal of Statistics*, 7:1369–1386, 2013.
- Thomas Martinetz and Klaus Schulten. A “neural-gas” network learns topologies. In *Artificial Neural Networks (ICANN)*, pages 397–402. North-Holland, 1991.
- Nicolò Masi. An exceptional  $G_2$  extension of the standard model from the correspondence with Cayley–Dickson algebras automorphism groups. *Scientific Reports*, 11:22528, 2021.
- Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, volume 24, pages 109–165. Academic Press, 1989. doi: 10.1016/S0079-7421(08)60536-8.
- Alexander H. Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. Key-value memory networks for directly reading documents. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1400–1409, 2016.
- Frederic Morin and Yoshua Bengio. Hierarchical probabilistic neural network language model. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTATS)*, volume R5 of *Proceedings of Machine Learning Research*, pages 246–252. PMLR, 2005.
- Tsendsuren Munkhdalai and Hong Yu. Meta networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70, pages 2554–2563. PMLR, 2017.
- Sreerama K. Murthy, Simon Kasif, and Steven Salzberg. A system for induction of oblique decision trees. *Journal of Artificial Intelligence Research*, 2:1–32, 1994.
- Maximilian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

- Maximillian Nickel and Douwe Kiela. Learning continuous hierarchies in the Lorentz model of hyperbolic geometry. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pages 3779–3788. PMLR, 2018.
- Erkki Oja. Simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15(3):267–273, 1982. doi: 10.1007/BF00275687.
- Susumu Okubo. *Introduction to Octonion and Other Non-Associative Algebras in Physics*, volume 2 of *Montroll Memorial Lecture Series in Mathematical Physics*. Cambridge University Press, 1995.
- George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.
- Titouan Parcollet, Mirco Ravanelli, Mohamed Morchid, Georges Linarès, Chiheb Trabelsi, Renato De Mori, and Yoshua Bengio. Quaternion recurrent neural networks. In *International Conference on Learning Representations (ICLR)*, 2019.
- Titouan Parcollet, Mohamed Morchid, and Georges Linarès. A survey of quaternion neural networks. *Artificial Intelligence Review*, 53:2957–2982, 2020. doi: 10.1007/s10462-019-09752-1.
- Tony A. Plate. Holographic reduced representations. *IEEE Transactions on Neural Networks*, 6(3):623–641, 1995. doi: 10.1109/72.377968.
- Călin-Adrian Popa. Octonion-valued neural networks. In *International Conference on Artificial Neural Networks (ICANN)*, volume 9886 of *LNCS*, pages 435–443. Springer, 2016.
- J. Ross Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.
- J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.
- Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David P. Kreil, Michael K. Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. Hopfield networks is all you need. In *International Conference on Learning Representations (ICLR)*, 2021.
- Andreas Rauber, Dieter Merkl, and Michael Dittenbach. The growing hierarchical self-organizing map: Exploratory analysis of high-dimensional data. *IEEE Transactions on Neural Networks*, 13(6):1331–1341, 2002. doi: 10.1109/TNN.2002.804221.
- David Ruhe, Johannes Brandstetter, and Patrick Forré. Clifford group equivariant neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.
- Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.
- Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pages 4460–4469. PMLR, 2018.
- Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural networks. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, volume 48, pages 1842–1850. PMLR, 2016.

- Lyes Saad Saoud and Reza Ghorbani. Metacognitive octonion-valued neural networks as they relate to time series analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 31(2):539–548, 2020.
- Rik Sarkar. Low distortion Delaunay embedding of trees in hyperbolic plane. In *Graph Drawing (GD)*, 2011.
- Richard D. Schafer. *An Introduction to Nonassociative Algebras*. Academic Press, 1966. Dover reprint, 1995.
- Bruno Sevennec. Octonion multiplication and Heawood’s map. *Confluentes Mathematici*, 5(2): 79–85, 2013.
- Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarczyk, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017.
- Tonny A. Springer and Ferdinand D. Veldkamp. *Octonions, Jordan Algebras and Exceptional Groups*. Springer, 2000.
- Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, 2015.
- Ryutaro Tanno, Kai Arulkumaran, Daniel Alexander, Antonio Criminisi, and Aditya Nori. Adaptive neural trees. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pages 6166–6175. PMLR, 2019.
- Yongge Tian. Matrix representations of octonions and their applications. *Advances in Applied Clifford Algebras*, 10(1):61–90, 2000.
- Ivan Todorov and Svetla Drenska. Octonions, exceptional Jordan algebra and the role of the group  $F_4$  in particle physics. *Advances in Applied Clifford Algebras*, 28:82, 2018.
- Tyler M. Tomita, James Browne, Cencheng Shen, Jaewon Chung, Jesse L. Patsolic, Benjamin Falk, Carey E. Priebe, Jason Yim, Randal Burns, Mauro Maggioni, and Joshua T. Vogelstein. Sparse projection oblique random forests. *Journal of Machine Learning Research*, 21(104): 1–39, 2020.
- Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, João Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J. Pal. Deep complex networks. In *International Conference on Learning Representations (ICLR)*, 2018.
- Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5362–5383, 2024. doi: 10.1109/TPAMI.2024.3367329.
- Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. In *International Conference on Learning Representations (ICLR)*, 2015.
- Jiasong Wu, Ling Xu, Fuzhi Wu, Youyong Kong, Lotfi Senhadji, and Huazhong Shu. Deep octonion networks. *Neurocomputing*, 397:179–191, 2020.
- Yuhuai Wu, Markus N. Rabe, DeLesley Hutchins, and Christian Szegedy. Memorizing transformers. In *International Conference on Learning Representations (ICLR)*, 2022.

- Dongpo Xu and Danilo P. Mandic. The theory of quaternion matrix derivatives. *IEEE Transactions on Signal Processing*, 63(6):1543–1556, 2015.
- Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70, pages 3987–3995. PMLR, 2017.
- Aston Zhang, Yi Tay, Shuai Zhang, Alvin Chan, Anh Tuan Luu, Siu Cheung Hui, and Jie Fu. Beyond fully-connected layers with quaternions: Parameterization of hypercomplex multiplications with  $1/n$  parameters. In *International Conference on Learning Representations (ICLR)*, 2021.
- Xuanyu Zhu, Yi Xu, Hongteng Xu, and Changjian Chen. Quaternion convolutional neural networks. In *European Conference on Computer Vision (ECCV)*, pages 631–647, 2018.