Memory as Structural Invariance: A Topological, Stochastic, and Noetherian Framework

Aslan Alwi[1,2], Munirah Munirah[1,2], and Almudaya Research Institute Team[1]

[1]Independent Researcher at Almudaya Research Institute, Yogyakarta, Indonesia
[2]Department of Informatics Engineering, Faculty of Engineering, Universitas Muhammadiyah Ponorogo, Indonesia, *Correspondence email*: aslan.alwi@umpo.ac.id

January 2026

Preface

Memory is commonly understood as storage: a place where information, events, or representations are kept. This intuition is deeply embedded in everyday language, in computer science, and even in many scientific theories of mind and matter. Yet, when examined closely, this view proves insufficient for explaining how memory operates in complex systems—natural or artificial.

This book begins from a simple but unsettling observation: many systems clearly exhibit memory without storing anything in an explicit sense. Neural networks remember patterns even when individual weights fluctuate. Biological organisms retain identity despite continuous material turnover. Physical systems carry traces of their past not as records, but as constraints on what can happen next. In these cases, memory does not reside in objects, symbols, or locations. It resides in structure.

The central claim of this work is that memory should be understood as a form of *structural invariance* induced by dynamical evolution. Systems remember not by preserving states, but by preserving relationships—by shaping landscapes within which future behavior unfolds. From this perspective, memory is neither an add-on nor a container; it is an emergent property of how a system evolves.

This book was written to articulate that perspective in a systematic way. It develops a framework in which memory is defined topologically, independent of explicit energy functions, loss functions, or symbolic representations. Deterministic and stochastic systems are treated on equal footing. Classical models such as Hopfield networks and Boltzmann machines are revisited as early instances of a more general principle. New computational realizations are proposed, and the framework is extended to domains ranging from machine learning to spacetime physics.

The reader should not expect a single unifying equation or a closed-form solution. This is not a book that seeks to reduce memory to a scalar quantity or an optimization objective. Instead, it offers a language—structural, topological, and dynamical—for recognizing memory wherever persistent constraints emerge from change.

This work is intentionally interdisciplinary. Mathematicians may be drawn to the use of topology, quotient spaces, and invariants. Researchers in machine learning may recognize familiar architectures reinterpreted through a structural lens. Physicists may find resonances with ideas of symmetry, conservation, and irreversibility. Philosophers may encounter an alternative account of identity, meaning, and continuity. No single background is assumed, and no single discipline is privileged.

Because of this breadth, the book can be read in multiple ways. Readers primarily interested in conceptual foundations may focus on the early chapters and the philosophical synthesis. Those interested in algorithms may proceed directly to the computational realizations and experiments, returning to the theoretical chapters as needed. The appendices provide mathematical and implementation details for readers who wish to engage more deeply with formal aspects.

Finally, this book does not claim to offer the last word on memory. It offers a shift in viewpoint. If the arguments presented here are successful, they suggest that memory is not something systems possess, but something systems *become*. What a system remembers is inseparable from what it can still do.

The hope is that this perspective will prove useful—not only for understanding existing systems, but for designing new ones, and for recognizing memory in places where it has not traditionally been sought.

Epigraph

Bismillah

Bagi sebagian orang.. memori adalah tempat bagi kata-kata dan peristiwa..
Akan tetapi mungkin.. itu adalah tempat bagi makna-makna..
Dan setiap cara pandang dunia..
Memberimu cara menyimpan ke dalam makna-makna...

Lalu ketika cara pandang duniamu berevolusi..
Maka caramu menyusun makna pun berevolusi..
Sehingga seluruh kaitan kelindang makna yang tersimpan
tercerahkan dan diperbaharui..

Tetapi setiap pencerahan makna mungkin melahirkan
keterkaitan makna yang baru..
Atau melahirkan makna-makna baru..
Yang demikian mungkin menjadi cara bagi kelindang syaraf
menemukan pengetahuan baru..

The Semantic Meanings and Their Evolution
22 Januari 2026

Contents

I Foundations 1

1 Introduction: Memory as Structure 3
- 1.1 Motivation and scope . . . 3
- 1.2 A short conceptual preview . . . 3
- 1.3 Relation to prior work . . . 4
- 1.4 Outline of the book . . . 4
- 1.5 Terminology and notation . . . 5
- 1.6 A note on evidence and methodology . . . 5
- 1.7 References used in this chapter . . . 5

2 State Space, Topology, and Implicit Landscapes 7
- 2.1 State space as a topological object . . . 7
- 2.2 Dynamics as continuous mappings . . . 7
- 2.3 Attractors, basins, and invariant sets . . . 8
  - 2.3.1 Invariant sets . . . 8
  - 2.3.2 Basins of attraction . . . 8
- 2.4 Landscape without energy . . . 8
- 2.5 Stochastic landscapes and ensembles . . . 9
- 2.6 Topological equivalence of landscapes . . . 9
- 2.7 Memory as structural constraint . . . 9
- 2.8 Relation to classical and modern models . . . 10
  - 2.8.1 Hopfield networks . . . 10
  - 2.8.2 Boltzmann machines . . . 10
  - 2.8.3 Clustering and unsupervised learning . . . 10
- 2.9 Summary of Part II . . . 10

3 Invariants and Equivalence Classes 11
- 3.1 Why invariants cannot be points . . . 11
- 3.2 Equivalence classes of landscapes . . . 11
  - 3.2.1 Landscape equivalence . . . 12
  - 3.2.2 Interpretation . . . 12
- 3.3 Definition of memory invariant . . . 12
- 3.4 Memory as hyperspace selection . . . 12
- 3.5 False memory and structural validity . . . 13
- 3.6 Forgetting as topological transition . . . 13
- 3.7 Relation to clustering and unsupervised learning . . . 13
- 3.8 Relation to neural and physical systems . . . 14
  - 3.8.1 Neural systems . . . 14

3.8.2 Physical systems 14
3.9 Summary of Part III 14

II Noetherian Generalization and Stochastic Principle 15

4 From Noether to Memory: Conceptual Bridge 17
4.1 Noether's theorem: the structural insight 17
4.2 Limits of classical Noetherian conservation 17
4.3 Structural equivalence as generalized symmetry 18
4.4 From symmetry to memory 18
4.5 Toward a Noether–Memory principle 18
4.6 Stochastic generalization 19
4.7 Energy as a special case 19
4.8 Implications for learning and adaptation 19
4.9 Implications for time and irreversibility 20
4.10 Summary of Part IV 20

5 Noether–Memory Stochastic Principle (Formal) 21
5.1 Structural interpretation of Noether's theorem 21
5.2 Axiomatic setting 21
5.3 Formal statement of the principle 22
5.4 Stochastic generalization 22
5.5 Forgetting as symmetry breaking 22
5.6 Relation to classical conservation laws 22
5.7 Summary 23

III Computational Realizations 25

6 Attractors, Hopfield, and Boltzmann Machines 27
6.1 Attractors as memory primitives 27
6.2 Hopfield networks as deterministic landscapes 27
6.2.1 Landscape interpretation 27
6.3 Boltzmann machines as stochastic ensembles 28
6.3.1 Ensemble view 28
6.4 Mapping to invariant-based memory 28
6.5 Limitations of classical models 28
6.6 Beyond classical models 29
6.7 Summary 29

7 Stochastic Topological Memory Networks (STMN) 31
7.1 Design principles 31
7.2 Network as a probabilistic graph 31
7.2.1 Graph definition 31
7.2.2 Probabilistic edges 32
7.3 Ensemble of network realizations 32
7.4 Dynamics and activation 32
7.5 Implicit landscapes and basins 32

7.6 Memory invariants in STMN 33
7.6.1 Operational definition 33
7.6.2 Structural view 33
7.7 Learning as distributional deformation 33
7.7.1 Beyond point updates 33
7.7.2 Structural learning rule (conceptual) 33
7.8 Relation to existing stochastic methods 33
7.8.1 Dropout 33
7.8.2 Bayesian neural networks 34
7.8.3 Ensemble learning 34
7.9 Robustness and continual learning 34
7.10 False memory and rare invariants 34
7.11 Minimal examples 34
7.11.1 Two-layer STMN 34
7.11.2 Relation to clustering 34
7.12 Summary of Part V 35

8 Clustering, Manifolds, and Non-Neural ML 37
8.1 Unsupervised learning as structure discovery 37
8.2 Clustering as basin formation 37
8.2.1 Implicit dynamics of clustering 38
8.2.2 Clusters as basins 38
8.3 Centroids as geometric charts 38
8.4 False clusters and spurious memory 38
8.5 Dimensionality reduction as hyperspace selection 39
8.6 Manifold learning and topological persistence 39
8.7 Topological Data Analysis (TDA) 40
8.7.1 Persistent homology 40
8.7.2 Relation to clustering 40
8.8 Supervised learning revisited 40
8.9 Summary of Part VI 40

IV Cosmic Scale: Memory of Spacetime 43

9 Spacetime as Memory: An Elastic Landscape 45
9.1 Why spacetime must be considered 45
9.2 Spacetime as a topological manifold 45
9.3 Deterministic memory in general relativity 46
9.3.1 Causal structure 46
9.3.2 Geodesic congruences 46
9.3.3 Horizons 46
9.4 Stochastic spacetime and ensemble geometries 46
9.5 Cosmic memory invariants 47
9.5.1 Definition 47
9.5.2 Relation to energy conservation 47
9.6 Arrow of time as memory accumulation 47
9.7 Spacetime as a learning system 47
9.8 Relation to biological and computational memory 48

9.9 Limits and open problems 48
9.10 Summary of Part VII 48

10 Arrow of Time, Irreversibility, and Memory 49
10.1 Time symmetry and its limits 49
10.2 Entropy-based accounts revisited 49
10.3 Memory as directional constraint 50
10.4 Irreversibility without dissipation 50
10.5 Structural accumulation and temporal depth 50
10.6 Learning as irreversible symmetry breaking 51
10.7 Relation to biological and cognitive time 51
10.8 Relation to cosmological time 51
10.9 Comparison with other approaches 51
10.10 Implications for reversibility and control 52
10.11 Summary of Part VIII 52

V Philosophical and Practical Implications 53

11 Identity, Meaning, and Consciousness (Speculative) 55
11.1 Identity beyond substance 55
11.2 Identity as equivalence class 55
11.3 Meaning as structural relation 56
11.3.1 Semantic landscapes 56
11.4 Evolution of meaning 56
11.5 Consciousness as structural attunement 56
11.6 Attention and awareness 57
11.7 False meaning and illusion 57
11.8 Ethical and epistemic implications 57
11.9 Human, artificial, and cosmic continuity 57
11.10 Summary of Part IX 58

12 Applications and Experiments 59

13 Conclusions and Future Work 61
13.1 Summary of contributions 61
13.2 What this framework does not claim 62
13.3 Empirical and computational directions 62
13.3.1 Machine learning experiments 62
13.3.2 Neuroscience and cognition 62
13.3.3 Physics and cosmology 63
13.4 Theoretical extensions 63
13.5 A note on future work 63
13.6 Closing reflection 63

A Mathematical Background 65
A.1 Topological spaces and continuity 65
A.1.1 Topological spaces 65
A.1.2 Continuous maps 66

A.2 Dynamical systems and invariant sets . . . 66
A.2.1 Deterministic dynamics . . . 66
A.2.2 Invariant sets . . . 66
A.2.3 Basins of attraction . . . 66
A.3 Quotient spaces and equivalence relations . . . 66
A.3.1 Equivalence relations induced by dynamics . . . 66
A.3.2 Quotient topology . . . 67
A.4 Random dynamical systems . . . 67
A.4.1 Basic definition . . . 67
A.4.2 Random invariant sets . . . 67
A.5 Structural stability and equivalence . . . 67
A.6 Persistent homology . . . 68
A.6.1 Simplicial complexes . . . 68
A.6.2 Filtrations . . . 68
A.6.3 Persistence diagrams . . . 68
A.6.4 Stability theorem . . . 68
A.7 Relation to memory invariants . . . 68

B Implementation Notes 69
B.1 Parameterization of stochastic edges . . . 69
B.1.1 Edge distributions . . . 69
B.1.2 Learnable parameters . . . 70
B.2 Sampling and forward propagation . . . 70
B.3 Training objectives . . . 70
B.3.1 Expected loss minimization . . . 70
B.3.2 Structural regularization . . . 70
B.4 Gradient estimation . . . 71
B.4.1 Score-function estimator . . . 71
B.4.2 Reparameterization and relaxations . . . 71
B.5 Training loop (pseudocode) . . . 71
B.6 Computational complexity . . . 71
B.7 Diagnostics for memory invariants . . . 72
B.8 Relation to existing methods . . . 72
B.9 Practical considerations . . . 72
B.10 Limitations . . . 73

C Supplementary Proofs 75
C.1 Structural validity of memory invariants . . . 75
C.2 Forgetting as topological transition . . . 75
C.3 Persistence under stochastic dynamics . . . 76
C.4 Operational equivalence via persistent homology . . . 76
C.5 Equivalence class stability under learning . . . 76
C.6 Arrow of time and invariant accumulation . . . 77
C.7 Scope and limitations . . . 77

D Applications and Experiments **79**
D.1 Experiment I: Stochastic associative memory . . . 79
D.1.1 Setup . . . 79
D.1.2 Procedure . . . 79
D.1.3 Observation . . . 80
D.1.4 Interpretation . . . 80
D.2 Experiment II: Structural persistence under noise . . . 80
D.2.1 Setup . . . 80
D.2.2 Procedure . . . 80
D.2.3 Results . . . 80
D.2.4 Interpretation . . . 80
D.3 Experiment III: Clustering as memory encoding . . . 80
D.3.1 Setup . . . 80
D.3.2 Procedure . . . 81
D.3.3 Results . . . 81
D.3.4 Interpretation . . . 81
D.4 Experiment IV: Learning dynamics and memory preservation . . . 81
D.4.1 Setup . . . 81
D.4.2 Procedure . . . 81
D.4.3 Results . . . 81
D.4.4 Interpretation . . . 81
D.5 Metrics and diagnostics . . . 82
D.6 Limitations of the experiments . . . 82
D.7 Reproducibility . . . 82
D.8 Summary . . . 82

Part I

Foundations

Chapter 1

Introduction: Memory as Structure

Overview

This work addresses a single foundational question: *how does memory arise naturally in complex systems?* We approach this question not by positing a substrate for storage, nor by assuming an explicit cost or energy function, but rather by identifying the structural conditions under which a system's past constrains its future. In short: memory is a *structural invariant* of the dynamical landscape the system inhabits.

1.1 Motivation and scope

Traditional notions of memory—whether in neuroscience, computer science, or physics—frequently appeal to metaphors of storage: a ledger, a tape, or a repository of states. In machine learning, "memory" is often equated with trained weights; in classical mechanics, conservation laws encode quantities preserved along trajectories. Yet these views either posit a privileged numerical quantity (energy, loss, weights) or reduce memory to artifact of implementation.

This book proposes a different, more general, and arguably more fundamental perspective: memory is an emergent *topological* property of a system's state space under its dynamics. This reframing has three immediate advantages:

1. **Generality:** it applies to deterministic and stochastic systems alike, and to systems across physical, biological, and computational domains.
2. **Robustness:** by elevating topology above metric quantities, memory becomes stable under deformations and many forms of noise.
3. **Unifying power:** it subsumes classical models (e.g., Hopfield attractors [Hopfield, 1982]) and stochastic models (e.g., Boltzmann machines [Ackley et al., 1985]) as special cases, while pointing toward new algorithmic architectures.

1.2 A short conceptual preview

The central conceptual moves of this work are:

- Treat the state space of a system as a *topological* space; do not assume a priori any metric, energy function, or linear structure.

- Regard the dynamics as a (possibly stochastic) continuous map on that topological space; invariance under this map identifies *invariant sets*.
- Define memory as the *class* of invariant sets that persist across realizations of the (possibly stochastic) dynamics. These classes are topological in nature: they are preserved under homeomorphisms or, in weaker formulations, under homologous transformations.
- Introduce a Noether-like principle: whereas Noether links continuous symmetries to conserved numerical quantities [Noether, 1918], our "Noether–Memory" principle links structural equivalence classes (topological symmetries) to classes of memory invariants.

1.3 Relation to prior work

The approach we develop builds on several well-established literatures:

- **Attractor networks and associative memory.** Hopfield's model [Hopfield, 1982] demonstrates how attractors in a dynamical landscape can implement content-addressable memory. Hopfield networks thus serve as a canonical deterministic instantiation of our more general framework.
- **Stochastic neural models.** Boltzmann machines [Ackley et al., 1985] and related energy-based models illustrate how stochastic sampling in a landscape produces an ensemble of realizations; these are natural precursors to our stochastic landscape viewpoint.
- **Topological data analysis (TDA).** Methods from computational topology (persistent homology, mapper, etc.) provide concrete tools to extract invariant topological structures from data [Carlsson, 2009, Edelsbrunner and Harer, 2010]. We use these tools both conceptually and practically to operationalize memory invariants.
- **Random dynamical systems.** The mathematical framework for stochastic flows and random attractors is well-developed and relevant for formal claims about persistence under noise [Arnold, 1998].
- **Modern machine learning techniques.** Techniques such as dropout [Srivastava et al., 2014] and Bayesian neural networks [Neal, 1996] embody stochasticity in learning; we reinterpret these in light of structural memory rather than (only) regularization or uncertainty estimation.

1.4 Outline of the book

The book is organized in four parts. Part I lays out the formal topological foundation of memory. Part II presents the Noether–Memory principle and its stochastic generalization. Part III derives computational realizations (Hopfield, Boltzmann, and Stochastic Topological Memory Networks). Part IV extends the framework to spacetime and cosmological considerations, and Part V discusses philosophical implications and applications.

1.5 Terminology and notation

We will use the following standard notational conventions throughout:

- $\mathcal{S}$: the state space (topological space) of a system.
- Φ_t: the (deterministic or stochastic) evolution operator / flow on $\mathcal{S}$.
- $M \subset \mathcal{S}$: an invariant set under Φ_t.
- $\mathcal{L}$: landscape (understood as the induced structure of reachability and stability), not necessarily a scalar function.
- $\{\mathcal{L}_\omega\}_{\omega\in\Omega}$: ensemble (family) of landscape realizations indexed by stochastic sample ω.

1.6 A note on evidence and methodology

This book is primarily theoretical and programmatic: we develop a conceptual and formal framework and show how canonical models fit as instances. That said, a central objective is to produce testable suggestions and implementable algorithms (Part III), and to identify empirical signatures in neuroscience and machine learning that would support the view that memory is structural and topological.

1.7 References used in this chapter

Key foundational works we will rely on include Noether [1918], Hopfield [1982], Ackley et al. [1985], Carlsson [2009], Edelsbrunner and Harer [2010], Arnold [1998], Srivastava et al. [2014], Neal [1996]. Additional references appear in each chapter.

Chapter 2

State Space, Topology, and Implicit Landscapes

Overview

This chapter establishes the minimal mathematical and conceptual structure required to speak about memory without invoking explicit storage mechanisms, energy functions, or loss landscapes. We argue that the appropriate foundational object is not a scalar-valued function, but a topological organization of a system's state space induced by its dynamics.

2.1 State space as a topological object

We begin with the most general notion of a system's configuration: its *state space*. Let

$$\mathcal{S}$$

denote the set of all possible states accessible to a system. No assumption is made *a priori* that $\mathcal{S}$ is a vector space, a metric space, or a differentiable manifold.

Instead, we equip $\mathcal{S}$ with a topology τ, rendering $(\mathcal{S}, \tau)$ a topological space. The topology encodes the notion of continuity, neighborhood, and connectedness—concepts that are more fundamental than distance or energy.

Rationale. In complex systems, proximity in meaning, function, or consequence often does not coincide with numerical proximity. Two states may be far apart in a metric sense yet adjacent in causal or semantic structure. Topology captures this adjacency without committing to a specific geometry.

2.2 Dynamics as continuous mappings

Let the temporal evolution of the system be given by a family of maps

$$\Phi_t : \mathcal{S} \to \mathcal{S},$$

parameterized by time $t \in \mathbb{R}$ or $t \in \mathbb{Z}$. We assume only that each Φ_t is continuous with respect to the topology τ.

This assumption is minimal and deliberately weak:

- Φ_t need not be invertible.
- Φ_t need not be differentiable.
- Φ_t may be stochastic, defining a random dynamical system [Arnold, 1998].

Continuity ensures that the evolution respects the topological structure of $\mathcal{S}$, preserving connectedness and allowing invariant sets to be well-defined.

2.3 Attractors, basins, and invariant sets

2.3.1 Invariant sets

Definition 2.3.1 (Invariant Set). *A subset $M \subseteq \mathcal{S}$ is said to be invariant under the dynamics Φ_t if*

$$\Phi_t(M) = M \quad \textit{for all } t.$$

Invariant sets generalize fixed points, limit cycles, strange attractors, and invariant manifolds. Crucially, invariance is a structural property, not a numerical one.

2.3.2 Basins of attraction

For an invariant set M, define its basin of attraction as

$$\mathcal{B}(M) = \{x \in \mathcal{S} \mid \lim_{t \to \infty} \Phi_t(x) \in M\},$$

where the limit is understood in a topological or probabilistic sense.

Basins induce a partition (possibly overlapping or fuzzy) of the state space. This partition is central to our notion of memory: it encodes how past states constrain future evolution.

2.4 Landscape without energy

In classical mechanics and many machine learning models, the notion of a *landscape* is tied to a scalar function—energy, cost, or loss—whose minima define attractors. However, this construction is neither necessary nor sufficient for defining memory.

Key claim. A landscape is not fundamentally a function $E : \mathcal{S} \to \mathbb{R}$, but a *topological structure* induced by:

- invariant sets,
- their basins of attraction,
- and the connectivity relations among these basins.

We therefore define:

Definition 2.4.1 (Implicit Landscape). *An implicit landscape $\mathcal{L}$ on $\mathcal{S}$ is the collection*

$$\mathcal{L} = \{(M_i, \mathcal{B}(M_i))\}_{i \in I},$$

where $\{M_i\}$ are invariant sets under Φ_t and $\mathcal{B}(M_i)$ their associated basins.

No scalar ordering is assumed among the M_i. What matters is reachability and stability, not numerical depth.

2.5 Stochastic landscapes and ensembles

In many systems of interest, the dynamics Φ_t is stochastic or subject to noise. In such cases, a single landscape realization is insufficient.

Let Ω denote a probability space indexing realizations of the dynamics. Each realization $\omega \in \Omega$ induces a landscape

$$\mathcal{L}_\omega.$$

We thus obtain an ensemble of landscapes:

$$\{\mathcal{L}_\omega\}_{\omega \in \Omega}.$$

This ensemble view aligns naturally with:

- Boltzmann machines [Ackley et al., 1985],
- Bayesian neural networks [Neal, 1996],
- dropout-based architectures [Srivastava et al., 2014],
- and random dynamical systems in physics [Arnold, 1998].

2.6 Topological equivalence of landscapes

We now arrive at a central notion.

Definition 2.6.1 (Topological Equivalence of Landscapes). *Two landscapes $\mathcal{L}_\omega$ and $\mathcal{L}_{\omega'}$ are said to be topologically equivalent if there exists a homeomorphism (or weaker, a homology-preserving map) $h : \mathcal{S} \to \mathcal{S}$ such that:*

$$h(M_i^\omega) = M_i^{\omega'}$$

and

$$h(\mathcal{B}(M_i^\omega)) = \mathcal{B}(M_i^{\omega'}).$$

This equivalence formalizes the idea that stochastic fluctuations may alter geometric details without altering the structural organization of the landscape.

2.7 Memory as structural constraint

We now connect these constructions to memory.

Core insight. Memory is not an element within $\mathcal{S}$, but a structural constraint that selects and stabilizes a class of admissible landscapes.

Invariant sets do not merely reside *in* a landscape; they determine which landscapes are dynamically accessible.

2.8 Relation to classical and modern models

2.8.1 Hopfield networks

Hopfield networks correspond to deterministic systems with explicitly defined energy functions [Hopfield, 1982]. In our framework, the energy function is a convenient coordinate chart on the implicit landscape.

2.8.2 Boltzmann machines

Boltzmann machines realize stochastic sampling over an ensemble of landscapes [Ackley et al., 1985]. Their stationary distributions correspond to invariant statistical structures.

2.8.3 Clustering and unsupervised learning

Clustering algorithms partition $\mathcal{S}$ into stable basins. Centroids serve as geometric representatives, but the true invariants are the basin structures themselves [Carlsson, 2009, Edelsbrunner and Harer, 2010].

2.9 Summary of Part II

In this chapter, we have established that:

- State spaces should be treated topologically, not metrically.
- Dynamics induce invariant sets and basins, forming implicit landscapes.
- Landscapes need not be defined by scalar functions.
- Stochastic systems give rise to ensembles of topologically equivalent landscapes.
- Memory emerges as a structural constraint on admissible landscapes.

These results prepare the ground for the Noether–Memory principle developed in the next part.

Chapter 3

Invariants and Equivalence Classes

Overview

In the previous chapter, we introduced implicit landscapes as topological structures induced by invariant sets and their basins under system dynamics. We now take the crucial next step: defining *memory invariants* not as elements embedded within a landscape, but as higher-order structures that bind, select, and stabilize entire classes of admissible landscapes.

This chapter formalizes memory as a second-order topological concept: an invariant of equivalence classes rather than of individual states.

3.1 Why invariants cannot be points

A common intuition—especially inherited from attractor-based models—is to regard memory as a point, state, or localized region within a state space. While this intuition is serviceable in simple deterministic systems, it fails to generalize to:

- stochastic dynamics,
- systems with fluctuating connectivity,
- learning systems whose state space itself evolves.

In such systems, any particular attractor realization may shift, deform, or even temporarily disappear, while the *structural role* it plays remains intact. This observation forces a conceptual shift:

> Memory is not what occupies a landscape, but what stabilizes the landscape itself.

3.2 Equivalence classes of landscapes

Recall from Chapter 2 that stochastic dynamics induce an ensemble of landscapes

$$\{\mathcal{L}_\omega\}_{\omega\in\Omega}.$$

3.2.1 Landscape equivalence

Definition 3.2.1 (Landscape Equivalence)**.** *Two landscapes $\mathcal{L}_\omega$ and $\mathcal{L}_{\omega'}$ are said to be equivalent, written*

$$\mathcal{L}_\omega \sim \mathcal{L}_{\omega'},$$

if there exists a homeomorphism (or, more weakly, a homology-preserving map)

$$h : \mathcal{S} \to \mathcal{S}$$

such that invariant sets, basins of attraction, and their adjacency relations are preserved under h.

This equivalence relation partitions the ensemble of landscapes into equivalence classes:

$$[\mathcal{L}] = \{\mathcal{L}_\omega \mid \mathcal{L}_\omega \sim \mathcal{L}_{\omega'}\}.$$

3.2.2 Interpretation

Each equivalence class represents a family of geometrically distinct but structurally identical realizations. These classes are the natural carriers of memory in stochastic and adaptive systems.

3.3 Definition of memory invariant

We are now in a position to state the central definition of this chapter.

Definition 3.3.1 (Memory Invariant)**.** *A memory invariant $\mathfrak{M}$ is a structural constraint that:*

1. *selects a class of topologically equivalent landscapes $[\mathcal{L}]$,*
2. *stabilizes this class under admissible dynamics and stochastic perturbations,*
3. *and restricts the set of hyperspaces (state spaces and their induced landscapes) accessible to the system.*

Formally, a memory invariant is identified with an equivalence class $[\mathcal{L}]$ rather than any particular landscape realization $\mathcal{L}_\omega$.

Key distinction. Invariant sets $M \subset \mathcal{S}$ are first-order invariants. Memory invariants $\mathfrak{M}$ are second-order invariants: invariants of landscape structure itself.

3.4 Memory as hyperspace selection

The definition above implies a profound shift in perspective.

Let $\mathcal{H}$ denote the space of all admissible state spaces and their induced landscapes. Then a memory invariant acts as a selector:

$$\mathfrak{M} : \mathcal{H} \to \mathcal{H}_{\mathfrak{M}} \subset \mathcal{H},$$

restricting the system's evolution to a subset of structurally compatible hyperspaces.

Consequences.

- Memory constrains future possibilities rather than reproducing past states.
- Learning corresponds to transitions between subsets $\mathcal{H}_{\mathfrak{M}}$.
- Forgetting corresponds to a loss or reconfiguration of admissible hyperspaces.

3.5 False memory and structural validity

In classical accounts, false memory is treated as error or noise. In the present framework, this distinction is reframed.

Lemma 3.5.1 (Structural Validity of False Memory)**.** *Any landscape equivalence class* $[\mathcal{L}]$ *that is dynamically stable constitutes a valid memory invariant, regardless of its correspondence to external ground truth.*

Sketch. Structural stability under admissible dynamics suffices for invariance. External correspondence is an epistemic criterion, not a topological one. □

This lemma explains why false memories can be persistent, coherent, and resistant to correction.

3.6 Forgetting as topological transition

Definition 3.6.1 (Forgetting)**.** *Forgetting occurs when a system undergoes a transition*

$$[\mathcal{L}] \to [\mathcal{L}']$$

such that no homeomorphism exists between the two equivalence classes.

This may occur through:

- bifurcations,
- loss or creation of invariant sets,
- merging or splitting of basins,
- or global changes in connectivity.

Forgetting is thus a *structural* event, not a deletion of content.

3.7 Relation to clustering and unsupervised learning

Clustering algorithms provide a concrete illustration of memory invariants.

- Clusters correspond to basins of attraction.
- Centroids are geometric representatives, not invariants.
- Stable cluster partitions correspond to landscape equivalence classes.

Thus, unsupervised learning can be interpreted as the discovery of memory invariants rather than the optimization of objective functions.

3.8 Relation to neural and physical systems

3.8.1 Neural systems

In biological and artificial neural systems, synaptic changes alter the accessibility of hyperspaces. Memory invariants correspond to stable regimes of connectivity and attractor organization, not to individual neuron states.

3.8.2 Physical systems

In physical systems, particularly in spacetime dynamics, causal structure and global topology play the role of memory invariants. Local geometric fluctuations do not erase these structures.

3.9 Summary of Part III

We have established that:

- Memory invariants are second-order structural objects.
- They bind and select classes of landscapes rather than occupying them.
- Learning and forgetting are transitions between hyperspace classes.
- False memories are structurally valid invariants.
- This framework unifies clustering, neural memory, and physical memory.

These results prepare the ground for the Noether–Memory principle, which formalizes the relationship between structural equivalence and conservation in the next chapter.

Part II

Noetherian Generalization and Stochastic Principle

Chapter 4

From Noether to Memory: Conceptual Bridge

Overview

This chapter establishes the conceptual and formal bridge between the theory of memory developed in the previous chapters and the foundational insights of Noether's theorem. Rather than reinterpreting Noether's result in physical terms, we extract its structural core and extend it to a broader class of systems—including stochastic, adaptive, and non-equilibrium systems—where classical conservation laws no longer apply, but structural persistence remains.

4.1 Noether's theorem: the structural insight

In its classical formulation, Noether's theorem states that every continuous symmetry of the action of a physical system corresponds to a conserved quantity [Noether, 1918]. While often presented as a statement about energy, momentum, or angular momentum, the deeper content of the theorem lies elsewhere.

Key observation. Noether's theorem does not privilege any specific conserved quantity. Instead, it establishes a general correspondence:

> *Structural invariance under transformation* $\implies$ *existence of a conserved structure.*

Energy conservation, in particular, is a consequence of invariance under time translation—not a primitive assumption.

4.2 Limits of classical Noetherian conservation

The classical formulation of Noether's theorem rests on assumptions that are not satisfied in many systems of contemporary interest:

- the existence of a well-defined action functional,
- deterministic dynamics,

- time-reversal symmetry,
- and global differentiability.

However, modern systems—neural networks, learning algorithms, biological systems, and even spacetime at quantum scales—often violate these assumptions. Yet such systems unmistakably exhibit memory, persistence, and structural constraint.

This motivates a shift:

From conservation of numerical quantities to conservation of structure.

4.3 Structural equivalence as generalized symmetry

In the framework developed so far, the fundamental invariant is not a number but an equivalence class of landscapes.

Definition 4.3.1 (Structural Equivalence)**.** *Two dynamical realizations are structurally equivalent if their induced landscapes belong to the same equivalence class under topological transformations.*

This notion plays the role of symmetry in the generalized setting. Instead of invariance under time translation or spatial rotation, we consider invariance under:

- deformation of geometry,
- stochastic perturbation,
- reparameterization of internal representation.

4.4 From symmetry to memory

We now state the central conceptual correspondence of this chapter.

Classical Noether	$\Longrightarrow$	Memory framework
Symmetry of action		Structural equivalence of landscapes
Conserved quantity		Memory invariant
Energy, momentum		Constraints on admissible hyperspaces
Deterministic paths		Ensemble of stochastic realizations

Interpretation. Memory invariants play the same structural role in complex systems that conserved quantities play in classical mechanics: they restrict the set of admissible future evolutions.

4.5 Toward a Noether–Memory principle

We are now in a position to formulate a generalized principle.

Principle 4.5.1 (Noether–Memory Principle (Conceptual))**.** *Whenever a system exhibits structural equivalence across its dynamical realizations, there exists a corresponding memory invariant that constrains its future evolution.*

This principle:

- does not assume an action functional,
- does not require determinism,
- and does not depend on numerical conservation laws.

It applies equally to physical, biological, and computational systems.

4.6 Stochastic generalization

In stochastic systems, structural equivalence is not exact but statistical.

Let $\{\mathcal{L}_\omega\}_{\omega\in\Omega}$ denote an ensemble of landscapes induced by stochastic dynamics. We say that the system satisfies stochastic structural equivalence if:

$$\Pr\left(\mathcal{L}_\omega \sim \mathcal{L}_{\omega'}\right) \geq \theta$$

for a sufficiently high threshold θ.

Principle 4.6.1 (Stochastic Noether–Memory Principle)**.** *If a system exhibits stochastic structural equivalence of its landscapes, then it possesses memory invariants that are conserved in a statistical sense.*

This principle replaces exact conservation with persistence under noise.

4.7 Energy as a special case

Within this framework, classical energy conservation appears as a limiting case.

Observation. When:

- dynamics are deterministic,
- the action is well-defined,
- and time-translation symmetry holds,

structural equivalence reduces to classical symmetry, and memory invariants reduce to conserved numerical quantities.

Thus, energy is not displaced but subsumed.

4.8 Implications for learning and adaptation

The Noether–Memory principle provides a new interpretation of learning:

- Learning is not optimization toward a minimum.
- Learning is the reorganization of structural equivalence classes.
- Stability of learning corresponds to persistence of memory invariants.

Catastrophic forgetting, under this view, is a failure to preserve structural equivalence.

4.9 Implications for time and irreversibility

In classical mechanics, time-reversal symmetry underlies energy conservation. In memory-bearing systems, time-reversal symmetry is typically broken.

Key insight. The arrow of time emerges not from entropy alone, but from the accumulation and persistence of memory invariants.

Memory thus provides a structural account of irreversibility.

4.10 Summary of Part IV

We have shown that:

- Noether's theorem encodes a general relation between invariance and conservation.
- Structural equivalence generalizes symmetry beyond numerical invariance.
- Memory invariants generalize conserved quantities.
- A Noether–Memory principle governs deterministic and stochastic systems alike.

This completes the conceptual foundation required to derive concrete computational and physical realizations in the following parts.

Chapter 5

Noether–Memory Stochastic Principle (Formal)

Overview

This chapter presents a formal statement of the Noether–Memory principle introduced conceptually in earlier chapters. The purpose here is not to reproduce Noether's theorem in its variational form, but to extract and generalize its structural core to systems lacking a global action functional, including stochastic and learning systems.

We show that memory invariants arise as conserved structural equivalence classes under symmetry-like constraints on dynamical evolution.

—

5.1 Structural interpretation of Noether's theorem

In its classical formulation, Noether's theorem establishes a correspondence between continuous symmetries of the action and conserved quantities of motion.

Abstracting from the variational formalism, the theorem can be interpreted structurally:

> If a system's dynamics is invariant under a class of transformations, then there exists a corresponding invariant structure preserved along its evolution.

This interpretation does not presuppose energy, momentum, or even numerical conservation.

—

5.2 Axiomatic setting

We adopt the following minimal assumptions:

Axiom 1 (State space). The system evolves on a topological space $(\mathcal{S}, \tau)$.

Axiom 2 (Dynamics). Evolution is given by a (possibly stochastic) dynamical rule

$$\Phi(t, \omega) : \mathcal{S} \to \mathcal{S}.$$

Axiom 3 (Structural symmetry). There exists a family of transformations $\mathcal{G}$ acting on $\mathcal{S}$ such that for all $g \in \mathcal{G}$,

$$g \circ \Phi(t, \omega) \sim \Phi(t, \omega) \circ g,$$

where $\sim$ denotes structural equivalence (e.g., topological or homological equivalence).

—

5.3 Formal statement of the principle

Theorem 5.3.1 (Noether–Memory Principle). *If the dynamics of a system is structurally invariant under a transformation group $\mathcal{G}$, then there exists a corresponding memory invariant, defined as an equivalence class of landscapes preserved under the induced evolution.*

Interpretation. The conserved object is not a scalar quantity, but a structural constraint on admissible future dynamics.

—

5.4 Stochastic generalization

In stochastic systems, invariance holds statistically rather than pointwise.

Definition 5.4.1 (Statistical structural invariance). *A structure is statistically invariant if it is preserved with probability at least $1 - \delta$ under stochastic evolution.*

Corollary 5.4.2 (Stochastic Noether–Memory). *If a stochastic dynamical system exhibits statistical structural invariance, then the associated memory invariant is conserved in probability.*

—

5.5 Forgetting as symmetry breaking

Forgetting corresponds to the loss of structural invariance.

Corollary 5.5.1 (Forgetting as topology change). *Forgetting occurs if and only if the symmetry class underlying a memory invariant is broken, resulting in a transition between topological equivalence classes.*

—

5.6 Relation to classical conservation laws

Classical conserved quantities are recovered as special cases:

- numerical invariants arise when structure reduces to scalar observables,
- energy conservation corresponds to time-translation symmetry,
- memory conservation generalizes this to non-numerical domains.

—

5.7 Summary

The Noether–Memory principle formalizes the intuition that memory is the structural residue of symmetry under dynamics. It extends Noether's insight beyond mechanics, encompassing learning systems, stochastic processes, and complex adaptive systems.

Part III

Computational Realizations

Chapter 6

Attractors, Hopfield, and Boltzmann Machines

Overview

This chapter situates classical associative memory models within the general framework of memory as structural invariance. We show that Hopfield networks and Boltzmann machines arise as specific instances of deterministic and stochastic memory landscapes, respectively.

—

6.1 Attractors as memory primitives

In dynamical systems theory, an attractor is an invariant set toward which trajectories converge.

Within the present framework:

- attractors correspond to first-order memory structures,
- basins of attraction encode retrieval dynamics,
- the full memory resides in basin topology rather than attractor states alone.

—

6.2 Hopfield networks as deterministic landscapes

A Hopfield network defines a deterministic dynamical system on a discrete state space, governed by an energy function Hopfield [1982].

6.2.1 Landscape interpretation

The energy function induces a landscape whose local minima correspond to stored patterns.

From a structural perspective:

- the precise energy values are inessential,

- what matters is the existence and stability of minima and basins,
- memory corresponds to the induced equivalence class of landscapes.

Thus, Hopfield memory is a deterministic instance of structural invariance.
—

6.3 Boltzmann machines as stochastic ensembles

Boltzmann machines generalize Hopfield networks by introducing stochastic state transitions Ackley et al. [1985].

6.3.1 Ensemble view

Rather than a single landscape, Boltzmann machines define:

- an ensemble of realizations,
- a probability distribution over states,
- statistical persistence of attractors.

This aligns naturally with the stochastic memory framework developed in this book.
—

6.4 Mapping to invariant-based memory

Both Hopfield and Boltzmann models can be mapped to the invariant-based view:

- deterministic memory $\rightarrow$ fixed topological class,
- stochastic memory $\rightarrow$ ensemble-preserved equivalence class.

They differ only in the mechanism by which invariance is realized, not in the nature of memory itself.
—

6.5 Limitations of classical models

Classical associative memory models exhibit:

- limited capacity,
- brittle attractor structure,
- difficulty with continual learning.

These limitations correspond to insufficient control over structural invariants during learning.
—

6.6 Beyond classical models

The general framework developed here:

- removes the need for explicit energy functions,
- allows stochasticity at the structural level,
- supports learning as landscape deformation.

Stochastic Topological Memory Networks extend classical models by operating directly on equivalence classes rather than fixed attractors.

—

6.7 Summary

Hopfield networks and Boltzmann machines are not isolated constructs, but early realizations of a broader principle: memory arises from the structural invariance of dynamical landscapes. Recognizing this continuity clarifies both their power and their limitations.

Chapter 7

Stochastic Topological Memory Networks (STMN)

Overview

This chapter derives a concrete class of learning systems—*Stochastic Topological Memory Networks* (STMN)—as a direct computational realization of the Noether–Memory principle. Unlike conventional neural networks, STMN do not encode memory in fixed parameter values. Instead, memory is realized as the persistence of topological structures across an ensemble of stochastic network realizations.

7.1 Design principles

The design of STMN follows four principles derived from previous chapters:

1. **Topology first:** memory is structural, not numerical.
2. **Stochastic realization:** individual network instances may fluctuate.
3. **Equivalence over instances:** memory persists as an equivalence class.
4. **Learning as deformation:** learning reshapes distributions, not points.

These principles deliberately avoid assuming an explicit energy or loss landscape.

7.2 Network as a probabilistic graph

7.2.1 Graph definition

An STMN is defined as a directed graph

$$G = (V, E),$$

where V denotes the set of computational units (nodes) and $E \subseteq V \times V$ the set of directed edges.

Unlike standard neural networks, edges are not fixed-weight parameters.

7.2.2 Probabilistic edges

Each edge $(i, j) \in E$ is associated with a random variable

$$W_{ij} \sim \mathcal{P}_{ij},$$

where $\mathcal{P}_{ij}$ is a probability distribution over $\mathbb{R}$ (or a bounded subset thereof).

In the simplest instantiation:

$$W_{ij} = Z_{ij} \cdot w_{ij}, \quad Z_{ij} \sim \text{Bernoulli}(p_{ij}),$$

with amplitude w_{ij} and activation probability p_{ij}.

Thus, edges may be fully active, weakened, or absent in any given realization.

7.3 Ensemble of network realizations

Each stochastic sample ω induces a concrete network instance

$$G_\omega = (V, E_\omega),$$

and a corresponding computational mapping

$$\mathcal{N}_\omega : \mathcal{X} \to \mathcal{Y}.$$

The system is therefore characterized not by a single network, but by an ensemble:

$$\{\mathcal{N}_\omega\}_{\omega \in \Omega}.$$

Interpretation. Individual networks fluctuate, but the ensemble structure carries memory.

7.4 Dynamics and activation

For a given realization ω, node activations evolve according to:

$$x_j^{(t+1)} = \sigma\left(\sum_i W_{ij}^{(\omega)} x_i^{(t)}\right),$$

where σ is a (possibly non-linear) activation function.

The dynamics define a stochastic flow on the state space $\mathcal{S}$, inducing a realization-specific implicit landscape $\mathcal{L}_\omega$.

7.5 Implicit landscapes and basins

Each realization $\mathcal{N}_\omega$ induces:

- invariant sets (fixed points, cycles, or distributions),
- basins of attraction,
- adjacency relations among basins.

Collectively, these form an implicit landscape $\mathcal{L}_\omega$ as defined in Chapter 2.

7.6 Memory invariants in STMN

7.6.1 Operational definition

A representation pattern R is said to constitute a *memory invariant* if:

$$\Pr_{\omega\sim\Omega}\left(R \in \mathcal{L}_\omega \text{ as a stable structure}\right) \geq \theta,$$

for some persistence threshold $\theta \in (0, 1)$.

Key point. Memory is not tied to any single realization, but to statistical persistence across the ensemble.

7.6.2 Structural view

Equivalently, memory invariants correspond to equivalence classes:

$$\mathfrak{M} = [\mathcal{L}_\omega],$$

stable under admissible stochastic perturbations.

7.7 Learning as distributional deformation

7.7.1 Beyond point updates

Conventional learning updates weights:

$$w_{ij} \leftarrow w_{ij} - \eta \nabla_{w_{ij}} \mathcal{L}.$$

In STMN, learning operates on distributions:

$$\mathcal{P}_{ij} \leftarrow \mathcal{P}'_{ij}.$$

The objective is not to minimize a scalar loss, but to reshape the ensemble so that desired structures become persistent.

7.7.2 Structural learning rule (conceptual)

A generic learning update takes the form:

$$\mathcal{P}_{ij}^{(t+1)} = \mathcal{U}\left(\mathcal{P}_{ij}^{(t)}, \Delta\mathfrak{M}\right),$$

where $\Delta\mathfrak{M}$ encodes the discrepancy between current and desired invariant structures.

Remark. Specific instantiations may recover gradient-based learning as a special case.

7.8 Relation to existing stochastic methods

7.8.1 Dropout

Dropout introduces stochastic edge removal during training but treats stochasticity as a regularization trick [Srivastava et al., 2014]. In contrast, STMN treat stochastic connectivity as an ontological feature present during both training and inference.

7.8.2 Bayesian neural networks

Bayesian neural networks interpret weight distributions as epistemic uncertainty [Neal, 1996]. STMN reinterpret distributions structurally: they encode admissible deformations of memory landscapes.

7.8.3 Ensemble learning

Ensemble methods average predictions across models. STMN average *structure*, not outputs.

7.9 Robustness and continual learning

Because memory resides in equivalence classes rather than specific parameters:

- local perturbations do not erase memory,
- catastrophic forgetting is mitigated,
- new memories can be incorporated by expanding or deforming equivalence classes.

7.10 False memory and rare invariants

Structures that are:

- statistically rare,
- but structurally stable,

constitute false or spurious memories.

Their persistence is explained naturally within the STMN framework as a consequence of structural stability, not error accumulation.

7.11 Minimal examples

7.11.1 Two-layer STMN

A minimal STMN consists of an input layer, a hidden layer with probabilistic edges, and an output layer. Even in this simple configuration, ensemble-induced basins can be observed.

7.11.2 Relation to clustering

When dynamics are restricted to partitioning input space, STMN reduce to stochastic clustering models, where clusters correspond to basins and centroids to geometric representatives.

7.12 Summary of Part V

We have shown that:

- STMN instantiate the Noether–Memory principle algorithmically.
- Memory is encoded as persistence across stochastic network realizations.
- Learning reshapes distributions, not fixed parameters.
- Classical neural models arise as limiting cases.

STMN thus provide a bridge between abstract topological memory theory and implementable machine learning systems.

Chapter 8

Clustering, Manifolds, and Non-Neural ML

Overview

Thus far, we have focused primarily on neural and network-based realizations of memory invariants. In this chapter, we demonstrate that the proposed framework is not specific to neural architectures. Instead, it applies naturally to non-neural machine learning methods, particularly clustering, dimensionality reduction, and manifold learning.

We argue that these methods instantiate memory not through parameterized networks, but through stable topological decompositions of data space.

8.1 Unsupervised learning as structure discovery

Unsupervised learning is often described as pattern discovery without labels. From the perspective developed in this book, this description is incomplete.

Reframing. Unsupervised learning is the discovery of *structural invariants* in the data-generating process.

Rather than fitting a function, the algorithm induces:

- partitions,
- embeddings,
- connectivity structures,
- and equivalence relations

on the data space.

These structures persist under perturbation and thus constitute memory invariants.

8.2 Clustering as basin formation

Let $\mathcal{X}$ denote the data space, equipped with a topology induced by similarity or neighborhood relations.

8.2.1 Implicit dynamics of clustering

Although clustering algorithms are not usually presented dynamically, they implicitly define a flow:

$$\Phi : \mathcal{X} \to \mathcal{C},$$

where $\mathcal{C}$ denotes a set of cluster assignments.

Iterative algorithms (e.g., k-means, EM, spectral clustering) repeatedly apply such mappings until a stable configuration is reached.

8.2.2 Clusters as basins

Each cluster $C_i \subset \mathcal{X}$ acts as a basin of attraction:

- points near the cluster center remain in the cluster,
- perturbations within the basin do not change assignment,
- transitions occur only at basin boundaries.

Key insight. The cluster itself—not the centroid—is the invariant structure.

Centroids are geometric representatives; clusters are topological objects.

8.3 Centroids as geometric charts

In many algorithms, clusters are summarized by centroids:

$$\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} x.$$

From our perspective:

- centroids are local coordinate charts,
- they facilitate computation,
- but they do not define memory.

Clusters may deform, shift, or stretch while preserving their topological identity. Memory resides in the persistence of the basin, not in the numerical stability of μ_i.

8.4 False clusters and spurious memory

Clustering algorithms often produce clusters that are:

- statistically weak,
- poorly aligned with semantic categories,
- or sensitive to initialization.

Yet some of these clusters persist across perturbations.

Lemma 8.4.1 (Structural Validity of False Clusters)**.** *Any cluster partition that remains stable under admissible perturbations constitutes a valid memory invariant, regardless of semantic correctness.*

This mirrors the phenomenon of false memory in cognitive systems: structural stability, not correctness, governs persistence.

8.5 Dimensionality reduction as hyperspace selection

Dimensionality reduction techniques (e.g., PCA, MDS, autoencoders) map data into lower-dimensional spaces:

$$f : \mathcal{X} \to \mathcal{Y}, \quad \dim(\mathcal{Y}) < \dim(\mathcal{X}).$$

In our framework, such mappings do not merely compress data—they select a *hyperspace.*

Interpretation. Choosing a lower-dimensional embedding restricts:

- which structures can be represented,
- which invariants are preserved,
- and which memories remain accessible.

Learning thus becomes an act of hyperspace selection.

8.6 Manifold learning and topological persistence

Manifold learning methods assume that data lie near a low-dimensional manifold embedded in high-dimensional space.

Examples include:

- Isomap,
- Locally Linear Embedding (LLE),
- Laplacian Eigenmaps.

These methods preserve:

- neighborhood relations,
- geodesic connectivity,
- local topology.

Memory interpretation. The learned manifold constitutes a memory invariant: it constrains future representations and operations on the data.

8.7 Topological Data Analysis (TDA)

Topological Data Analysis provides explicit tools to extract invariants from data [Carlsson, 2009, Edelsbrunner and Harer, 2010].

8.7.1 Persistent homology

Persistent homology identifies:

- connected components,
- loops,
- higher-dimensional holes,

that persist across scales.

These features are natural candidates for memory invariants:

- they are robust to noise,
- independent of coordinate choice,
- and stable under deformation.

8.7.2 Relation to clustering

Clusters correspond to connected components at appropriate scales. Persistence diagrams encode the lifespan of these components, offering a quantitative measure of memory strength.

8.8 Supervised learning revisited

Even in supervised learning, labels do not create structure; they select among existing invariants.

Claim. Labels act as external constraints that bias the system toward particular equivalence classes of landscapes.

Thus, supervised learning can be seen as guided memory selection rather than pure function approximation.

8.9 Summary of Part VI

We have shown that:

- Non-neural ML methods naturally induce memory invariants.
- Clustering creates basin structures that function as memory.
- Centroids and embeddings are representational aids, not invariants.
- Dimensionality reduction selects hyperspaces.

- TDA provides explicit tools to identify memory invariants.

This confirms that memory as structural invariance is a general property of learning systems, independent of specific architectures.

Part IV

Cosmic Scale: Memory of Spacetime

Chapter 9

Spacetime as Memory: An Elastic Landscape

Overview

This chapter extends the theory of memory as structural invariance to its broadest possible domain: spacetime itself. We argue that spacetime—whether described deterministically by general relativity or stochastically in quantum and semiclassical regimes—exhibits memory in precisely the sense developed throughout this book. Memory, at cosmic scales, manifests not as stored events but as persistent causal and topological structures that constrain the universe's future evolution.

9.1 Why spacetime must be considered

If memory is defined as a structural constraint induced by dynamical evolution, then any theory claiming universality must address spacetime. Spacetime is not merely the arena in which dynamics unfold; it is itself dynamical.

In general relativity, geometry responds to matter and energy. In quantum gravity and cosmology, geometry fluctuates. In both cases, spacetime bears the imprint of its history.

Central claim. Spacetime is a memory-bearing system because its structure constrains future causal possibilities based on past configurations.

9.2 Spacetime as a topological manifold

Let $(\mathcal{M}, g)$ denote spacetime as a smooth manifold $\mathcal{M}$ equipped with a Lorentzian metric g.

From the perspective of this book:

- $\mathcal{M}$ is the state space,
- g is a geometric realization,
- causal structure defines adjacency and reachability.

The topology of $\mathcal{M}$ encodes global properties such as connectivity, boundaries, and horizons, which persist under continuous deformations of the metric.

9.3 Deterministic memory in general relativity

In classical general relativity, spacetime evolution is governed by Einstein's field equations:

$$G_{\mu\nu} = 8\pi T_{\mu\nu}.$$

Although local geometry may change dynamically, several structures exhibit persistence:

9.3.1 Causal structure

The light-cone structure defines which events can influence which others. Once established, causal relationships constrain all future evolution.

9.3.2 Geodesic congruences

Families of geodesics encode information about curvature and past matter distributions. Their focusing and divergence are governed by Raychaudhuri-type equations, embedding memory of past curvature.

9.3.3 Horizons

Event horizons and apparent horizons partition spacetime into causally distinct regions. These partitions function as topological invariants under large classes of perturbations.

Interpretation. Spacetime remembers not events, but constraints on influence.

9.4 Stochastic spacetime and ensemble geometries

At quantum or semiclassical scales, spacetime cannot be treated as a single smooth geometry.

Instead, one considers:

- path integrals over metrics,
- ensembles of geometries,
- fluctuating causal structures.

Let $\{(\mathcal{M}_\omega, g_\omega)\}_{\omega\in\Omega}$ denote an ensemble of spacetime realizations.

Each realization induces a causal landscape $\mathcal{L}_\omega$.

Key observation. While metric details fluctuate, large-scale causal and topological features may persist across realizations.

This persistence is precisely the stochastic structural equivalence discussed in earlier chapters.

9.5 Cosmic memory invariants

9.5.1 Definition

Definition 9.5.1 (Cosmic Memory Invariant). *A cosmic memory invariant is an equivalence class of spacetime structures—topological or causal—that persists across admissible deterministic or stochastic evolutions.*

Examples include:

- global topology of spacetime,
- causal ordering relations,
- horizon structures,
- conserved homology classes.

9.5.2 Relation to energy conservation

In general curved spacetime, global energy conservation is not well-defined. However, causal and topological invariants remain meaningful.

This reinforces a central thesis of the book:

> Where energy fails to be globally conserved, structure persists.

9.6 Arrow of time as memory accumulation

Traditional accounts explain the arrow of time via entropy increase. While compatible, entropy alone does not explain why certain structures persist.

Memory-based account. The arrow of time emerges from the accumulation and persistence of memory invariants that break time-reversal symmetry at the structural level.

Each invariant constrains future possibilities asymmetrically.

9.7 Spacetime as a learning system

Under the Noether–Memory framework, spacetime itself can be viewed as a learning system:

- matter and energy act as inputs,
- geometry responds dynamically,
- persistent structures encode memory,
- future evolution is constrained.

Learning here does not imply intention, but structural adaptation.

9.8 Relation to biological and computational memory

The same abstract structure appears across scales:

$$\begin{array}{lcl} \text{Neural systems} & \leftrightarrow & \text{Attractor landscapes} \\ \text{Machine learning} & \leftrightarrow & \text{Basin partitions} \\ \text{Spacetime} & \leftrightarrow & \text{Causal topology} \end{array}$$

In all cases, memory is structural persistence under dynamics.

9.9 Limits and open problems

Several questions remain open:

- How to rigorously characterize stochastic equivalence classes of spacetime?
- What role do topology-changing processes play in forgetting?
- Can cosmic memory be quantified operationally?

These questions mark the boundary between established theory and future research.

9.10 Summary of Part VII

We have argued that:

- Spacetime qualifies as a memory-bearing system.
- Memory manifests as persistent causal and topological structures.
- Deterministic and stochastic descriptions fit naturally within the same framework.
- Energy conservation is a special case of deeper structural persistence.

This completes the extension of memory as structural invariance from artificial systems to the universe itself.

Chapter 10

Arrow of Time, Irreversibility, and Memory

Overview

The emergence of a temporal arrow remains one of the most persistent conceptual challenges in physics, cognitive science, and complex systems theory. While microscopic laws are often time-reversal symmetric, macroscopic phenomena display clear irreversibility. In this chapter, we argue that memory—understood as structural invariance—provides a unifying account of irreversibility across domains.

Rather than deriving the arrow of time solely from entropy or probabilistic asymmetry, we show that the accumulation and persistence of memory invariants impose directional constraints on system evolution.

10.1 Time symmetry and its limits

Many fundamental equations in physics are invariant under time reversal. Classical mechanics, electromagnetism, and even quantum mechanics (in isolation) do not privilege a temporal direction.

Yet real systems:

- age,
- learn,
- adapt,
- and remember.

These processes are intrinsically directional. The question is not merely why entropy increases, but why *structure persists asymmetrically in time.*

10.2 Entropy-based accounts revisited

Entropy provides a statistical account of irreversibility: macrostates evolve toward higher multiplicity. While correct, this account leaves open several issues:

- Entropy increase does not specify which structures persist.
- High entropy states may still encode long-lived structures.
- Entropy does not explain learning or adaptation.

Entropy measures dispersion; memory measures constraint.

10.3 Memory as directional constraint

In the framework developed throughout this book, memory invariants restrict admissible future configurations.

Definition 10.3.1 (Temporal Asymmetry Induced by Memory)**.** *A system exhibits an arrow of time if its set of admissible future landscapes is strictly constrained by memory invariants induced by past dynamics.*

This definition shifts focus:

- from probability distributions over states,
- to structural restrictions over landscape equivalence classes.

10.4 Irreversibility without dissipation

Not all irreversibility is dissipative. Learning systems, for instance, may:

- conserve energy locally,
- remain near equilibrium,
- yet undergo irreversible structural change.

Example. Once a learning system has selected a particular hyperspace of representations, returning to a prior hyperspace may be structurally inaccessible—even if microscopic reversibility holds.

This irreversibility is topological, not thermodynamic.

10.5 Structural accumulation and temporal depth

Memory invariants accumulate over time. Each invariant:

- restricts future transitions,
- increases structural depth,
- reduces symmetry between past and future.

Temporal depth. We define the temporal depth of a system as the number and complexity of memory invariants constraining its evolution.

Systems with greater temporal depth exhibit stronger arrows of time.

10.6 Learning as irreversible symmetry breaking

From the Noether–Memory perspective:

- learning breaks structural symmetries,
- establishes new equivalence classes,
- and thus induces irreversibility.

Once a memory invariant is formed, reversing learning would require a topological transition—an operation not generally available through local dynamics.

10.7 Relation to biological and cognitive time

In biological systems:

- development,
- maturation,
- and aging

are all memory-driven processes.

Cognitive time—the subjective sense of past and future—emerges from the ordering of memory invariants, not from clock time alone.

10.8 Relation to cosmological time

At cosmic scales, the universe accumulates:

- causal structures,
- horizon boundaries,
- topological features.

These features define a macroscopic arrow of time independent of local entropy fluctuations.

10.9 Comparison with other approaches

Entropy-based arrow	Statistical dispersion
Causal arrow	Light-cone asymmetry
Memory-based arrow	Structural persistence

These approaches are complementary. Memory provides the missing link between microscopic reversibility and macroscopic directionality.

10.10 Implications for reversibility and control

If irreversibility is structural:

- controlling a system requires modifying invariants,
- prediction requires identifying persistent structures,
- reversal requires topology change.

This has implications for:

- machine learning (catastrophic forgetting),
- neuroscience (plasticity limits),
- physics (cosmological phase transitions).

10.11 Summary of Part VIII

We have argued that:

- the arrow of time emerges from memory invariants,
- irreversibility is primarily structural, not energetic,
- learning induces symmetry breaking in landscape equivalence classes,
- entropy alone is insufficient to explain temporal direction.

Memory thus provides a unifying explanation for time's arrow across physical, biological, and computational systems.

Part V

Philosophical and Practical Implications

Chapter 11

Identity, Meaning, and Consciousness (Speculative)

Overview

Having developed a general theory of memory as structural invariance—spanning neural systems, machine learning, and spacetime—we now examine its philosophical implications. This chapter addresses three interrelated concepts: identity, meaning, and consciousness. Our aim is not to provide a reductive theory of mind, but to clarify how these concepts can be coherently reinterpreted within a structural and topological framework of memory.

11.1 Identity beyond substance

Classical metaphysics often treats identity as substance-based: an entity remains identical if some core essence persists. In contrast, modern science frequently defines identity through state continuity or informational content.

Both views encounter difficulties:

- substances change,
- states fluctuate,
- information degrades.

Within the present framework, identity is neither substance nor state.

Structural identity. An entity retains identity insofar as it preserves its memory invariants—i.e., the equivalence classes of landscapes that constrain its evolution.

11.2 Identity as equivalence class

Let $\mathfrak{M}$ denote the set of memory invariants characterizing a system.

Definition 11.2.1 (Structural Identity)**.** *Two systems are identical in the relevant sense if they belong to the same equivalence class of memory invariants.*

This definition naturally accommodates:

- gradual change,
- learning and development,
- partial replacement of components.

Identity is thus a homotopy-like concept: preserved under continuous deformation, but lost under topological transition.

11.3 Meaning as structural relation

Meaning is often treated as reference, representation, or symbol manipulation. However, these accounts presuppose a stable backdrop of interpretation.

From the perspective of memory as structural invariance:

Key claim. Meaning is not attached to isolated symbols or states, but emerges from stable relational structures among memory invariants.

11.3.1 Semantic landscapes

Consider a system that assigns significance to certain patterns. These patterns form basins in a semantic landscape. Meaning arises when:

- relations among patterns are stable,
- transitions among basins are constrained,
- and reinterpretation preserves structural coherence.

Thus, meaning is a property of a landscape, not of a point.

11.4 Evolution of meaning

As memory invariants evolve, so do semantic relations.

Interpretation. A change in worldview corresponds to a transition between classes of admissible semantic landscapes. Old meanings may persist structurally while acquiring new relational roles.

This view aligns with the observation that understanding deepens not by accumulating facts, but by reorganizing relations among them.

11.5 Consciousness as structural attunement

Consciousness is among the most contested concepts in philosophy and cognitive science. Rather than defining consciousness directly, we characterize its structural role.

Proposal. Consciousness is the capacity of a system to:

- navigate its memory landscape,
- detect structural invariants,
- and modulate its own hyperspace of admissible representations.

Conscious systems are not those that store more information, but those that can reorganize the constraints governing their future states.

11.6 Attention and awareness

Attention can be understood as the selective amplification of certain basins or invariant structures within the landscape.

Awareness corresponds to the system's sensitivity to transitions between equivalence classes—i.e., to moments when its own memory structure is at stake.

11.7 False meaning and illusion

Just as false memories are structurally valid, false meanings and illusions arise from stable but misaligned invariants.

The persistence of illusion is explained not by error, but by structural reinforcement.

11.8 Ethical and epistemic implications

If identity and meaning are structural:

- knowledge is the discovery of invariants,
- misunderstanding is the stabilization of misleading structures,
- transformation requires topology change.

Ethical growth, learning, and creativity thus involve deliberate engagement with memory invariants.

11.9 Human, artificial, and cosmic continuity

The same structural principles apply across scales:

- human identity,
- artificial agents,
- social systems,
- and the universe itself.

What differs is not the existence of memory, but the richness and flexibility of invariant structures.

11.10 Summary of Part IX

We have argued that:

- identity is structural, not substantial,
- meaning arises from persistent relational landscapes,
- consciousness is attunement to memory invariants,
- illusion and error are structurally explainable,
- and continuity across scales is natural within this framework.

This philosophical synthesis prepares the ground for the concluding chapter, which summarizes contributions and outlines future directions.

Chapter 12

Applications and Experiments

Chapter 13

Conclusions and Future Work

Overview

This book set out to answer a single foundational question: *how does memory arise naturally in complex systems?* We have argued that memory is neither a stored entity nor a numerical quantity to be optimized, but a form of *structural invariance*—a persistent constraint on the space of admissible futures induced by a system's dynamics.

In this concluding chapter, we summarize the core contributions, clarify the scope and limits of the framework, and outline directions for future research.

13.1 Summary of contributions

The main contributions of this work can be summarized as follows:

1. **Memory as structural invariance.** We redefined memory as a second-order topological object: an invariant not of states, but of equivalence classes of landscapes that constrain system evolution.

2. **Implicit landscapes without energy.** We demonstrated that landscapes need not be defined by scalar energy or loss functions; instead, they arise from invariant sets, basins of attraction, and their adjacency relations.

3. **Stochastic generalization.** By introducing ensembles of landscape realizations, we showed how memory persists statistically under stochastic dynamics, unifying deterministic and probabilistic systems.

4. **Noether–Memory principle.** We extracted the structural core of Noether's theorem and generalized it beyond classical symmetries and conserved quantities, linking structural equivalence to memory invariants.

5. **Computational realizations.** We derived Stochastic Topological Memory Networks (STMN) as an explicit algorithmic instantiation, and showed how classical neural networks, clustering, and manifold learning arise as special cases.

6. **Cosmic extension.** We extended the framework to spacetime, arguing that causal and topological structures function as cosmic memory invariants even when global energy conservation fails.

7. **Philosophical synthesis.** We reinterpreted identity, meaning, and consciousness as structural phenomena grounded in memory invariants, offering a unified language across disciplines.

Taken together, these results support a single thesis:

> Memory is the persistence of structure under change.

13.2 What this framework does not claim

To avoid overextension, it is important to state clearly what this framework does *not* claim:

- It does not claim that all systems possess memory in a psychologically or phenomenologically rich sense.
- It does not reduce consciousness to topology or dynamics alone.
- It does not replace thermodynamic or information-theoretic accounts of irreversibility, but complements them.
- It does not propose a new physical law in the traditional sense.

Rather, this work provides a unifying *structural language* within which diverse phenomena can be coherently related.

13.3 Empirical and computational directions

Several avenues for concrete investigation follow naturally from this framework:

13.3.1 Machine learning experiments

- Implement STMN architectures and compare robustness and continual learning performance against conventional neural networks.
- Use topological data analysis to identify memory invariants during training.
- Study the formation and dissolution of equivalence classes under curriculum learning.

13.3.2 Neuroscience and cognition

- Investigate whether neural plasticity preserves topological invariants across learning tasks.
- Analyze false memories as structurally stable but semantically misaligned invariants.
- Relate attention and awareness to transitions between memory equivalence classes.

13.3.3 Physics and cosmology

- Explore stochastic equivalence classes of spacetime geometries in semiclassical gravity.
- Investigate topology-changing processes as cosmic forgetting.
- Examine whether causal set approaches naturally encode memory invariants.

13.4 Theoretical extensions

Beyond immediate applications, several theoretical extensions suggest themselves:

- Formalizing memory invariants using category theory or higher-order homotopy.
- Developing quantitative measures of structural persistence.
- Extending the Noether–Memory principle to discrete and non-smooth systems.

These directions represent a research program rather than a closed theory.

13.5 A note on future work

Throughout this book, we deliberately restricted attention to the question of how memory arises from structure. Other profound questions—such as the relationship between memory and value, meaning and ethics, or memory and creativity—were intentionally left open.

These questions require additional conceptual tools and deserve independent treatment.

13.6 Closing reflection

The guiding intuition of this work is simple yet far-reaching: systems remember not by storing the past, but by being shaped by it. The past persists as constraint, the future unfolds within structure, and memory lives in between.

If this perspective proves fruitful, it will be because it allows researchers across disciplines to recognize a familiar pattern—structural persistence under change—beneath phenomena that once seemed unrelated.

Final statement. Memory is not a thing the universe contains. Memory is a way the universe continues.

Appendix A

Mathematical Background

Purpose of this appendix

This appendix provides the minimal mathematical background required to support the main claims of the book. The presentation is deliberately selective: we do not aim for completeness, but for conceptual sufficiency and internal consistency.

The focus is on four technical pillars:

1. Topological state spaces and quotient constructions,
2. Invariant sets and basins under deterministic dynamics,
3. Random dynamical systems and stochastic invariants,
4. Persistent homology and stability of topological features.

Readers already familiar with these topics may safely skim this appendix.
—

A.1 Topological spaces and continuity

A.1.1 Topological spaces

A *topological space* is a pair $(\mathcal{S}, \tau)$, where $\mathcal{S}$ is a set and τ is a collection of subsets of $\mathcal{S}$ satisfying:

1. $\emptyset, \mathcal{S} \in \tau$,
2. arbitrary unions of elements of τ belong to τ,
3. finite intersections of elements of τ belong to τ.

Elements of τ are called open sets. Throughout this book, state spaces are assumed to be at least Hausdorff and second-countable unless otherwise stated. These assumptions are mild and exclude pathological cases while preserving generality.

A.1.2 Continuous maps

A map $f : (\mathcal{S}, \tau) \to (\mathcal{S}', \tau')$ is continuous if the preimage of every open set in τ' is open in τ.

Continuity is the minimal requirement for discussing invariant sets and structural persistence; no differentiability or metric structure is assumed unless explicitly introduced.

—

A.2 Dynamical systems and invariant sets

A.2.1 Deterministic dynamics

A (deterministic) dynamical system is defined by a family of maps

$$\Phi_t : \mathcal{S} \to \mathcal{S}, \quad t \in \mathbb{R} \text{ or } \mathbb{Z},$$

satisfying:

$$\Phi_0 = \mathrm{id}, \quad \Phi_{t+s} = \Phi_t \circ \Phi_s.$$

We assume Φ_t is continuous for each t.

A.2.2 Invariant sets

Definition A.2.1 (Invariant set)**.** *A subset $M \subset \mathcal{S}$ is invariant under Φ_t if*

$$\Phi_t(M) = M \quad \textit{for all } t.$$

Invariant sets generalize fixed points, periodic orbits, invariant manifolds, and attractors.

A.2.3 Basins of attraction

Let M be an invariant set. Its basin of attraction is defined as

$$\mathcal{B}(M) = \left\{ x \in \mathcal{S} \mid \lim_{t \to \infty} \Phi_t(x) \in M \right\},$$

where the limit is understood in a topological sense (or metric sense if a metric is available).

Basins need not be open or disjoint; their adjacency relations form a key component of the implicit landscape structure discussed in the main text.

—

A.3 Quotient spaces and equivalence relations

A.3.1 Equivalence relations induced by dynamics

Let $\sim$ be an equivalence relation on $\mathcal{S}$. In the context of this book, $\sim$ is typically induced by dynamics:

$$x \sim y \iff x \text{ and } y \text{ belong to the same invariant structure or basin.}$$

A.3.2 Quotient topology

The quotient space $\mathcal{S}/\sim$ is the set of equivalence classes under $\sim$, equipped with the quotient topology: a set $U \subset \mathcal{S}/\sim$ is open if and only if its preimage under the canonical projection

$$\pi : \mathcal{S} \to \mathcal{S}/\sim$$

is open in $\mathcal{S}$.

Quotient constructions formalize the notion that encoding corresponds to identifying classes of states rather than individual states.

—

A.4 Random dynamical systems

A.4.1 Basic definition

A *random dynamical system* (RDS) consists of:

- a probability space $(\Omega, \mathcal{F}, \mathbb{P})$,
- a measurable flow $\theta_t : \Omega \to \Omega$,
- a cocycle mapping

  $$\Phi(t, \omega) : \mathcal{S} \to \mathcal{S}$$

  satisfying the cocycle property:

  $$\Phi(0, \omega) = \mathrm{id}, \quad \Phi(t + s, \omega) = \Phi(t, \theta_s\omega) \circ \Phi(s, \omega).$$

This formalism captures stochastic perturbations of deterministic systems.

A.4.2 Random invariant sets

A random set $M(\omega) \subset \mathcal{S}$ is invariant if

$$\Phi(t, \omega)M(\omega) = M(\theta_t\omega).$$

Random attractors are invariant random sets that attract bounded sets in probability or almost surely.

These notions underpin the definition of stochastic memory invariants as equivalence classes persisting across realizations.

—

A.5 Structural stability and equivalence

Two dynamical systems are said to be *topologically conjugate* if there exists a homeomorphism h mapping trajectories of one system to the other.

Structural stability refers to the persistence of qualitative dynamical features under small perturbations. While full structural stability is rare in high-dimensional systems, weaker notions (e.g., homology preservation or persistence stability) suffice for the purposes of this book.

—

A.6 Persistent homology

A.6.1 Simplicial complexes

Given a point cloud or graph, one can construct a simplicial complex (e.g., Vietoris–Rips or Čech complex) parameterized by a scale ϵ.

A.6.2 Filtrations

A filtration is a nested family of complexes:

$$\mathcal{K}_{\epsilon_1} \subseteq \mathcal{K}_{\epsilon_2} \subseteq \cdots$$

Topological features (connected components, loops, voids) appear and disappear along the filtration.

A.6.3 Persistence diagrams

Persistent homology tracks the birth and death of these features. The resulting persistence diagram encodes topological invariants that are robust to noise.

A.6.4 Stability theorem

A central result is the stability theorem: small perturbations of the input data lead to small changes (under bottleneck distance) in the persistence diagram.

This result provides the mathematical foundation for treating persistent topological features as operational memory invariants.

—

A.7 Relation to memory invariants

The constructions in this appendix justify several key claims of the main text:

- Encoding as quotient mapping is mathematically well-defined.
- Structural invariants persist under stochastic perturbations.
- Topological features extracted via persistent homology are stable and observable.
- Memory invariants can be treated independently of specific coordinates or metrics.

Further technical details and proofs may be found in the cited literature.

Appendix B

Implementation Notes

Purpose of this appendix

This appendix provides practical guidance for implementing the theoretical constructs introduced in the main text, with particular emphasis on *Stochastic Topological Memory Networks (STMN)*. The goal is not to prescribe a single algorithmic instantiation, but to delineate a family of implementable procedures consistent with the theory.

We focus on:

1. Parameterization of stochastic network structures,
2. Training objectives and estimators,
3. Sampling strategies and variance control,
4. Computational complexity considerations,
5. Practical diagnostics for memory invariants.

—

B.1 Parameterization of stochastic edges

B.1.1 Edge distributions

In STMN, each edge (i, j) is associated with a probability distribution rather than a fixed weight. A minimal and practical parameterization is:

$$W_{ij}(\omega) = Z_{ij}(\omega) \cdot A_{ij},$$

where:

- $Z_{ij} \sim \text{Bernoulli}(p_{ij})$ models edge activation,
- $A_{ij} \in \mathbb{R}$ is an amplitude parameter.

More expressive variants replace the Bernoulli distribution with:

- logistic-normal distributions,

- Gaussian mixtures,
- discrete distributions over a finite support.

The choice reflects a trade-off between expressivity and computational cost.

B.1.2 Learnable parameters

The learnable parameters are:

$$\Theta = \{p_{ij}, A_{ij}\}_{(i,j)\in E},$$

or, more generally, parameters of the chosen edge distributions. Importantly, learning acts on *distributions*, not on individual sampled realizations.

—

B.2 Sampling and forward propagation

Given a fixed parameter set Θ, a single forward pass proceeds as follows:

1. Sample a network realization G_ω by sampling each $W_{ij}(\omega)$,
2. Perform standard forward propagation on G_ω,
3. Record outputs and intermediate activations as needed.

In practice, multiple realizations $\omega_1, \ldots, \omega_K$ are sampled per minibatch to estimate expected quantities.

—

B.3 Training objectives

B.3.1 Expected loss minimization

Given a task-specific loss $\mathcal{L}$, the primary objective is:

$$\mathbb{E}_{\omega\sim\mathbb{P}_\Theta}[\mathcal{L}(G_\omega)].$$

This expectation is approximated via Monte Carlo sampling:

$$\hat{\mathcal{L}} = \frac{1}{K}\sum_{k=1}^{K}\mathcal{L}(G_{\omega_k}).$$

B.3.2 Structural regularization

To encourage stable memory invariants, additional regularization terms may be introduced:

- entropy penalties on $\{p_{ij}\}$ to control sparsity,
- penalties on variance of outputs across realizations,
- persistence-based regularizers computed on intermediate representations.

These terms bias learning toward structurally stable landscapes rather than brittle parameter configurations.

—

B.4 Gradient estimation

B.4.1 Score-function estimator

For discrete distributions (e.g., Bernoulli), gradients can be estimated using the score-function estimator:

$$\nabla_\theta \mathbb{E}_\omega[\mathcal{L}] = \mathbb{E}_\omega[\mathcal{L} \nabla_\theta \log p_\theta(\omega)].$$

While unbiased, this estimator often exhibits high variance.

B.4.2 Reparameterization and relaxations

To reduce variance, continuous relaxations may be employed:

- Gumbel–Softmax / Concrete distributions for Bernoulli edges,
- Gaussian reparameterization for continuous weights.

These techniques enable low-variance gradient estimates at the cost of introducing approximation bias.

—

B.5 Training loop (pseudocode)

A minimal training loop for STMN is outlined below.

```
Initialize distribution parameters \Theta
for epoch = 1 ... N:
    for minibatch B:
        Sample realizations {G_omega_k}_{k=1}^K
        for each G_omega_k:
            Compute forward pass and loss L_k
        Compute Monte Carlo loss \hat{L} = (1/K) \sum_k L_k
        Compute gradients \nabla_{\Theta} \hat{L}

        Update \Theta using optimizer (e.g., Adam)
```

This loop generalizes standard neural network training by replacing deterministic weights with stochastic structural sampling.

—

B.6 Computational complexity

Let:

- $|E|$ be the number of potential edges,
- K the number of samples per minibatch.

The computational cost per minibatch scales as:

$$\mathcal{O}(K \cdot |E|),$$

modulo task-specific forward costs.

Memory usage increases linearly with K, motivating small-sample regimes combined with variance-reduction techniques.

—

B.7 Diagnostics for memory invariants

Beyond task loss, STMN requires structural diagnostics:

- variance of outputs across realizations,
- clustering stability of internal representations,
- persistence diagrams computed on activation manifolds.

Stable memory corresponds to invariants that persist across samples and training epochs.

—

B.8 Relation to existing methods

STMN subsumes several known architectures as limiting cases:

- $p_{ij} \equiv 1$: deterministic neural networks,
- fixed p_{ij} with sampling: DropConnect-like models,
- learned distributions with variational objectives: Bayesian neural networks.

The distinguishing feature of STMN is that learning is explicitly oriented toward *structural persistence*, not merely predictive accuracy.

—

B.9 Practical considerations

- Small values of K (e.g., $K = 3$–10) often suffice in practice.
- Initializing p_{ij} near intermediate values (e.g., 0.3–0.7) avoids premature structural collapse.
- Monitoring both loss and structural diagnostics is essential to detect overfitting.

—

B.10 Limitations

This appendix does not address:

- global convergence guarantees,
- optimality of learned equivalence classes,
- scalability to extremely large graphs without sparsity assumptions.

These remain open research directions.

Appendix C

Supplementary Proofs

Purpose of this appendix

This appendix provides formal statements and proof sketches for key claims made throughout the book. The emphasis is on logical consistency rather than exhaustive mathematical generality. When full proofs are omitted, precise references or proof strategies are indicated.

The results are grouped according to their conceptual role:

- validity of memory invariants,
- forgetting as topological transition,
- persistence under stochastic perturbations,
- operational equivalence via persistent homology.

—

C.1 Structural validity of memory invariants

Lemma C.1.1 (Structural validity of false memories)**.** *Let $(\mathcal{S}, \Phi_t)$ be a dynamical system and let $\mathcal{M}$ denote the set of invariant structures induced by Φ_t. Any invariant structure $M \in \mathcal{M}$ constitutes a valid memory invariant regardless of its correspondence to external truth or semantics.*

Proof sketch. Memory invariants are defined purely structurally as persistent constraints on admissible futures. The dynamics does not encode semantic correctness; it encodes stability. Therefore, any invariant structure that constrains future trajectories qualifies as memory by definition. Semantic alignment is an external criterion and does not affect structural persistence. □

—

C.2 Forgetting as topological transition

Definition C.2.1 (Topological forgetting)**.** *A memory invariant M is said to be forgotten if the equivalence class it defines ceases to exist under a continuous deformation of the system dynamics.*

Proposition C.2.2 (Forgetting requires topology change)**.** *Let M be a memory invariant associated with an equivalence class of landscapes. Forgetting M requires a change in the topological equivalence class of the induced landscape.*

Proof sketch. If the landscape remains within the same topological equivalence class, there exists a homeomorphism preserving invariant sets and basin relations. Under such a transformation, M persists structurally. Therefore, forgetting cannot occur without violating topological equivalence, i.e., without a topology-changing transition. □

—

C.3 Persistence under stochastic dynamics

Proposition C.3.1 (Statistical persistence of memory)**.** *Let $(\Phi(t,\omega))$ be a random dynamical system inducing a family of landscapes $\{\mathcal{L}_\omega\}$. If the probability measure over ω concentrates on a single topological equivalence class with probability at least $1-\delta$, then the associated memory invariant persists statistically.*

Proof sketch. By assumption, for almost all realizations ω, the induced landscapes $\mathcal{L}_\omega$ are topologically equivalent. Hence, the quotient structure defining the memory invariant is preserved in probability. This defines statistical persistence independent of individual realizations. □

—

C.4 Operational equivalence via persistent homology

Definition C.4.1 (Persistence-based equivalence)**.** *Two landscapes $\mathcal{L}_1$ and $\mathcal{L}_2$ are persistence-equivalent if the bottleneck distance between their persistence diagrams is below a threshold ϵ.*

Proposition C.4.2 (Operational detectability of invariants)**.** *If two landscapes are persistence-equivalent for sufficiently small ϵ, then they belong to the same memory equivalence class up to homology.*

Justification. The stability theorem of persistent homology guarantees that small perturbations of the underlying data induce small perturbations in persistence diagrams. Therefore, persistence equivalence serves as an operational proxy for structural equivalence at the homological level. □

—

C.5 Equivalence class stability under learning

Proposition C.5.1 (Learning preserves memory under structural regularization)**.** *Let Θ_t denote learning dynamics over model parameters. If learning updates are constrained to remain within a structural regularization regime preserving topological equivalence, then memory invariants are preserved during learning.*

Proof sketch. Structural regularization penalizes transitions that would alter invariant sets or basin adjacency relations. Consequently, learning trajectories remain within the same equivalence class, preserving memory invariants. □

—

C.6 Arrow of time and invariant accumulation

Theorem C.6.1 (Memory-induced arrow of time)**.** *If a system accumulates memory invariants over time, then its dynamics exhibits an intrinsic temporal asymmetry.*

Proof sketch. Each newly formed invariant restricts the set of admissible future trajectories. Since past constraints cannot be retroactively removed without topology change, the space of futures becomes increasingly constrained, defining an arrow of time. □

—

C.7 Scope and limitations

The results in this appendix rely on:

- continuity of dynamics,
- existence of invariant sets,
- applicability of homological stability results.

They do not assume global optimality, differentiability, or metric completeness. Extensions to non-continuous or highly singular systems remain open problems.

—

Closing remark

This appendix establishes that the central claims of the book are not merely metaphorical, but rest on coherent mathematical principles. Where full proofs are omitted, the path toward formalization is explicit.

Appendix D

Applications and Experiments

Overview

This chapter demonstrates how the theoretical framework developed in this book can be operationalized and empirically explored. The goal is not to establish state-of-the-art performance benchmarks, but to validate the core claims:

- memory manifests as structural invariants,
- such invariants persist across stochastic realizations,
- and can be detected using topological and statistical tools.

We present a set of illustrative experiments spanning neural and non-neural systems.
—

D.1 Experiment I: Stochastic associative memory

D.1.1 Setup

We consider a small associative memory task inspired by classical Hopfield networks. A set of binary patterns $\{x^{(1)}, \dots, x^{(P)}\}$ is embedded in a network.

Instead of fixed weights, we employ a Stochastic Topological Memory Network (STMN) where each edge weight is sampled from a Bernoulli-amplitude distribution:

$$W_{ij}(\omega) = Z_{ij}(\omega) \cdot A_{ij}, \quad Z_{ij} \sim \text{Bernoulli}(p_{ij}).$$

Multiple network realizations are sampled at inference time.

D.1.2 Procedure

1. Initialize distributions $\{p_{ij}, A_{ij}\}$ randomly.
2. Train the network to minimize expected reconstruction error across realizations.
3. For each trained model, sample K realizations and simulate dynamics from random initial states.
4. Record attractors and basin memberships.

D.1.3 Observation

Although individual realizations differ in detailed dynamics, the set of attractors and their basin adjacency relations remain statistically stable across realizations.

D.1.4 Interpretation

This confirms that memory is not tied to a specific weight configuration, but to a structurally persistent equivalence class of landscapes.

—

D.2 Experiment II: Structural persistence under noise

D.2.1 Setup

A low-dimensional continuous dynamical system (e.g., a two-well potential system) is subjected to additive stochastic perturbations.

D.2.2 Procedure

1. Generate multiple noisy trajectories with different noise seeds.
2. Estimate invariant sets and basins from trajectory data.
3. Construct simplicial complexes from sampled state points.
4. Compute persistence diagrams for each realization.

D.2.3 Results

Persistence diagrams show consistent long-lived features corresponding to dominant basins, despite fluctuations in short-lived topological noise.

D.2.4 Interpretation

Persistent homological features provide an operational signature of memory invariants under stochastic dynamics.

—

D.3 Experiment III: Clustering as memory encoding

D.3.1 Setup

We generate synthetic data consisting of multiple clusters with varying density and noise levels.

D.3.2 Procedure

1. Apply clustering algorithms (e.g., k-means, DBSCAN) across a range of hyperparameters.

2. For each clustering result, construct a graph-based representation of cluster adjacency.

3. Compute persistence diagrams over cluster connectivity graphs.

D.3.3 Results

Clusters that are stable across hyperparameter variations correspond to persistent topological features. Transient clusters appear as short-lived features.

D.3.4 Interpretation

Stable cluster cores function as memory invariants: they encode persistent structural information about the data distribution rather than individual data points.

—

D.4 Experiment IV: Learning dynamics and memory preservation

D.4.1 Setup

An STMN is trained sequentially on two related tasks (continual learning setting).

D.4.2 Procedure

1. Train the network on Task A.

2. Continue training on Task B with and without structural regularization.

3. Monitor task performance and structural diagnostics.

D.4.3 Results

Without structural regularization, learning induces topology-changing transitions leading to forgetting. With regularization, task performance on Task A is preserved, and topological invariants remain stable.

D.4.4 Interpretation

Catastrophic forgetting corresponds to loss of memory invariants, while continual learning corresponds to preserving equivalence classes of landscapes.

—

D.5 Metrics and diagnostics

Across all experiments, we employ the following diagnostics:

- variance of outputs across stochastic realizations,
- stability of attractor counts and basin relations,
- bottleneck distance between persistence diagrams,
- task-level performance metrics (where applicable).

Structural persistence is indicated by low variance and stable topological signatures despite stochasticity.

—

D.6 Limitations of the experiments

The experiments presented here are intentionally minimal:

- they do not address large-scale benchmarks,
- they do not optimize for predictive accuracy,
- they rely on low-dimensional or synthetic settings.

Their purpose is conceptual validation rather than performance competition.

—

D.7 Reproducibility

All experiments are designed to be reproducible using standard numerical libraries and topological data analysis toolkits. Reference implementations and scripts are provided in the accompanying repository.

—

D.8 Summary

These experiments collectively support the central thesis of the book:

- memory manifests as structural invariants,
- such invariants persist under stochastic dynamics,
- and can be detected and analyzed using topological tools.

They demonstrate that the proposed framework is not merely philosophical, but empirically accessible.

Bibliography

D. H. Ackley, G. E. Hinton, and T. J. Sejnowski. A learning algorithm for boltzmann machines. *Cognitive Science*, 9(1):147–169, 1985. URL `https://www.cs.toronto.edu/~fritz/absps/cogscibm.pdf`.

L. Arnold. *Random Dynamical Systems*. Springer, 1998.

Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46 (2):255–308, 2009. doi: 10.1090/S0273-0979-09-01249-X.

H. Edelsbrunner and J. Harer. *Computational Topology: An Introduction*. American Mathematical Society, 2010. doi: 10.1090/mbk/069.

J. 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. URL `https://www.pnas.org/doi/10.1073/pnas.79.8.2554`.

Radford M. Neal. *Bayesian Learning for Neural Networks*. Springer, 1996. Available as technical report and book; classic reference on Bayesian NNs.

Emmy Noether. Invariante variationsprobleme. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-physikalische Klasse*, pages 235–257, 1918. URL `https://web.stanford.edu/~cantwell/AA218_Course_Material/Resources/Invariant_Variation_Problems_Emmy_Noether.pdf`. English translation available: "Invariant Variation Problems". See e.g. Delphenich translation / Stanford copy.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014. URL `https://jmlr.org/papers/v15/srivastava14a.html`.