STABLE STATES AS BLOCK-UNIFORM FLIPS OF THE SYMBOLIC CENTROID IN BINARY HOPFIELD NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing analyses of spurious attractors in binary Hopfield neural networks (HNNs) have largely been developed within statistical-mechanical frameworks, often under specific memory ensembles, loading regimes, or structured storage configurations. While these approaches provide important macroscopic insights, they do not directly yield an exact structural characterization of spurious attractors for a given memory realization and local field configuration, particularly with respect to the boundaries of collective block flips. In this work we show that, for $M \geq 3$ stored memories, **stable states in binary HNNs are block-uniform flips of the symbolic centroid**, the bit-wise majority of the stored memories. By partitioning neurons into local domains according to their memory-alignment patterns, we prove that all neurons in a block share the same alignment with the centroid, so *stability is decided block by block rather than neuron by neuron*. Reformulating the weight matrix accordingly, we derive a matrix-based stability criterion for identifying stable block configurations and characterizing spurious attractors. This reduces the analysis **from N neuron-level conditions to a finite set of conditions over the P memory-induced blocks**. Beyond this reduction, the block variables open a route toward a block-macrostate network in which variable-assignment methods may be able to determine basins of attraction exactly, laying the groundwork for the precise identification and elimination of spurious attractors.

1 INTRODUCTION

The Hopfield neural network (HNN) is a foundational framework for content-addressable memory, formalizing associative memory through statistical mechanics (Hopfield, 1982; 1984). By modeling memories as stable fixed points, that is, local minima of a global energy function, HNNs tie neural dynamics to energy-based optimization.

The classical framework is nevertheless limited by *spurious attractors*: stable states that correspond to no stored memory and typically arise as mixtures of several memories. They are a primary obstacle to reaching the theoretical storage capacity: they distort the energy landscape, capture retrieval trajectories, and degrade recall as the number of memories grows.

Spurious attractors also obstruct sequential learning. Early biological and computational studies observed that they act as parasitic sinks and give rise to catastrophic interference (Crick & Mitchison, 1983; Hopfield et al., 1983), motivating unlearning and "dreaming" strategies that reshape the energy landscape through stochastic sampling or Hebbian unlearning rules (Bruck & Roychowdhury, 1990; Van Hemmen, 1997; Robins & McCallum, 1998). These approaches remain heuristic: they lack a criterion separating genuine memories from spurious mixtures, and they predict little about when or why such states emerge. Recent work revisits spurious attractors post hoc, from energy profiles or robustness metrics (Behera et al., 2025; McAlister et al., 2025; Robins & McCallum, 2004), yet one important issue remains under-explored.

Research question. Given M stored memories, can the emergence of spurious attractors **be predicted via algebraic criteria**, without dynamical simulation, stochastic sampling or heuristic energy ratios?

The question is already salient in the simplest nontrivial setting: for three stored memories $\{\xi^1, \xi^2, \xi^3\}$ and their symbolic centroid ζ obtained via majority voting, a stable ζ creates a "shadow" in the state

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

space that intercepts convergence trajectories intended for the stored memories, **so that anticipating its stability a priori would enable principled memory-set design and proactive suppression of interference.** Existing stability analyses, however, rest predominantly on global Lyapunov arguments, so that the combinatorial structure by which the stored memories determine these states remains implicit.

In this work we develop a structural characterization of spurious attractors for binary HNNs with $M \geq 3$ stored memories. Rather than treating their emergence as a black box, we decompose the neurons by bit-level signatures, which record each neuron's memory-alignment pattern across the stored memories. The signatures thus partition the neuron space into P blocks capturing the combinatorial structure of the memory set, and stability is characterized by explicit algebraic conditions on that structure. *The decomposition is not a preprocessing step but a change of coordinates*: since neurons of a block share their signature, the full interaction structure is carried by a signature matrix of rank at most M, and the weights it induces are encoded in a symmetric $\mathbf{\Gamma}$-matrix of inter-block interactions. $\mathbf{\Gamma}$ is the complete descriptor of the inter-block interaction coefficients; the stability of a given block configuration is decided by the signed block quantities $(\mathbf{\Gamma}, \boldsymbol{L}, \boldsymbol{\sigma})$ of Eq. 8. At any stable state, block uniformity reduces stability to P block-level inequalities $\mathbf{\Gamma}^{\boldsymbol{\sigma}} \boldsymbol{L} \succeq M \mathbf{1}_P$ in place of the N neuron-level conditions.

Research answer. Yes. For any fixed set of stored memories, the spurious attractors are exactly the block configurations that satisfy Eq. 8 but correspond to no stored memory (Eq. 9); computing them requires no simulation of the dynamics: the block structure and $\mathbf{\Gamma}$ follow from the memories in $O(MN + P^2M)$ time, and a single candidate is certified in $O(PM)$ directly from the signatures, or in $O(P)$ once $\mathbf{\Gamma}$ is stored. For $M = 3$ the criterion recovers the known class of spurious attractors: every one of them is a signed superposition $\mathrm{sign}(\pm\boldsymbol{\xi}^1 \pm \boldsymbol{\xi}^2 \pm \boldsymbol{\xi}^3)$ of the stored memories, so that at most 8 can coexist.

Our contributions can be summarized as follows:

- **Signature-Based Structural Decomposition.** A bit-level signature representation induces a **canonical, permutation-invariant block decomposition** of the neuron space, *determined by the stored memories* up to a global sign of each block's alignment pattern ($P \leq \min(2^{M-1}, N)$, Appendix F.4); it is valid for general $M \geq 3$, and the quantities entering the stability criterion are invariant under that sign (Proposition F.10).
- **Block-Uniform Flip Property. Every stable state is uniform within each block**: $\zeta_i x_i$ is constant on A_p, so *partial flips within a block can never be stable* (Theorem 4.2), and a stable state is exactly a binary vector $\boldsymbol{\sigma} \in \{-1, +1\}^P$ over blocks.
- **$\mathbf{\Gamma}$-Matrix Characterization of Stability and Dynamics.** Stability reduces to block-level algebraic conditions on the symmetric interaction matrix $\mathbf{\Gamma}$, bypassing energy-based analyses. Since the search is restricted to **the 2^P block configurations rather than the 2^N neuron-level states**, each candidate is tested by one multilinear inequality in $O(P)$, and the complete spurious set is obtained by enumerating those configurations, with no simulation of the dynamics. The same block quantities govern the dynamics: synchronous and asynchronous updating share the fixed-point set of Eq. 8 (Proposition 4.6), and the asynchronous dynamics terminates in that set (Section 4.5).

Together, these results turn spurious attractors in HNNs from an empirical phenomenon into a predictable algebraic characterization: they follow from the block structure of the stored memories, opening a principled route to suppressing interference. Since asynchronous updating acts on the block variables as well (Section 4.5), and Appendix H removes a certified spurious attractor directly from them, the block quantities lay the groundwork for a block-macrostate network in which variable-assignment methods could determine basins of attraction exactly.

The remainder of the paper is organized as follows. Section 3 fixes the model, the update rules and the notation; Section 4.1 introduces the block decomposition, Theorem 4.2 states the Block-Uniform Flip Property, and Section 4.3 reduces stability to the $\mathbf{\Gamma}$-criterion of Eq. 8. Section 4.4 characterizes the spurious attractors, confirmed exactly by the $M = 3$ case of Theorem 4.5, and Section 4.5 carries the same block quantities into the two update schedules. Section 5 records the limitations and the directions the framework opens, and Appendix H applies the criterion to the removal of a certified spurious attractor.

2 Related Work

Foundations and Statistical Mechanics. The HNN was introduced as a paradigmatic content-addressable memory, connecting neural computation to spin-glass theory by casting retrieval as energy minimization in a high-dimensional landscape (Hopfield, 1982; 1984). Later work embedded it in statistical mechanics and probabilistic modeling, revealing formal equivalences with Boltzmann-type models under suitable assumptions (Barra et al., 2012; Agliari et al., 2013; 2019). These connections have resurfaced in modern deep learning, where attention-based energy models reinterpret HNN dynamics through Gaussian–Bernoulli restricted Boltzmann machines and denoising autoencoders (Ota & Karakida, 2023).

Evolution of Memory Capacity. Storage capacity has been a central theoretical concern since the model's inception. Early estimates of roughly $0.14N$ patterns were refined by information-theoretic and probabilistic analyses characterizing extensive storage regimes and thresholds for error-free retrieval (Abu-Mostafa & St-Jacques, 1985; McEliece et al., 1987; Kanter & Sompolinsky, 1987). Classical binary HNNs have polynomially bounded capacity, but dense associative memories (DAM) and modified learning or update rules attain super-linear and even exponential scaling in specific regimes (Storkey, 1997; Krotov & Hopfield, 2016; Agliari & De Marzo, 2020; Ramsauer et al., 2021), and recent work provides tight capacity bounds and convergence guarantees across binary, modern, and spherical variants (Krotov & Hopfield, 2021; Bao et al., 2022; Bao & Zhao, 2025).

Dynamics, Convergence, and Stability. Convergence is traditionally analyzed through Lyapunov functions, with symmetric weight matrices ensuring monotonic energy descent to attractors (Peretto, 1984). Beyond idealized settings, studies have documented multi-step convergence, limit cycles, and spurious attractors coexisting with intended memory states (McEliece et al., 1987; Baldi & Venkatesh, 1987; Komlós & Paturi, 1988), while noisy, diluted, or asymmetric extensions reveal phase transitions between stable retrieval, oscillatory dynamics, and chaos (Folli et al., 2018; Hwang et al., 2019). In contrast to these ensemble-level analyses, the present study fixes a single memory realization: neurons sharing the same alignment pattern are grouped into blocks, the N neuron-level conditions collapse to P block-level inequalities, and the spurious attractors are precisely the stable block configurations that correspond to no stored memory. The two update schedules share the same fixed-point set, while differing at the level of trajectories. A complementary line of work asks where trajectories terminate: basins of attraction have been measured for the Hopfield and the Storkey learning rules (Keeler, 1986; Storkey & Valabregue, 1999), and spurious attractors targeted by suppression and dreaming schemes (Abe, 1993; Fachechi et al., 2019). These approaches are empirical or asymptotic; the block-level representation developed here is intended to supply the state space for an exact basin description.

Overlap-Based Stability and Spurious Mixtures. Spurious attractors were originally studied through overlaps among the stored patterns: in the replica and mean-field treatments of the Hopfield model, stability is an inequality on order parameters whose solutions include symmetric-mixture states, spurious by construction (Amit et al., 1985; 1987; Amit & Amit, 1989); subsequent work makes those mixtures explicit as sign combinations of a small number of memories (Gardner & Derrida, 1988; Bruck & Roychowdhury, 1990), and for memory sets of equal pairwise Hamming distance the centroid can become stable while the memories are annihilated (Kumar et al., 1996). Recent asymptotic work gives conditions on the growth of M under which such mixtures are exact local minima in the classical, dense, and modern Hopfield models, and conjectures that they exhaust the local minima (Gayrard, 2025). The criterion developed here is of the same overlap type, but is stated for one finite memory realization and evaluated in closed form: the coefficients Γ_{pq} and the block sizes $\boldsymbol{L}$ carry the overlap structure, and **the block uniformity of stable states makes the criterion finite over 2^P configurations rather than 2^N states**, which is what allows the completeness of the mixture class to be decided exactly at $M = 3$ and the robustness of retrieval to be assessed a priori (Gorman et al., 2017).

Modern Frontiers and Interdisciplinary Synthesis. HNNs have since seen a resurgence across neuroscience, machine learning, and physics: associative memory dynamics model learning and recall in hippocampal and insect circuits (Tyulmankov et al., 2021; Liang et al., 2021; Rolls, 2024); modern variants handle tabular and sparse data (Schäfl et al., 2023; Xu et al., 2024); sparse interactions and U-Hop$^+$ updates address scalability while retaining capacity guarantees (Hu et al., 2023; 2024a;b); and complementary theory has clarified expressive power, circuit complexity, and gradient-based learning (Li et al., 2024; Cabannes et al., 2024). Continuous-time and memristive implementations (Santos et al., 2025; Niu et al., 2024), together with work on attractor structure and nonequilibrium dynamics

(McAlister et al., 2025; Behera et al., 2025), have kept HNNs a unifying framework. Recent work has further broadened their dynamics and architectures, from universal and dense associative memories and Bayesian adaptive variants (Millidge et al., 2022; Mimura et al., 2026; Wang et al., 2026a) to parameter-free Hopfield attention, vision-transformer attractor dynamics, and manifold-constrained models (Masumura & Taki, 2026; Xia & Jiang, 2026; Bogakovskaya et al., 2026; Wang et al., 2026b). These developments, however, still primarily regard individual feature vectors as the basic memory units.

3 PRELIMINARIES

3.1 BINARY HOPFIELD NEURAL NETWORK

Consider a binary HNN with N neurons, each taking values in $\{-1, +1\}$. The global state $\boldsymbol{x} \in \mathcal{X} = \{-1, +1\}^N$ evolves by updating neurons under a local-field rule, either synchronously (all neurons at once, defining a map $T : \mathcal{X} \to \mathcal{X}$) or asynchronously (one neuron at a time). Using Hebbian learning on M stored memories $\{\xi^1, \ldots, \xi^M\} \subset \mathcal{X}$, the symmetric weight matrix $W \in \mathbb{R}^{N \times N}$ (with zero diagonal to remove self-coupling) is

$$W = \frac{1}{N} \left(\sum_{m=1}^{M} \xi^m \xi^{m\top} - M I_N \right), \text{ where } I_N \text{ is the } N \times N \text{ identity.} \tag{1}$$

Both schedules update each neuron by the same local-field rule: for neuron i at time $t + 1$,

$$x_i^{(t+1)} = \begin{cases} +1, & H_i(t) > 0, \\ -1, & H_i(t) < 0, \\ x_i^{(t)}, & H_i(t) = 0, \end{cases} \quad \text{with} \quad H_i(t) = \sum_{j=1}^{N} w_{ij} x_j^{(t)}. \tag{2}$$

Since the zero diagonal of W keeps H_i independent of x_i, the two schedules share the same fixed points, and stability is a property of the state alone; the asynchronous case is analyzed in Section 4.5. A notation glossary is provided in Appendix A.

3.2 DEFINITIONS OF ATTRACTORS AND HAMMING DISTANCE

Definition 3.1 (Attractor)**.** A state $\boldsymbol{x}^* \in \mathcal{X}$ is an **attractor** (or fixed point) if every neuron is aligned with its local field, $x_i H_i \geq 0$ for all i, so that no update changes it under either schedule, and $\mathcal{A} \coloneqq \{\boldsymbol{x} \in \mathcal{X} \mid x_i H_i \geq 0 \text{ for all } i\}$ denotes the set of all stable states.

Definition 3.2 (Stored vs. Spurious Attractors)**.** Given a set of M stored memories $\{\xi^m\}_{m=1}^M \subset \mathcal{X}$, the set of **stored attractors** $\mathcal{A}_{\text{mem}} \coloneqq \{\xi^m, -\xi^m \mid m = 1, \ldots, M\}$ is the union of the memories and their bit-wise negations, reflecting the inherent bipolar symmetry of the energy function. We assume throughout that the M memories are distinct up to this symmetry, that is, $\xi^m \neq \pm\xi^{m'}$ for $m \neq m'$: a pair with $\xi^m = -\xi^{m'}$ contributes a single attractor pair, so such a configuration is more accurately described as storing fewer than M patterns.

An attractor $\boldsymbol{x}^*$ is a **spurious attractor** if it is stable but does not correspond to any stored memory, so that the set of spurious attractors is the difference $\mathcal{A}_{\text{sp}} \coloneqq \mathcal{A} \setminus \mathcal{A}_{\text{mem}}$.

Definition 3.3 (Retrieval and Convergence)**.** For any initial probe state $\boldsymbol{x}^0 \in \mathcal{X}$, the retrieval process is defined by iterating the update rule from $\boldsymbol{x}^0$, under either schedule. Convergence is achieved when the trajectory enters a fixed point $\boldsymbol{x}^* \in \mathcal{A}$. A retrieval is considered successful if $\boldsymbol{x}^* \in \mathcal{A}_{\text{mem}}$, and a failure if $\boldsymbol{x}^* \in \mathcal{A}_{\text{sp}}$, implying that the network has been trapped in a spurious attractor.

Definition 3.4 (Symbolic Centroid)**.** Given a set of M stored memories $\{\xi^1, \ldots, \xi^M\} \subset \{-1, +1\}^N$, the *symbolic centroid* $\boldsymbol{\zeta} \in \{-1, +1\}^N$ is defined component-wise as

$$\zeta_i \coloneqq \operatorname{sign}\left(\sum_{m=1}^{M} \xi_i^m \right), \quad i = 1, \ldots, N, \tag{3}$$

where $\operatorname{sign}(z) = +1$ for $z > 0$ and -1 for $z < 0$. This corresponds to the bit-wise majority vote. When the vote is tied, i.e., $\sum_{m=1}^{M} \xi_i^m = 0$ (which can only occur for even M), we adopt the

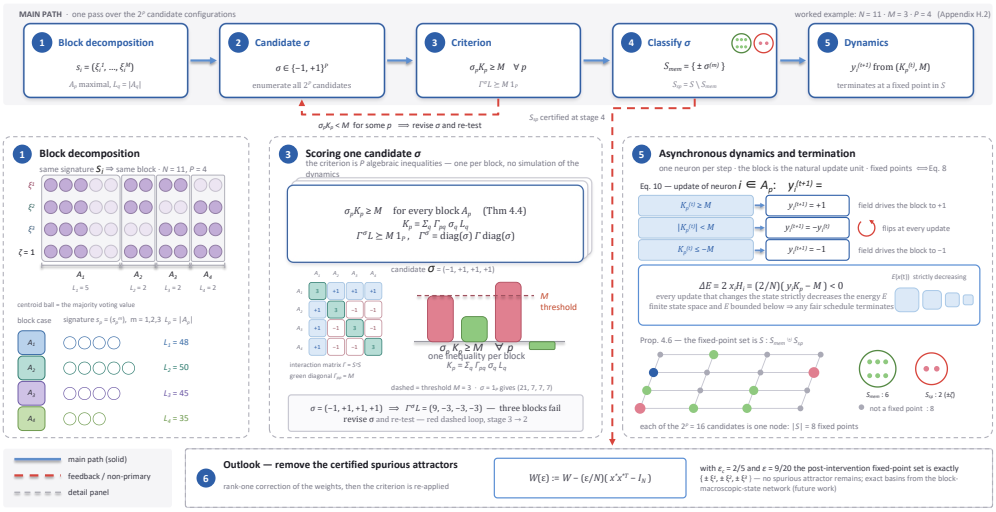


Figure 1: Overview of the block-level framework, on the $N = 11$, $M = 3$, $P = 4$ example of Appendix H. Candidate block configurations are scored by Eq. 8 without simulating the dynamics, and a certified spurious attractor is deleted by the rank-one correction of Eq. H.1.

tie-breaking rule ($\zeta_i = \xi_i^1$), where ξ^1 is the first stored memory. This convention ensures that the symbolic centroid is uniquely defined for every memory combination and every neuron index, and any rule that resolves a tie by a stored memory changes $\boldsymbol{\zeta}$ only by a block-uniform sign, under which the block decomposition and the criterion of Eq. 8 are invariant (Proposition F.10).

Definition 3.5 (Hamming Distance)**.** For two binary vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{X}$, their Hamming distance $d(\boldsymbol{u}, \boldsymbol{v}) := \frac{1}{2}\sum_{i=1}^{N} |u_i - v_i|$ counts the positions with different corresponding components.

Definition 3.6 (Block-wise Hamming Distance)**.** Let x and ζ be equal-length sequences partitioned into P blocks (Section 4.1), and let d_p denote the Hamming distance of x and ζ restricted to the p-th block, for $p = 1, \ldots, P$. The overall Hamming distance satisfies $d(x, \zeta) = \sum_{p=1}^{P} d_p$, and we refer to d_p as the block-wise Hamming distance for the p-th block.

4 Structural Decomposition Framework

4.1 Motivation and Block Decomposition

Aligning neurons with the stored memories shows that the coupling between two blocks is carried by a single coefficient, the individual neuron indices entering only through the signs of the symbolic centroid $\boldsymbol{\zeta}$. The neuron-level dynamics therefore admit a natural block representation, in which neurons with identical memory-alignment patterns form a block and inter-block interactions are fixed by the stored information. We formalize this observation through a **block decomposition induced by the stored memories**, which provides the framework for analyzing spurious attractors in the general case of $M \geq 3$. Fig. 1 gives an overview of the resulting pipeline, from the block decomposition to the removal of a certified spurious attractor, on the example of Appendix H.

Definition 4.1 (Block)**.** Given the stored memories $(\xi^1, \ldots, \xi^M)$, a **block** A_p is a maximal subset of neuron indices whose memory-alignment patterns $(\xi_i^1, \ldots, \xi_i^M)$ agree *up to an overall sign*, for $p = 1, \ldots, P$. Together the blocks form a partition of the neuron set.

Fig. 2 illustrates this procedure on a concrete instance: the per-bit majority yields $\boldsymbol{\zeta}$, and the neuron alignment patterns, identified up to an overall sign, yield the block partition.

By Eq. 3 and Definition 4.1, w_{ij} admits, for $i \in A_p$ and $j \in A_q$, the centroid-based representation in terms of the block-level factor Γ_{pq} (Eq. D.1):

$$w_{ij} = \frac{1}{N}\sum_{m=1}^{M} \xi_i^m \xi_j^m := \frac{\zeta_i \zeta_j}{N} \Gamma_{pq}. \tag{4}$$

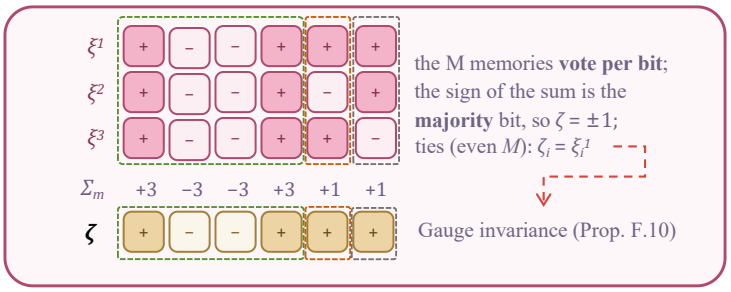


Figure 2: Block decomposition of a concrete instance ($N = 6$, $M = 3$). The per-bit majority gives the symbolic centroid $\boldsymbol{\zeta}$, and the dashed boxes group the neurons by alignment pattern up to an overall sign: $A_1 = \{1, 2, 3, 4\}$, $A_2 = \{5\}$, $A_3 = \{6\}$, so that $P = 3$ and $\boldsymbol{L} = (4, 1, 1)^\mathsf{T}$.

For $i \in A_p$ and $j \in A_q$, the m-th memory components follow the decomposition $\xi_i^m := s_p^m \zeta_i$ and $\xi_j^m := s_q^m \zeta_j$, where the terms s_p^m and s_q^m are block-specific coefficients for the m-th memory, independent of individual neurons.

Eq. 4 allows the local field H_i to be decomposed into block-wise contributions. For $i \in A_p$, the within-block term is $\sum_{j \in A_p, j \neq i} w_{ij} x_j = \frac{\zeta_i}{N} \Gamma_{pp} (L_p - 2d_p - \zeta_i x_i)$, and the contribution of any other block is $\sum_{j \in A_q, q \neq p} w_{ij} x_j = \frac{\zeta_i}{N} \Gamma_{pq} \sum_{j \in A_q} \zeta_j x_j = \frac{\zeta_i}{N} \Gamma_{pq} (L_q - 2d_q)$, so that

$$H_i = \sum_{q=1}^{P} \sum_{j \in A_q, j \neq i} w_{ij} x_j = \frac{\zeta_i}{N} [\sum_{q=1}^{P} \Gamma_{pq} (L_q - 2d_q) - M \zeta_i x_i], \quad i \in A_p, \tag{5}$$

where $L_q := |A_q|$ is the block size, that is, the number of neurons contained in block A_q, and d_q stands for the block-wise Hamming distance (Definition 3.6). This identity rewrites the sum $\sum_{j \in A_q} \zeta_j x_j$ in terms of the block size and the block-wise Hamming distance.

For convenience, define the block-level constant

$$K_p := \sum_{q=1}^{P} \Gamma_{pq} (L_q - 2d_q). \tag{6}$$

For a stable state $\boldsymbol{x}$, K_p is shared by all neurons in A_p, so that Eq. 5 becomes

$$H_i = \frac{\zeta_i}{N} \left(K_p - M \zeta_i x_i \right), \qquad i \in A_p. \tag{7}$$

4.2 Block-Uniform Flip Property

Combining the characterization of H_i with the block partition reveals that **neurons belonging to the same block must undergo the same flip at a stable state**, which yields the Block-Uniform Flip Property (see Appendix D.1 for the proof).

Theorem 4.2 (Block-Uniform Flip Property)**.** *Consider a binary HNN with $M \geq 3$ stored memories, and let $\boldsymbol{x}$ be a stable state. For any block A_p, **the quantity $\zeta_i x_i$ is constant within the block**, $\zeta_i x_i = \zeta_j x_j$ for all $i, j \in A_p$; equivalently, $x_i = \zeta_i \sigma_p$ for all $i \in A_p$ and some $\sigma_p \in \{-1, +1\}$.*

Thus, at a stable state the block-wise Hamming distance is determined directly by the block flip state, $d_q = 0$ if $\sigma_q = +1$ and $d_q = L_q$ if $\sigma_q = -1$, that is, $L_q - 2d_q = \sigma_q L_q$. Substituting this into Eq. 6 gives $K_p = \sum_{q=1}^{P} \Gamma_{pq} \sigma_q L_q$, so that **although $\boldsymbol{x}$ is defined at the neuron level, its stable configuration is characterized by the block-level binary vector $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_P)^\mathsf{T} \in \{-1, +1\}^P$.**

Remark 4.3 (**Set-level content of stability**). The Block-Uniform Flip Property imposes a structural constraint on the stable configurations. Although neurons within the same block may differ during transient evolution, they cannot maintain different flips at a stable state: **all neurons of a block share the same alignment with the reference sequence ζ and are therefore described by a single binary variable σ_p.** *Each block thus acts as a collective degree of freedom*, which is *the structural basis for reducing the neuron-level stability problem to a block-level representation.*

4.3 BLOCK-LEVEL STABILITY CRITERION

We now derive the stability condition in terms of the block-level configuration $\boldsymbol{\sigma}$. The proof is given in Appendix E.1.

Theorem 4.4 (Block-Level Stability Condition)**.** *For $M \geq 3$ stored memories, at a stable state characterized by the block configuration $\boldsymbol{\sigma} \in \{-1,+1\}^P$, the stability of every neuron $i \in A_p$ is equivalent to $\sigma_p K_p \geq M$ for $p = 1, \ldots, P$.*

Substituting $K_p = \sum_q \Gamma_{pq}\sigma_q L_q$ into the block-level stability condition gives

$$\boldsymbol{\Gamma}^{\boldsymbol{\sigma}}\boldsymbol{L} \succeq M\mathbf{1}_P, \qquad (\succeq \text{ is component-wise}) \tag{8}$$

where $\boldsymbol{\Gamma}^{\boldsymbol{\sigma}} = \mathrm{diag}(\boldsymbol{\sigma})\boldsymbol{\Gamma}\,\mathrm{diag}(\boldsymbol{\sigma})$ and $\boldsymbol{L} = (L_1, \ldots, L_P)^\mathsf{T}$, so that $(\boldsymbol{\Gamma}^{\boldsymbol{\sigma}}\boldsymbol{L})_p = \sum_{q=1}^P \sigma_p\Gamma_{pq}\sigma_q L_q = \sigma_p K_p$. Since $\Gamma_{pp} = M$, the self-block contribution to $\sigma_p K_p$ is $\Gamma_{pp}L_p$, and the remaining term $\sigma_p \sum_{q\neq p}\Gamma_{pq}\sigma_q L_q$ measures the inter-block support a configuration needs to remain stable.

Thus, **the N neuron-level stability conditions are reduced to P block-level inequalities**: the stability of a state is determined entirely by the block configuration $\boldsymbol{\sigma}$, the block-size vector $\boldsymbol{L}$, and the interaction matrix $\boldsymbol{\Gamma}$. Accordingly, the set of stable block configurations is $\mathcal{S} \coloneqq \{\boldsymbol{\sigma} \in \{-1,+1\}^P : \boldsymbol{\Gamma}^{\boldsymbol{\sigma}}\boldsymbol{L} \succeq M\mathbf{1}_P\}$, *which corresponds to the neuron-level stable states through $x_i = \zeta_i\sigma_p$* for $i \in A_p$.

4.4 BLOCK-LEVEL STRUCTURE OF SPURIOUS ATTRACTORS

The preceding results characterize all stable states in terms of the block configuration $\boldsymbol{\sigma}$. The structural reduction applies for arbitrary $M \geq 1$, while the spurious-attractor analysis focuses on $M \geq 3$, the first regime in which nontrivial mixture states arise. We now distinguish the stored memories from the remaining stable states. For a fixed reference sequence $\boldsymbol{\zeta}$, each stored memory $\boldsymbol{\xi}^m$ is represented by a block-level configuration $\boldsymbol{\sigma}^{(m)} \in \{-1,+1\}^P$, $m = 1, \ldots, M$, defined by $\xi_i^m = \zeta_i\sigma_p^{(m)}$ for $i \in A_p$, which is well defined because ξ_i^m/ζ_i is constant within a block. Since the weight matrix is invariant under $\boldsymbol{x} \to -\boldsymbol{x}$, the negation $-\boldsymbol{\xi}^m$ is a stored attractor whenever $\boldsymbol{\xi}^m$ is; at the block level this corresponds to $-\boldsymbol{\sigma}^{(m)}$, which satisfies Eq. 8 exactly when $\boldsymbol{\sigma}^{(m)}$ does, because replacing $\boldsymbol{\sigma}$ by $-\boldsymbol{\sigma}$ leaves $\boldsymbol{\Gamma}^{\boldsymbol{\sigma}}$ unchanged. Let $\mathcal{S}_{\mathrm{mem}} = \{\pm\boldsymbol{\sigma}^{(m)}\}_{m=1}^M$ denote the collection of these configurations together with their negations. The set of spurious stable configurations is then $\mathcal{S}_{\mathrm{sp}} \coloneqq \mathcal{S} \setminus \mathcal{S}_{\mathrm{mem}}$, with cardinality $N_{\mathrm{sp}} = |\mathcal{S}_{\mathrm{sp}}|$. A stable state is a spurious attractor precisely when its corresponding block configuration belongs to $\mathcal{S}_{\mathrm{sp}}$; by Theorems 4.2 and 4.4, the correspondence $x_i = \zeta_i\sigma_p$ is **a bijection from the stable states $\mathcal{A}$ onto $\mathcal{S}$**, under which the neuron-level spurious set $\mathcal{A}_{\mathrm{sp}}$ of Section 3 is mapped onto $\mathcal{S}_{\mathrm{sp}}$.

$$\mathcal{A} \;\cong\; \{\boldsymbol{\sigma} \in \{-1,+1\}^P : \boldsymbol{\Gamma}^{\boldsymbol{\sigma}}\boldsymbol{L} \succeq M\mathbf{1}_P\}, \qquad \mathcal{A}_{\mathrm{sp}} \;\cong\; \mathcal{S} \setminus \mathcal{S}_{\mathrm{mem}}. \tag{9}$$

The spurious attractors are therefore *exactly* the stable block configurations belonging to $\mathcal{S}_{\mathrm{sp}}$. In particular, $|\mathcal{A}_{\mathrm{sp}}| = N_{\mathrm{sp}}$, so a spurious attractor and its negation arise as the two orientations of a single block configuration, and the enumeration of that set yields the complete list of spurious attractors.

For the reference sequence $\boldsymbol{\zeta}$ itself, $\zeta_i x_i = +1$ for all i, and hence its block configuration is $\boldsymbol{\sigma} = \mathbf{1}_P$. Relative to the reference sequence, any additional stable configuration is obtained by flipping a set $\mathcal{F} \subseteq \{1, \ldots, P\}$ of blocks, that is, $\sigma_q = -1$ for $q \in \mathcal{F}$ and $\sigma_q = +1$ otherwise, equivalently $\boldsymbol{\sigma} = \mathbf{1}_P - 2\boldsymbol{f}$ with $\boldsymbol{f} \in \{0,1\}^P$ the indicator vector of $\mathcal{F}$. Under the criterion of Eq. 8, an unflipped block $p \notin \mathcal{F}$ requires $\sum_q \Gamma_{pq}L_q - 2\sum_{q\in\mathcal{F}}\Gamma_{pq}L_q \geq M$, whereas a flipped block $p \in \mathcal{F}$ requires $2\sum_{q\in\mathcal{F}}\Gamma_{pq}L_q - \sum_q \Gamma_{pq}L_q \geq M$. The stability of a flipped block is therefore determined by the collective contribution of all flipped blocks, and multiple flipped blocks may jointly reinforce or suppress one another through the entries of $\boldsymbol{\Gamma}$. **Hence, spurious attractor formation is not determined solely by the number of flipped neurons, but by the block structure and the signed interaction pattern induced by $\boldsymbol{\Gamma}$.**

Theorem 4.5 (Upper Bound for Three Stored Memories)**.** *For three stored memories, i.e., $M = 3$, the set of spurious attractors is contained in the set of signed superpositions, $\mathcal{S}_{\mathrm{sp}} \subseteq \{\mathrm{sign}(\pm\boldsymbol{\xi}^1 \pm \boldsymbol{\xi}^2 \pm \boldsymbol{\xi}^3)\}$, so that the number of spurious attractors is bounded by $N_{\mathrm{sp}}^{(M=3)} \leq 8$.*

The stability condition for this class, and the inclusion itself, are proved in Appendix B; the tightness of the bound is verified numerically in Appendix C. This is the classical symmetric-mixture class (Amit & Amit, 1989; Bruck & Roychowdhury, 1990), recovered here without ensemble assumptions; the $M = 3$ case therefore validates the block-level criterion rather than establishing a new bound.

Theorem 4.5 reveals that the existence of spurious attractors is governed by the interaction between three structural components: the block sizes $\boldsymbol{L}$, the centroid-induced interaction matrix $\boldsymbol{\Gamma}$, and the signed block configuration $\boldsymbol{\sigma}$.

4.5 Block-Level Dynamics

The results so far characterize the stable configurations themselves. We now show that the same block-level constants govern the dynamics by which they are reached, and that both update schedules share the same fixed-point set. By Eq. 7, the local field of a neuron depends on its state only through the pair (y_i, K_p), where $y_i \coloneqq \zeta_i x_i \in \{-1, +1\}$ is the alignment of neuron i with the reference sequence $\boldsymbol{\zeta}$, uniform within a block at a stable state (Theorem 4.2).

Consider an asynchronous schedule in which one neuron is updated per step, and write $y_i^{(t)}$ and $K_p^{(t)}$ for the alignment and the block-level constant of Eq. 6 at step t. At an update of neuron $i \in A_p$ all other neurons retain their values, and the field of Eq. 7 reads $H_i^{(t)} = (\zeta_i/N)(K_p^{(t)} - My_i^{(t)})$. Since $K_p^{(t)} - My_i^{(t)}$ involves $K_p^{(t)}$ only through its position relative to $\pm M$, and retaining the current state when $H_i^{(t)} = 0$, the update reduces to the three-way classification

$$y_i^{(t+1)} = \begin{cases} +1, & K_p^{(t)} \geq M, \\ -y_i^{(t)}, & |K_p^{(t)}| < M, \\ -1, & K_p^{(t)} \leq -M, \end{cases} \qquad i \in A_p. \tag{10}$$

A neuron of block A_p is thus driven to $+1$ whenever $K_p^{(t)} \geq M$ and to -1 whenever $K_p^{(t)} \leq -M$, whereas for $|K_p^{(t)}| < M$ it flips at every update, so that a configuration containing a block with $|K_p| < M$ cannot be a fixed point. Because any two neurons of a block that share the same alignment update identically, the block, rather than the neuron, is the natural unit of the asynchronous dynamics. The rule that a neuron is left unchanged when its local field is zero (Section 3) is already contained in Eq. 10: at the boundary values $K_p^{(t)} = \pm M$, a neuron with $y_i^{(t)} = \operatorname{sign}(K_p^{(t)})$ is left unchanged.

Proposition 4.6 (Block-level fixed points)**.** *For $M \geq 3$ stored memories, a configuration $\boldsymbol{x}$ is a fixed point of the asynchronous dynamics, that is, no single-neuron update changes it, if and only if its block configuration satisfies Eq. 8: for every p, the alignment is uniform within the block, $y_i = \sigma_p$ for all $i \in A_p$, and $\sigma_p K_p \geq M$.*

See Appendix E.2 for the detailed proof. Finally, the asynchronous dynamics always terminates at one of these fixed points. For a symmetric weight matrix with zero diagonal, every update that changes the state strictly decreases the energy $E(\boldsymbol{x}) = -\frac{1}{2}\sum_{i \neq j} w_{ij} x_i x_j$, since $\Delta E = 2x_i H_i = \frac{2}{N}(y_i K_p - M) < 0$ by the condition $y_i K_p < M$ that triggers the update. As the state space is finite and E is bounded below, any fair schedule, in which every unstable neuron is eventually selected, terminates at a configuration satisfying $y_i K_p \geq M$ for all neurons, that is, at a stable block configuration of Eq. 8. During transients the alignment need not be uniform within a block; Proposition 4.6 shows that uniformity is restored as soon as the dynamics settles at a fixed point.

The synchronous map T of Section 3 obeys the same local-field rule; what differs is the moment at which the shared block field is applied. Since all neurons of a block are updated with the same $K_p^{(t)}$ at the same time, **a single step moves whole blocks rather than single neurons**. Let $d_p^{(t)}$ be the block-wise Hamming distance of block A_p at step t. Since every neuron of A_p is updated with the same $K_p^{(t)}$, within a block that is already uniformly aligned the per-neuron classification of Eq. 10 is uniform across the block, and the block-level step rule becomes

$$d_p^{(t+1)} = \begin{cases} 0, & K_p^{(t)} \geq M, \\ L_p - d_p^{(t)}, & |K_p^{(t)}| < M, \\ L_p, & K_p^{(t)} \leq -M, \end{cases} \tag{11}$$

where $K_p^{(t)}$ is the block-level constant of Eq. 6 at $\boldsymbol{x}^{(t)}$. Within a uniformly aligned block the two outer regimes drive it to a uniform state within one step, whereas in the middle regime every neuron of the block flips and the block is complemented; a non-uniformly aligned block may instead split, since neurons of one block whose states differ receive the same K_p but different $-\zeta_i x_i$ terms. A uniform block is always mapped onto a uniform state again, so **block-uniformity, once attained, is never lost under synchronous updating; in the asynchronous case a single update may instead break the block-uniform structure of the block containing the updated neuron**. Because Eq. 11 expresses the next state through the current $\boldsymbol{d}^{(t)}$ alone, on the block-uniform states the synchronous dynamics is an autonomous map on the block-distance lattice, which those states occupy at its vertices, one per $\boldsymbol{\sigma} \in \{-1, +1\}^P$; the asynchronous dynamics traverses the lattice one coordinate at a time, whereas one synchronous step changes d_p by up to L_p. On the remaining states of the lattice the next distances are not determined by $\boldsymbol{d}^{(t)}$, since the per-neuron fields differ within a block; the trajectories of the synchronous map are therefore described by the 2^P block configurations rather than by the 2^N neuron-level states.

Remark 4.7 (Role of block size). Since $\Gamma_{pp} = M$, the criterion $\sigma_p K_p \geq M$ can be rewritten as

$$M\,(L_p - 1) \;\geq\; -\,\sigma_p \sum_{q \neq p} \Gamma_{pq} \sigma_q L_q, \tag{12}$$

so that the self-support of a block grows linearly with its length: a singleton block ($L_p = 1$) is stable only if its signed external support is non-negative, whereas a block of length L_p absorbs external opposition up to $M(L_p - 1)$. Since the collective field weights each block by its length (Section 4.4), stability is decided by the size-weighted pattern of signed inter-block interactions.

Finally, the termination guarantee of the asynchronous dynamics does not carry over to synchronous updating: for $M = 3$ with the memories $\{(+1, -1, -1), (+1, -1, +1), (+1, +1, -1)\}$, we have $\zeta = (+1, -1, -1)$ and three singleton blocks with $\Gamma_{12} = \Gamma_{13} = 1$ and $\Gamma_{23} = -1$. For the block configuration $\boldsymbol{\sigma} = (+1, -1, -1)$ and its negation, $\sigma_p K_p = 1 < 3 = M$, so that Eq. 11 complements all three blocks at every step and the state alternates between $(+1, +1, +1)$ and $(-1, -1, -1)$; neither is a fixed point, whereas the asynchronous dynamics from the same state reaches the stored memory $(-1, +1, +1) = -\boldsymbol{\xi}^1$ within one update. **The fixed-point characterization of Eq. 8 is therefore common to both schedules, but trajectory-level properties are not.**

Appendix H carries the criterion to two networks with $N = 11$, one at $M = 3$ and one at $M = 4$, and removes a single certified spurious attractor from each by the rank-one correction of Eq. H.1, whose strength ε_c is computable in $O(P)$ from $\boldsymbol{\Gamma}$ and $\boldsymbol{L}$. Fig. 3 in that appendix shows the resulting energy landscapes, before and after the removal, with every basin counted exactly over all 2^{11} initial states.

5 Conclusion, Limitation, and Future Work

Conclusion. This work develops a block-level framework for analyzing stable states and spurious attractors in binary HNNs. Partitioning neurons by their memory-alignment patterns reduces the neuron-level interaction structure to the block sizes $\boldsymbol{L}$ and the interaction matrix $\boldsymbol{\Gamma}$; **the Block-Uniform Flip Property makes stable states uniform within each block**, so that Eq. 8 describes *all* block-flip configurations through a binary vector $\boldsymbol{\sigma}$ over blocks. Non-memory configurations are thereby determined by the blocks and their mutual interactions.

Limitation and Future Work. The analysis assumes binary HNNs with symmetric weights, so it does not transfer to continuous-valued states, other activation functions, asymmetric interactions, or alternative learning rules; the reduction also depends on the stored memories, as many distinct alignment patterns drive P toward N, whereas structured memory sets keep it small. Simulation is removed, but not the worst-case exponential enumeration over P. Within this setting, Eq. 8 characterizes the stable configurations and the terminal states of the asynchronous dynamics (Proposition 4.6), but not their basins of attraction: a principal direction is the block-macrostate network supported by the block representation, on which basins can be computed exactly and spurious attractors eliminated in a principled way. Appendix H carries out the single-attractor case, whereas removing several at once, whose corrections interact through Eq. H.2, remains open, as do the scaling of spurious attractors and their behavior under correlated memories.

AI USE STATEMENT

In this work, we used generative AI tools for four purposes: language polishing of the manuscript, namely for correcting grammar and improving the wording and readability of text drafted by the authors; generating components of the schematic diagram in Fig. 1; developing the Python code that renders Fig. 3; and developing the verification code used in Appendix H. The theoretical derivations, the experimental design, the simulation, enumeration, and verification code of the $M = 3$ experiments in Appendix C, and the analysis of the results were produced by the authors; we have not used generative AI tools for research ideation. All AI-assisted text, code, and figure components were reviewed by the authors: the polished text preserves the technical content, claims, and notation of the original draft, and each script was run by the authors and its output checked against the underlying data. We take responsibility for the final content of this work, including text, claims, and artifacts.

REPRODUCIBILITY STATEMENT

The code and supplementary material for this work are available in an anonymous repository at `https://anonymous.4open.science/r/anonymous165dfghoad65465`.

ACKNOWLEDGMENTS

Use unnumbered third level headings for the acknowledgments. All acknowledgments, including those to funding agencies, go at the end of the paper.

REFERENCES

Shigeo Abe. Global convergence and suppression of spurious states of the hopfield neural networks. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 40(4): 246–257, 1993.

Yaser Abu-Mostafa and Jeannine St-Jacques. Information capacity of the Hopfield model. *IEEE Transactions on Information Theory*, 31(4):461–464, 1985.

Elena Agliari and Giordano De Marzo. Tolerance versus synaptic noise in dense associative memories. *The European Physical Journal Plus*, 135(11):1–22, 2020.

Elena Agliari, Adriano Barra, Andrea De Antoni, and Andrea Galluzzi. Parallel retrieval of correlated patterns: From hopfield networks to boltzmann machines. *Neural Networks*, 38:52–63, 2013.

Elena Agliari, Adriano Barra, and Matteo Notarnicola. The relativistic hopfield network: Rigorous results. *Journal of Mathematical Physics*, 60(3):033302, 2019.

Daniel J Amit and Daniel J Amit. *Modeling brain function: The world of attractor neural networks*. Cambridge university press, 1989.

Daniel J Amit, Hanoch Gutfreund, and Haim Sompolinsky. Spin-glass models of neural networks. *Physical Review A*, 32(2):1007, 1985.

Daniel J Amit, Hanoch Gutfreund, and Haim Sompolinsky. Statistical mechanics of neural networks near saturation. *Annals of physics*, 173(1):30–67, 1987.

Pierre Baldi and Santosh S Venkatesh. Number of stable points for spin-glasses and neural networks of higher orders. *Physical Review Letters*, 58(9):913, 1987.

Han Bao and Zhongying Zhao. Binary associative memory networks: A review of mathematical framework and capacity analysis. *Information Sciences*, 694:121697, 2025.

Han Bao, Richong Zhang, and Yongyi Mao. The capacity of the dense associative memory networks. *Neurocomputing*, 469:198–208, 2022.

Adriano Barra, Alberto Bernacchia, Enrica Santucci, and Pierluigi Contucci. On the equivalence of Hopfield networks and Boltzmann machines. *Neural Networks*, 34:1–9, 2012.

Agnish Kumar Behera, Matthew Du, Uday K Jagadisan, Srikanth Sastry, Madan Rao, and Suriyanarayanan Vaikuntanathan. Connection between hebbian unlearning and steady states generated by nonequilibrium dynamics. *PRX Life*, 3(4):043010, 2025.

Sofiya Bogakovskaya, Vladimir Fanaskov, and Ivan Oseledets. Dynamics of modern hopfield networks. In *New Frontiers in Associative Memories-Workshop at ICLR 2026*, 2026.

Jehoshua Bruck and Vwani P Roychowdhury. On the number of spurious memories in the hopfield model (neural network). *IEEE Transactions on Information Theory*, 36(2):393–397, 1990.

Vivien Cabannes, Berfin Simsek, and Alberto Bietti. Learning associative memories with gradient descent. *arXiv preprint arXiv:2402.18724*, 2024.

Francis Crick and Graeme Mitchison. The function of dream sleep. *Nature*, 304(5922):111–114, 1983.

Alberto Fachechi, Elena Agliari, and Adriano Barra. Dreaming neural networks: forgetting spurious memories and reinforcing pure ones. *Neural Networks*, 112:24–40, 2019.

Viola Folli, Giorgio Gosti, Marco Leonetti, and Giancarlo Ruocco. Effect of dilution in asymmetric recurrent neural networks. *Neural Networks*, 104:50–59, 2018.

Elizabeth Gardner and Bernard Derrida. Optimal storage properties of neural network models. *Journal of Physics A: Mathematical and general*, 21(1):271, 1988.

Véronique Gayrard. Mixed memories in hopfield networks. *arXiv preprint arXiv:2504.04879*, 2025.

Chris Gorman, Anthony Robins, and Alistair Knott. Hopfield networks as a model of prototype-based category learning: A method to distinguish trained, spurious, and prototypical attractors. *Neural Networks*, 91:76–84, 2017.

John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554--2558, 1982.

John J Hopfield. Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the national academy of sciences*, 81(10):3088–3092, 1984.

John J Hopfield, David I Feinstein, and Richard G Palmer. 'unlearning'has a stabilizing effect in collective memories. *Nature*, 304(5922):158–159, 1983.

Jerry Yao-Chieh Hu, Donglin Yang, Dennis Wu, Chenwei Xu, Bo-Yu Chen, and Han Liu. On sparse modern hopfield model. *Advances in Neural Information Processing Systems*, 36:27594–27608, 2023.

Jerry Yao-Chieh Hu, Thomas Lin, Zhao Song, and Han Liu. On computational limits of modern hopfield models: A fine-grained complexity analysis. *arXiv preprint arXiv:2402.04520*, 2024a.

Jerry Yao-Chieh Hu, Dennis Wu, and Han Liu. Provably optimal memory capacity for modern hopfield models: Transformer-compatible dense associative memories as spherical codes. *Advances in Neural Information Processing Systems*, 37:70693–70729, 2024b.

Sungmin Hwang, Viola Folli, Enrico Lanza, Giorgio Parisi, Giancarlo Ruocco, and Francesco Zamponi. On the number of limit cycles in asymmetric neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(5):053402, 2019.

Ido Kanter and Haim Sompolinsky. Associative recall of memory without errors. *Physical Review A*, 35(1):380, 1987.

James D Keeler. Basins of attraction of neural network models. In *AIP Conference Proceedings*, volume 151, pp. 259–264. American Institute of Physics, 1986.

János Komlós and Ramamohan Paturi. Convergence results in an associative memory model. *Neural Networks*, 1(3):239–250, 1988.

Dmitry Krotov and John J Hopfield. Dense associative memory for pattern recognition. In *Advances in Neural Information Processing Systems*, pp. 1172–1180, 2016.

Dmitry Krotov and John J Hopfield. Large associative memory problem in neurobiology and machine learning. In *International Conference on Learning Representations*, 2021.

Satish Kumar, Sanjay Saini, and Prem Prakash. Alien attractors and memory annihilation of structured sets in hopfield networks. *IEEE transactions on neural networks*, 7(5):1305–1309, 1996.

Xiaoyu Li, Yuanpeng Li, Yingyu Liang, Zhenmei Shi, and Zhao Song. On the expressive power of modern hopfield networks. *arXiv preprint arXiv:2412.05562*, 2024.

Yuchen Liang, Chaitanya K Ryali, Benjamin Hoover, Leopold Grinberg, Saket Navlakha, Dmitry Krotov, and Mohammed J Zaki. Can a fruit fly learn word embeddings? In *International Conference on Learning Representations*, 2021.

Tsubasa Masumura and Masato Taki. On the role of hidden states of modern hopfield network in transformer. *Advances in Neural Information Processing Systems*, 38:68607–68626, 2026.

Hayden McAlister, Anthony Robins, and Lech Szymanski. Classifying states of the hopfield network with improved accuracy, generalization, and interpretability. *arXiv preprint arXiv:2503.03018*, 2025.

Robertj McEliece, Edwardc Posner, Eugener Rodemich, and Santoshs Venkatesh. The capacity of the Hopfield associative memory. *IEEE Transactions on Information Theory*, 33(4):461–482, 1987.

Beren Millidge, Tommaso Salvatori, Yuhang Song, Thomas Lukasiewicz, and Rafal Bogacz. Universal Hopfield networks: A general framework for single-shot associative memory models. In *International Conference on Machine Learning*, pp. 15561–15583. PMLR, 2022.

Kazushi Mimura, Junichi Takeuchi, Yuto Sumikawa, Yoshiyuki Kabashima, and Anthony CC Coolen. Dynamical properties of dense associative memory. In *International Conference on Learning Representations*, volume 2026, pp. 19723–19744, 2026.

Chang Niu, Huanyu Zhang, Chuanlong Xu, Wenjie Hu, Yunzhuo Wu, Yu Wu, Yadi Wang, Tong Wu, Yi Zhu, Yinyan Zhu, et al. A self-learning magnetic hopfield neural network with intrinsic gradient descent adaption. *Proceedings of the National Academy of Sciences*, 121(51):e2416294121, 2024.

Toshihiro Ota and Ryo Karakida. Attention in a family of boltzmann machines emerging from modern hopfield networks. *Neural Computation*, 35(8):1463–1480, 2023.

Pierre Peretto. Collective properties of neural networks: a statistical physics approach. *Biological cybernetics*, 50(1):51–62, 1984.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, et al. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021.

Anthony Robins and Simon McCallum. Catastrophic forgetting and the pseudorehearsal solution in hopfield-type networks. *Connection Science*, 10(2):121–135, 1998.

Anthony V Robins and Simon JR McCallum. A robust method for distinguishing between learned and spurious attractors. *Neural Networks*, 17(3):313–326, 2004.

Edmund T Rolls. The memory systems of the human brain and generative artificial intelligence. *Heliyon*, 10(11), 2024.

Saul Santos, António Farinhas, Daniel C McNamee, and André FT Martins. Modern hopfield networks with continuous-time memories. *arXiv preprint arXiv:2502.10122*, 2025.

Bernhard Schäfl, Lukas Gruber, Angela Bitto-Nemling, and Sepp Hochreiter. Hopular: Modern Hopfield networks for tabular data. In *NeurIPS 2023 Second Table Representation Learning Workshop*, 2023.

Amos Storkey. Increasing the capacity of a hopfield network without sacrificing functionality. In *Artificial Neural Networks—ICANN'97: 7th International Conference Lausanne, Switzerland, October 8–10, 1997 Proceeedings 7*, pp. 451–456. Springer, 1997.

Amos J Storkey and Romain Valabregue. The basins of attraction of a new hopfield learning rule. *Neural Networks*, 12(6):869--876, 1999.

Danil Tyulmankov, Ching Fang, Annapurna Vadaparty, and Guangyu Robert Yang. Biological learning in key-value memory networks. In *Advances in Neural Information Processing Systems*, volume 34, pp. 22247–22258, 2021.

JL Van Hemmen. Hebbian learning, its correlation catastrophe, and unlearning. *Network: Computation in Neural Systems*, 8(3):V1, 1997.

Shurong Wang, Yuqi Pan, Zhuoyang Shen, Meng Zhang, Hongwei Wang, and Guoqi Li. Adaptive hopfield network: Rethinking similarities in associative memory. In *International Conference on Learning Representations*, volume 2026, pp. 116942–116974, 2026a.

Tianwei Wang, Xinhui Ma, and Wei Pang. Asymptotically stable quaternion-valued hopfield-structured neural network with periodic projection-based supervised learning rules. *Advances in Neural Information Processing Systems*, 38:87040–87063, 2026b.

Runze Xia and Richard Jiang. Energy landscape-aware vision transformers: Layerwise dynamics and adaptive task-specific training via hopfield states. *Advances in Neural Information Processing Systems*, 38:12509–12532, 2026.

Chenwei Xu, Yu Chao Huang, Jerry Yao Chieh Hu, Weijian Li, Ammar Gilani, Hsi Sheng Goan, and Han Liu. Bishop: Bi-directional cellular learning for tabular data with generalized sparse modern hopfield model. *Proceedings of Machine Learning Research*, 235:55048–55075, 2024.

Appendix

This appendix carries the proofs, the notation glossary and the numerical validation of the main text. The material is organized as follows.

Appendix A is the notation glossary. Appendix B characterizes the $M = 3$ case of Theorem 4.5 and proves the completeness of the signed-superposition class; Appendix C tests that characterization exhaustively over the full state space for $N \leq 7$ and by sampled dynamics at $N = 100, 200, 500$. Appendix D records the intermediate steps of the block-level rewrite of the weights and the local field used in Section 4.1; Appendix D.1 proves the Block-Uniform Flip Property of Theorem 4.2, and Appendix E the block-level stability criterion of Theorem 4.4 and the fixed-point Proposition 4.6.

Appendix F.1 collects the structure, the spectrum and the gauge invariance of $\mathbf{\Gamma}$, together with the block counting (Appendix F.4) and the formal definition of the bit-level signature. Appendix G works through a closed-form $M = 4$ example, Appendix H applies the criterion to the exact deletion of a single certified spurious attractor, Appendix I gives the complexity of the resulting algorithm.

Appendix Contents

A Notation Glossary **16**

A.1 Network Basic Parameters 16

A.2 Block Structure & Signature 16

A.3 $\mathbf{\Gamma}$-Matrix & Stability Criteria 16

A.4 Distance & Attractor Definitions 17

A.5 Computational Complexity 17

B Derivation of the $M = 3$ Characterization **18**

B.1 Completeness for $M = 3$: Proof of Theorem 4.5 19

C Experiments in $M = 3$ **21**

C.1 Experimental Setup 21

C.2 Traversal Experiments: Verification of the Predicted Class with $M = 3$ 21

C.3 Exhaustive Verification over the Full State Space 22

C.4 Sampling Experiments: Verification of Spurious Attractor Completeness with $M = 3$ 22

C.5 Scope of Empirical Validation 23

D Details for Eq. 4 **24**

D.1 Proof of Theorem 4.2 24

E Proofs of the Stability Criterion and the Fixed-Point Proposition **25**

E.1 Proof of Theorem 4.4 25

E.2 Proof of Proposition 4.6 25

F Formal Definitions for the Block Partition of Section 4.1 **26**

F.1 Structure of the Interaction Matrix 26

F.2 Spectrum and Gram Structure of $\mathbf{\Gamma}$ 26

F.3 Gauge Invariance of the Reference Centroid 27

F.4 Block Counting 28
F.5 Worked Instance of the Block Decomposition 29
F.6 Definition of Bit-level Signature 29

G Complete Example for $M = 4$ **30**
G.1 Minimal Experimental Setup 30
G.2 Block Partition via Bit-level Signature 30
G.3 $\mathbf{\Gamma}$-matrix Construction 30

H Exact Elimination of a Spurious Attractor **32**
H.1 Removing One Spurious Attractor 32
H.2 Case $M = 3$: $N = 11$, $P = 4$ 33
H.3 Case $M = 4$: $N = 11$, $P = 7$ 34
H.4 Relation to Existing Unlearning Schemes 35

I Computational Complexity and Algorithm **36**
I.1 Computational Complexity Analysis 36
I.2 Algorithmic Characterization 36

A NOTATION GLOSSARY

This section defines all key variables, symbols, and mathematical objects used throughout the paper for clarity and reproducibility.

A.1 NETWORK BASIC PARAMETERS

- N: Total number of neurons in the binary HNN.
- M: Number of stored binary memories, satisfying $M \geq 3$.
- $\mathcal{X} = \{-1, +1\}^N$: State space of the HNN.
- $\boldsymbol{x} \in \mathcal{X}$: Network state, with x_i the state of neuron i.
- $\xi^m \in \mathcal{X}$: The m-th stored memory ($m = 1, 2, \ldots, M$).
- ξ_i^m: The i-th neuron state of the m-th stored memory.
- $W \in \mathbb{R}^{N \times N}$: Weight matrix of the HNN, trained by Hebbian rule.
- w_{ij}: The weight between neuron i and neuron j ($i \neq j$).
- H_i: The local field of neuron i at time t, that is, the weighted sum of the postsynaptic potentials of the neurons connected to i, $H_i(t) = \sum_{j=1}^{N} w_{ij} x_j^{(t)}$ (Section 3).
- $E(\boldsymbol{x})$: Energy function of the HNN, $E(\boldsymbol{x}) = -\frac{1}{2} \sum_{i \neq j} w_{ij} x_i x_j$.

A.2 BLOCK STRUCTURE & SIGNATURE

- P: Number of disjoint neuron blocks after block decomposition.
- A_p: The p-th neuron block ($p = 1, 2, \ldots, P$), that is, a maximal set of neurons whose memory-alignment patterns agree up to an overall sign (Definition 4.1).
- L_p: Size of block A_p, that is, the number of neurons it contains ($L_p = |A_p|$); $\boldsymbol{L} = [L_1, L_2, \ldots, L_P]^T$ is the block-size vector.
- $\mathcal{M}_i$: Bit-level signature of neuron i, the set of memories in which neuron i disagrees with the symbolic centroid, used for block partitioning (Definition F.13).
- $\mathcal{M}_p \coloneqq \{m : s_p^m = -1\}$: Minority index set of block A_p, that is, the memories in which its signature takes the value -1; its size $t_p = |\mathcal{M}_p|$ is the minority size of the block.
- s_p^m: Block-specific coefficient of the m-th memory for block A_p, defined by $\xi_i^m = s_p^m \zeta_i$ on A_p (Section 4.1).
- $\boldsymbol{\zeta} \in \mathcal{X}$: Symbolic centroid of the stored memories, the bit-wise majority vote of $\xi^1, \ldots, \xi^M$ with ties resolved by the first memory (Definition 3.4); it is the reference sequence relative to which blocks and signatures are defined.
- $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}}$: Centroid of the sign-combined memories $(\varepsilon_1 \xi^1, \ldots, \varepsilon_M \xi^M)$ for a sign combination $\boldsymbol{\varepsilon} \in \{\pm 1\}^M$ (Corollary F.11); for $M = 3$ these are the eight signed superpositions of Theorem 4.5. All 2^M centroids induce the same block partition and are related by the gauge transformation of Proposition F.10.
- $y_i \coloneqq \zeta_i x_i \in \{-1, +1\}$: Alignment of neuron i with the symbolic centroid; $y_i = \sigma_p$ for all $i \in A_p$ at a stable state (Theorem 4.2).
- $S \in \{\pm 1\}^{P \times M}$: Block-signature matrix, whose row p is the signature s_p of block A_p (Lemma F.5).

A.3 $\boldsymbol{\Gamma}$-MATRIX & STABILITY CRITERIA

- $\boldsymbol{\Gamma} \in \mathbb{R}^{P \times P}$: Block-interaction $\boldsymbol{\Gamma}$-matrix, symmetric structural descriptor.
- $\odot$: Hadamard product (element-wise multiplication).
- Γ_{pq}: Interaction coefficient between block A_p and block A_q.
- K_p: Block-level constant of block A_p, defined in Eq. 6 as $K_p = \sum_{q=1}^{P} \Gamma_{pq}(L_q - 2d_q)$.
- $\sigma_p \in \{+1, -1\}$: Block flip state of block A_p.

- $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_P]^T$: Block configuration, encoding a state through $x_i = \zeta_i \sigma_p$ for $i \in A_p$.
- $\mathcal{F} \subseteq \{1, \ldots, P\}$: Set of blocks flipped relative to the reference configuration $\mathbf{1}_P$, and $\boldsymbol{f} \in \{0,1\}^P$ its indicator vector, so that $\boldsymbol{\sigma} = \mathbf{1}_P - 2\boldsymbol{f}$ (Section 4.4).
- $\mathbf{\Gamma}^{\boldsymbol{\sigma}} = \mathrm{diag}(\boldsymbol{\sigma})\mathbf{\Gamma}\,\mathrm{diag}(\boldsymbol{\sigma})$: Signed interaction matrix entering the stability criterion of Eq. 8.
- $D_{\boldsymbol{\sigma}} \coloneqq \mathrm{diag}(\boldsymbol{\sigma})$: Diagonal sign matrix of a block configuration (Proposition F.10).
- $\mathbf{1}_P$: P-dimensional all-ones vector.
- $\succeq$: Component-wise partial order for vector inequalities.
- ε_c: Critical strength of the rank-one correction that removes a certified spurious attractor (Appendix H).

A.4 Distance & Attractor Definitions

- $d(\boldsymbol{u}, \boldsymbol{v})$: Standard Hamming distance between two binary vectors $\boldsymbol{u}$ and $\boldsymbol{v}$.
- $d(\boldsymbol{x}, \boldsymbol{\zeta})$: Standard Hamming distance between the state $\boldsymbol{x}$ and the symbolic centroid $\boldsymbol{\zeta}$.
- d_p: Block-wise Hamming distance on block A_p, measuring the local Hamming discrepancy between $\boldsymbol{x}$ and $\boldsymbol{\zeta}$ (Definition 3.6).
- $d_H(\boldsymbol{u}, \boldsymbol{v})$: Hamming distance on block signatures, $\boldsymbol{u}, \boldsymbol{v} \in \{\pm 1\}^M$ (Lemma F.1).
- $\mathcal{A}$: Set of all attractors, that is, all stable states (Definition 3.1); calligraphic letters denote sets throughout, whereas the italic A_p denotes the p-th block of neuron indices.
- $\cong$: One-to-one correspondence (bijection) between two sets; no further structure is implied. Eq. 9 records that the map $x_i = \zeta_i \sigma_p$ ($i \in A_p$) is a bijection from the block configurations satisfying Eq. 8 onto $\mathcal{A}$, and that its restriction maps $\mathcal{S} \setminus \mathcal{S}_{\mathrm{mem}}$ bijectively onto $\mathcal{A}_{\mathrm{sp}}$.
- $\mathcal{A}_{\mathrm{mem}}$: Set of stored attractors at the neuron level, that is, the stored memories and their negations (Section 3).
- $\mathcal{A}_{\mathrm{sp}}$: Set of spurious attractors at the neuron level (stable states not belonging to $\mathcal{A}_{\mathrm{mem}}$).
- $\mathcal{S}$: Set of stable block configurations, as defined in Section 4.3.
- $\boldsymbol{\sigma}^{(m)}$: Block configuration corresponding to the m-th stored memory, $m = 1, \ldots, M$ (Section 4.4).
- $\mathcal{S}_{\mathrm{mem}} = \{\pm \boldsymbol{\sigma}^{(m)}\}_{m=1}^{M}$: Set of block configurations corresponding to the stored memories and their negations, as defined in Section 4.4.
- $\mathcal{S}_{\mathrm{sp}}$: Set of spurious stable block configurations (stable configurations not belonging to $\mathcal{S}_{\mathrm{mem}}$), as defined in Eq. 9.
- $N_{\mathrm{sp}} = |\mathcal{S}_{\mathrm{sp}}| = |\mathcal{A}_{\mathrm{sp}}|$: Number of spurious attractors (Section 4.4).
- $T(\cdot)$: Synchronous state update mapping of the HNN.

A.5 Computational Complexity

- $\mathcal{O}(\cdot)$: Big-O notation for asymptotic time complexity.

B Derivation of the $M = 3$ Characterization

This section derives the $M = 3$ statement of Theorem 4.5 from the block-level criterion of Eq. 8. Throughout, the block partition is computed with respect to the symbolic centroid $\boldsymbol{\zeta} = \mathrm{sign}(\xi^1 + \xi^2 + \xi^3)$ of the stored triple. The eight signed combinations do not require a separate partition: by Corollary F.11 the centroid $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}}$ of $(\varepsilon_1\xi^1, \varepsilon_2\xi^2, \varepsilon_3\xi^3)$ induces the same blocks as $\boldsymbol{\zeta}$ and differs from it only by a block-uniform sign factor, so their stability questions are evaluations of the single criterion of Eq. 8 rather than eight independent problems.

For $M = 3$ and any sign combination, a neuron either agrees with the corresponding centroid in all three signed memories, in which case it belongs to the all-majority block A_{123}, or it disagrees in exactly one of them, in which case it belongs to one of the three 2–1 blocks $A_{12|3}$, $A_{13|2}$, $A_{23|1}$, where the notation lists the two memories that agree with the majority before the bar; we abbreviate these blocks by 12, 13, and 23. We write $n_{123} = |A_{123}|$ and $L_p = |A_p|$ for a 2–1 block. The interaction coefficients follow from Eq. 4: $\Gamma_{pp} = 3$ for every block, $\Gamma_{pq} = 1$ between A_{123} and any 2–1 block, and $\Gamma_{pq} = -1$ between two distinct 2–1 blocks. These coefficients are the same for every combination: by Eq. F.10 passing to another combination negates the entries s_p^m jointly with the reference, so that the products $s_p^m s_q^m$ and hence Γ_{pq} are unchanged.

Theorem B.1. *For a binary HNN with $M = 3$, let $\boldsymbol{\zeta} = \mathrm{sign}(\xi^1 + \xi^2 + \xi^3)$ be the symbolic centroid of the stored triple and, for a sign combination $\boldsymbol{\varepsilon} \in \{\pm 1\}^3$, let $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}} = \mathrm{sign}(\varepsilon_1\xi^1 + \varepsilon_2\xi^2 + \varepsilon_3\xi^3)$ be the centroid of the corresponding signed combination, the eight combinations $\pm\boldsymbol{\zeta}$, $\mathrm{sign}(\pm\xi^1 \pm \xi^2 \pm \xi^3)$ being those of Theorem 4.5. For each of them, the signed superposition is a fixed point of T if and only if*

$$2n_{123} + 4L_p \;\geq\; N + 3 \qquad \textit{for every nonempty 2–1 block } A_p, \tag{B.1}$$

where n_{123} and L_p are the block sizes of the stored triple. The all-majority block A_{123}, when nonempty, is always stable. Moreover, a signed superposition is a spurious attractor precisely when it is a fixed point and it does not coincide with a stored memory or its negation.

Proof. Since the tie-breaking rule of Eq. 3 is never invoked for $M = 3$, each combination sum $\varepsilon_1\xi_i^1 + \varepsilon_2\xi_i^2 + \varepsilon_3\xi_i^3$ is a sum of three odd terms and never vanishes, so $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}}$ is well defined and, by Corollary F.11, agrees with the reference centroid $\boldsymbol{\zeta}$ up to the block-uniform factor $\eta_p^{\boldsymbol{\varepsilon}}$. At the candidate state $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}}$ the block configuration is therefore $\boldsymbol{\sigma} = \mathbf{1}_P$, so that Eq. 7 gives the local field

$$H_i = \frac{\zeta_i^{\boldsymbol{\varepsilon}}}{N}\,(K_p - 3)\,, \qquad K_p = \sum_{q=1}^{P} \Gamma_{pq} L_q, \tag{B.2}$$

and, since $\sigma_p K_p$ of Eq. 8 is gauge invariant (Proposition F.10), this evaluation in the gauge of the candidate is the same condition as the one in the reference gauge. Hence $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}}$ is a fixed point if and only if $K_p \geq 3$ for every nonempty block A_p.

For the all-majority block, $K_{123} = 3n_{123} + \sum_{p \neq 123} L_p = 2n_{123} + N \geq 3$ whenever $n_{123} \geq 1$, so this block imposes no additional condition. For a nonempty 2–1 block A_p with $L_p = |A_p|$, the coefficient $\Gamma_{pq} = 1$ applies once (against A_{123}) and $\Gamma_{pq} = -1$ applies to the remaining $N - n_{123} - L_p$ neurons, hence

$$K_p = n_{123} + 3L_p - (N - n_{123} - L_p) = 2n_{123} + 4L_p - N. \tag{B.3}$$

Requiring $K_p \geq 3$ for every nonempty 2–1 block is therefore exactly Eq. B.1; since the right-hand side of Eq. B.3 increases with L_p, it is enough to check the smallest nonempty 2–1 block.

Finally, by the definition of the spurious set in Section 3, a signed superposition is a spurious attractor exactly when it is a fixed point and it does not coincide with a stored memory or its negation. □

Remark B.2. When all three 2–1 configurations occur, writing k for the smallest of their sizes and $k + t_1$, $k + t_2$ for the other two, we have $N = n_{123} + 3k + t_1 + t_2$, and Eq. B.1 reduces to the compact form $n_{123} + k - 3 \geq t_1 + t_2$. If one of the three configurations is absent, the corresponding class is empty and imposes no condition: applying the compact form with k taken over all three configuration sizes is then sufficient but not necessary, and Eq. B.1 must be checked on the nonempty blocks only.

Theorem B.1 characterizes the stability of each of the eight signed superpositions of Theorem 4.5. The converse inclusion, that no spurious attractor lies outside this class, is proved in Section B.1 and verified exhaustively over the full state space in Appendix C.3.

B.1 Completeness for $M = 3$: Proof of Theorem 4.5

Theorem B.1 identifies which of the eight signed superpositions are stable; we now prove the converse inclusion, so that no spurious attractor escapes the class. We take the reference sequence of Section 4.1 to be the symbolic centroid $\boldsymbol{\zeta} = \operatorname{sign}(\xi^1 + \xi^2 + \xi^3)$ of the stored triple, so that the partition consists of the all-majority block A_{123} and the three 2–1 blocks $A_{12|3}, A_{13|2}, A_{23|1}$ of Section B. For $M = 3$ the sum $\xi_i^1 + \xi_i^2 + \xi_i^3$ is a sum of three odd terms and never vanishes, so $\boldsymbol{\zeta}$ is defined without invoking the tie-breaking rule. Writing $m \coloneqq n_{123} = |A_{123}|$ and relabeling the memories so that $|A_{12|3}| \le |A_{13|2}| \le |A_{23|1}|$, we set

$$|A_{12|3}| = k, \qquad |A_{13|2}| = k + t_1, \qquad |A_{23|1}| = k + t_2, \qquad N = m + 3k + t_1 + t_2, \tag{B.4}$$

with $t_1, t_2 \ge 0$, and abbreviate $C \coloneqq m + k - t_1 - t_2$.

Let $\boldsymbol{x}$ be a stable state, and let $d_{123}, d_{12}, d_{13}, d_{23}$ be the block-wise Hamming distances between $\boldsymbol{x}$ and $\boldsymbol{\zeta}$ on the four blocks, with $d \coloneqq d_{123} + d_{12} + d_{13} + d_{23}$. Since $\Gamma_{pp} = 3$, $\Gamma_{pq} = 1$ between A_{123} and a 2–1 block, and $\Gamma_{pq} = -1$ between two distinct 2–1 blocks, the block-level constants of Eq. 6 evaluate to

$$\begin{aligned} K_{123} &= N + 2m - 2d - 4d_{123}, & K_{12} &= C + 2d - 4d_{123} - 8d_{12}, \\ K_{13} &= C + 4t_1 + 2d - 4d_{123} - 8d_{13}, & K_{23} &= C + 4t_2 + 2d - 4d_{123} - 8d_{23}, \end{aligned} \tag{B.5}$$

and Theorem 4.4 states that $\boldsymbol{x}$ is stable if and only if $\sigma_p K_p \ge 3$ for every nonempty block. By Theorem 4.2 each block of a stable state is flipped uniformly, so every block-wise Hamming distance attains one of its two extreme values, $(d_{123}, d_{12}, d_{13}, d_{23}) \in \{0, m\} \times \{0, k\} \times \{0, k+t_1\} \times \{0, k+t_2\}$. At most $2^4 = 16$ block configurations remain, and the eight with $\sigma_{123} = -1$ are the negations of the eight with $\sigma_{123} = +1$.

One of the four blocks may be empty, in which case complementing it changes no neuron and the corresponding coordinate of $\boldsymbol{\sigma}$ is immaterial. The enumeration below is then to be read with that coordinate dropped: the last configuration of Lemma B.3 coincides with the row $\{13, 23\}$ when $A_{12|3} = \emptyset$, with the row $\{12, 23\}$ when $A_{13|2} = \emptyset$, and with the row $\{12, 13\}$ when $A_{23|1} = \emptyset$, while an empty A_{123} leaves $-\zeta$, the negation of the first row. In each of these cases the state is again a signed superposition or a stored memory, so only the case in which all four blocks are nonempty has to be excluded separately.

Lemma B.3 (Block configurations with $\sigma_{123} = +1$)**.** *Let $\boldsymbol{\zeta} = \operatorname{sign}(\xi^1 + \xi^2 + \xi^3)$, let $A_{123}, A_{12|3}, A_{13|2}, A_{23|1}$ be the induced blocks, and let $\boldsymbol{\sigma}$ be a block configuration with $\sigma_{123} = +1$. Writing $F \subseteq \{12, 13, 23\}$ for the set of flipped 2–1 blocks, the corresponding state $x_i = \zeta_i \sigma_p$, $i \in A_p$, is given by*

$$\begin{aligned} F = \emptyset &: \ \boldsymbol{x} = \operatorname{sign}(\xi^1 + \xi^2 + \xi^3) = \boldsymbol{\zeta}, & F = \{13, 23\} &: \ \boldsymbol{x} = \operatorname{sign}(\xi^1 + \xi^2 - \xi^3), \\ F = \{23\} &: \ \boldsymbol{x} = \xi^1, & F = \{12, 23\} &: \ \boldsymbol{x} = \operatorname{sign}(\xi^1 - \xi^2 + \xi^3), \\ F = \{13\} &: \ \boldsymbol{x} = \xi^2, & F = \{12, 13\} &: \ \boldsymbol{x} = \operatorname{sign}(-\xi^1 + \xi^2 + \xi^3), \\ F = \{12\} &: \ \boldsymbol{x} = \xi^3, & F = \{12, 13, 23\} &: \ \boldsymbol{x} \text{ is not stable} \end{aligned} \tag{B.6}$$

the last row being the content of Lemma B.4 and applying when all four blocks are nonempty.

Proof. By definition of the blocks, $\xi^1 = \boldsymbol{\zeta}$ on $A_{123} \cup A_{12|3} \cup A_{13|2}$ and $\xi^1 = -\boldsymbol{\zeta}$ on $A_{23|1}$, so complementing $A_{23|1}$ turns $\boldsymbol{\zeta}$ into ξ^1; the rows for ξ^2 and ξ^3 follow in the same way, since ξ^2 differs from $\boldsymbol{\zeta}$ exactly on $A_{13|2}$ and ξ^3 exactly on $A_{12|3}$. For the mixed rows, evaluating the combination block by block gives

$$\xi_i^1 + \xi_i^2 - \xi_i^3 = \begin{cases} \zeta_i, & i \in A_{123}, \\ 3\zeta_i, & i \in A_{12|3}, \\ -\zeta_i, & i \in A_{13|2}, \\ -\zeta_i, & i \in A_{23|1}, \end{cases} \tag{B.7}$$

so that $\operatorname{sign}(\xi^1 + \xi^2 - \xi^3)$ agrees with $\boldsymbol{\zeta}$ on $A_{123} \cup A_{12|3}$ and with $-\boldsymbol{\zeta}$ on $A_{13|2} \cup A_{23|1}$, which is the row $F = \{13, 23\}$. The remaining two mixed rows are obtained by permuting the roles of the three memories. □

Lemma B.4 (No stable state complements A_{123} alone)**.** *Suppose the four blocks are nonempty. Then the configuration that complements A_{123} while leaving the three 2–1 blocks unchanged, and the configuration that complements all three 2–1 blocks while leaving A_{123} unchanged, are not stable states.*

Proof. The two configurations are negations of each other, and stability is preserved under $\boldsymbol{x} \to -\boldsymbol{x}$ because $H_i(-\boldsymbol{x}) = -H_i(\boldsymbol{x})$; hence it suffices to treat the first one, for which $d_{123} = m$, $d_{12} = d_{13} = d_{23} = 0$ and $d = m$. By Eq. B.5,

$$\begin{aligned} \sigma_{123}K_{123} &= -(N-4m), & \sigma_{12}K_{12} &= C-2m, \\ \sigma_{13}K_{13} &= C+4t_1-2m, & \sigma_{23}K_{23} &= C+4t_2-2m, \end{aligned} \tag{B.8}$$

so that stability would require

$$\begin{cases} 3a+t_1+t_2 \le -3, \\ a-t_1-t_2 \ge 3, \\ a+3t_1-t_2 \ge 3, \\ a-t_1+3t_2 \ge 3, \end{cases} \tag{B.9}$$

where $a \coloneqq k-m$ and $t_1, t_2 \ge 0$. The second inequality gives $a \ge 3+t_1+t_2$ and thereby implies the third and the fourth, whereas the first gives $3a \le -3$, that is, $a \le -1$. The two bounds are incompatible, so no such stable state exists. □

Proof of Theorem 4.5. Let $\boldsymbol{\sigma} \in \mathcal{S}_{\mathrm{sp}}$ and let $\boldsymbol{x}$ be the corresponding state, $x_i = \zeta_i \sigma_p$ for $i \in A_p$. Suppose first that $\sigma_{123} = +1$. By Lemma B.3, $\boldsymbol{x}$ is one of the states listed there. The three stored memories ξ^1, ξ^2, ξ^3 have block configurations $\boldsymbol{\sigma}^{(m)} \in \mathcal{S}_{\mathrm{mem}}$ and are excluded by definition of $\mathcal{S}_{\mathrm{sp}}$. The configuration of the last row is excluded by Lemma B.4 when all four blocks are nonempty, and otherwise coincides with one of the rows $\{13, 23\}$, $\{12, 23\}$, $\{12, 13\}$, or with $-\zeta$ when $A_{123} = \emptyset$; each of these is a signed superposition. The remaining four states are ζ, $\mathrm{sign}(\xi^1 + \xi^2 - \xi^3)$, $\mathrm{sign}(\xi^1 - \xi^2 + \xi^3)$ and $\mathrm{sign}(-\xi^1 + \xi^2 + \xi^3)$, each of which is a signed superposition of the stored memories. Suppose next that $\sigma_{123} = -1$. Then $-\boldsymbol{x}$ is stable as well, because reversing the state reverses every local field, and $-\boldsymbol{x}$ has $\sigma_{123} = +1$, so the previous paragraph applies to it: $-\boldsymbol{x}$ is either one of the stored memories, in which case $\boldsymbol{x}$ is a stored memory too and cannot belong to $\mathcal{S}_{\mathrm{sp}}$, or a signed superposition of the stored memories. In the latter case $\boldsymbol{x}$ is a signed superposition as well, since negating $\boldsymbol{x}$ negates all three coefficients. Hence

$$\mathcal{S}_{\mathrm{sp}} \subseteq \left\{ \mathrm{sign}\left(\pm\xi^1 \pm \xi^2 \pm \xi^3 \right) \right\}, \tag{B.10}$$

and as the class on the right contains at most one state for each of the eight sign patterns, $N_{\mathrm{sp}}^{(M=3)} \le 8$. □

C EXPERIMENTS IN $M = 3$

C.1 EXPERIMENTAL SETUP

The experiments in this section aim to empirically validate the effectiveness of Theorem 4.5 along two directions: (i) verifying that all spurious attractors arising in the $M = 3$ setting conform to the theoretical characterization, and (ii) confirming the state consistency between spurious attractors and their negations. All experiments were conducted on a local workstation equipped with an Intel Core Ultra 5 225H CPU (1.70 GHz), 32 GB DDR5 RAM, an Intel Arc 130T GPU, and a 954 GB SSD, running on 64-bit Windows 11 Professional.

To balance exhaustive verification at small scales with computational feasibility at large scales, we adopt two complementary experimental settings. For small networks ($N = 3$–8), exhaustive experiments enumerate all memory triples formed from the 2^N binary states, enabling complete coverage and direct verification of Theorem 4.5. For large networks ($N = 100, 200, 500$), where exhaustive traversal is infeasible due to the exponential growth of 2^N, we employ a targeted sampling strategy that draws 5% of the initial states from stored-memory patterns and their near neighbours, 40% from the boundary regions between stored memories, 30% at intermediate Hamming distances, and 25% from far or random control states. Together, these settings provide a unified framework for rigorous small-scale verification and efficient large-scale validation.

C.2 TRAVERSAL EXPERIMENTS: VERIFICATION OF THE PREDICTED CLASS WITH $M = 3$

All experiments are designed to directly verify Theorem 4.5. Specifically, for each group of memories, we first compute the symbolic centroids $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}}$ corresponding to all 8 possible sign combinations $\boldsymbol{\varepsilon} \in \{\pm 1\}^3$. Using the synchronous update rule described in Section 3, we then test whether each centroid is a fixed point by checking if it remains invariant after one update. Centroids belonging to $\mathcal{A}_{\text{mem}}$ are stored attractors, and the remaining fixed points are spurious according to the definition in Section 3; these are classified according to Theorem 4.5, whose predicted class consists of the signed superpositions of the three stored memories. The results are summarized in Table 1, which reports exhaustive traversal statistics for small-scale HNNs and provides direct empirical validation of the theorem. Because this procedure tests the eight candidates themselves, it verifies the stability condition for the predicted class; the complementary question, whether spurious attractors exist outside the class, is settled analytically in Section B.1 and tested empirically in Section C.3.

Table 1: Statistics of spurious attractors. This table reports counts of total spurious attractors (SA) and predicted-class spurious attractors (those predicted by Theorem 4.5), along with consistency metrics, for HNNs of different sizes N. Counts are reported per sign-equivalence class of memory triples: the eight triples obtained by $\xi^m \to \pm\xi^m$ generate the same weight matrix and are counted once. A key takeaway is that all identified spurious attractors belong to the predicted class, with 100% consistency across all tested configurations.

N	Total SA	Pred. SA	Consist.	Inv. Cons.
3	0	0	100%	100%
4	0	0	100%	100%
5	320	320	100%	100%
6	1280	1280	100%	100%
7	44800	44800	100%	100%
8	229376	229376	100%	100%

As shown in Table 1, no spurious attractors are observed for $N = 3$ and $N = 4$, which can be attributed to the strong update volatility in ultra-small networks that prevents the formation of spurious attractors. For $N = 5$ to 8, the total number of spurious attractors (**Total SA**) exactly matches the number predicted by Theorem 4.5 (**Pred. SA**), with both the consistency rate (**Consist.**) and the inverse consistency (**Inv. Cons.**) reaching 100% across all N. This exact correspondence shows that every signed superposition counted as spurious is a genuine spurious attractor of the network, and that the stored memories, the centroid, and their negations are precisely the members of the class that fail to be stable. The complementary inclusion—that no spurious attractor lies outside the predicted

class—is not tested by this procedure; it is proved in Section B.1 and verified next over the full state space.

C.3 Exhaustive Verification over the Full State Space

The procedure of Section C.2 tests the eight signed superpositions themselves and therefore verifies the stability condition of Theorem 4.5 for the predicted class; it does not exclude spurious attractors outside that class. Section B.1 proves that no such attractor exists; here we test the inclusion directly by enumerating the full state space. For $N = 3, \ldots, 7$ we consider every triple of distinct binary memories, iterate the synchronous map T of Section 3 from each of the 2^N initial states, collect all fixed points, and classify each of them as stored ($\boldsymbol{x}^* \in \mathcal{A}_{\mathrm{mem}}$) or spurious ($\boldsymbol{x}^* \in \mathcal{A}_{\mathrm{sp}}$); a zero local field leaves the neuron unchanged, as in Section 3. Every spurious attractor is then compared with the eight signed superpositions of the stored triple. All trajectories are obtained by direct iteration of T; a memory triple and its sign-equivalent triples share one weight matrix and are therefore simulated once.

Table 2: Exhaustive verification over the full state space. Counts are summed over all triples of distinct binary memories of size N, with every one of the 2^N states simulated to convergence for each triple. A key observation is that no spurious attractor outside the predicted class is found, and that the number of spurious attractors equals the number of signed superpositions that are actual fixed points.

N	Memory triples	Spurious SA (full)	In predicted class	Outside
3	56	0	0	0
4	560	0	0	0
5	4960	2560	2560	0
6	41664	10240	10240	0
7	341376	358400	358400	0

No spurious attractor outside the predicted class is observed at any tested size, so the inclusion of Theorem 4.5 is confirmed on the tested range, and the equality of the last two columns shows that every signed superposition satisfying the stability condition is realized as a spurious attractor. The counts are eight times the corresponding entries of Table 1, since the eight sign-equivalent triples generate the same weight matrix, which provides a cross-check between the two enumeration schemes.

C.4 Sampling Experiments: Verification of Spurious Attractor Completeness with $M = 3$

While Section C.3 verifies the characterization of Theorem 4.5 in small-scale HNNs, the exponential growth of 2^N makes full traversal infeasible for larger networks. Following the experimental framework of Section C.1, we therefore conduct targeted validation experiments for large-scale scenarios with $N = 100, 200, 500$, which are representative of practical HNN applications. For each N, we run 100 independent trials with different random seeds. Each trial draws one random triple of stored memories ($\xi^1, \xi^2, \xi^3 \in \{-1, +1\}^N$) and evaluates 100,000 sampled initial states, applying a targeted sampling strategy to balance computational efficiency and verification rigor. The verification procedure mirrors that of Section C.2: spurious attractors are identified according to the definition $\zeta^{\boldsymbol{\epsilon}} \notin \mathcal{A}_{\mathrm{mem}}$ and subsequently evaluated for consistency with the conditions of Theorem 4.5. The results are summarized in Table 3, quantifying the empirical performance of Theorem 4.5 in large-scale networks. Table 3 reports results from targeted sampling in regions prone to spurious attractor generation, evaluating the feasibility of Theorem 4.5 for large N. The total number of **Outside SA** (spurious attractors outside the predicted class) is zero across all large N settings, indicating no unrecognized spurious attractors exist in the sampled space—Theorem 4.5 covers all potential spurious attractors in large-scale scenarios. **Pred. SA** (predicted-class spurious attractors, 6-8 for $N = 100$, 8 for $N = 200, 500$) denotes the number of spurious attractors per given memories, and every spurious attractor found in the sampled space belongs to the class predicted by Theorem 4.5. A 100% **Consist.** (consistency rate, the ratio of identified to actual spurious attractors in the sampling) and 100% **Inv. Cons.** (inverse consistency, the fraction of spurious attractors whose negation is also spurious) further validate that the sampled spurious attractors are accounted for by

Table 3: Statistics of sampled spurious attractors. This table presents the statistical results of sampled spurious attractors for large-scale HNNs with varying sizes N. We report the number of sampled initial states per trial, spurious attractors outside the predicted class (SA), predicted-class SA, consistency rate, and inverse consistency. A key observation is that no spurious attractor outside the predicted class is detected across all network sizes, with 100% consistency and inverse consistency.

N	Samp. G.	Outside SA	Pred. SA	Consist. (%)	Inv. Cons. (%)
100	100000	0	6-8	100	100
200	100000	0	8	100	100
500	100000	0	8	100	100

the predicted class in large N scenarios (with $100{,}000$ **Samp. G.** sampled initial states per trial, over 100 independent trials for each N), confirming its practical feasibility for the real-world scene.

C.5 Scope of Empirical Validation

The results of this work are supported by complete and self-contained proofs. For the minimal case $M = 3$, the stability condition and the completeness of the predicted class are both proved in Appendix B, and the exhaustive experiments above confirm the resulting characterization with 100% consistency. The generalization to arbitrary $M \geq 3$ follows as a deductive extension of the same block-level argument and introduces no additional empirical assumptions.

D DETAILS FOR EQ. 4

We derive w_{ij} based on the Hebbian rule. Our core goal is to express w_{ij} using the symbolic centroid of the stored memories and the defined $\mathbf{\Gamma}$-matrix. This helps compress information, using a single symbolic centroid instead of multiple stored memories for expression.

Key definitions for derivation are as follows: $\xi_i^m = s_p^m \zeta_i$, where ζ_i is the value of neuron i in the symbolic centroid of the stored memories, and s_p^m is the signature entry of block A_p for the m-th memory, i.e., $s_p^m = \xi_i^m / \zeta_i \in \{-1, +1\}$.

The detailed derivation of w_{ij} is as follows:

$$w_{ij} = \frac{1}{N} \sum_{m=1}^{M} \xi_i^m \xi_j^m = \frac{1}{N} \sum_{m=1}^{M} s_p^m \zeta_i \cdot s_q^m \zeta_j = \frac{\zeta_i \zeta_j}{N} \sum_{m=1}^{M} s_p^m s_q^m \triangleq \frac{\zeta_i \zeta_j}{N} \Gamma_{pq} \tag{D.1}$$

This derivation achieves our goal: w_{ij} is expressed by the symbolic centroid (via ζ_i, ζ_j) and the $\mathbf{\Gamma}$-matrix (via Γ_{pq}). It compresses information effectively, avoiding the need to use multiple stored memories directly.

D.1 PROOF OF THEOREM 4.2

Proof. If $L_p \leq 1$, the result holds trivially. Otherwise, suppose, for contradiction, that there exist two distinct neurons $i, j \in A_p$ such that $\zeta_i x_i \neq \zeta_j x_j$. Since $\zeta_i x_i, \zeta_j x_j \in \{-1, +1\}$, without loss of generality, let $\zeta_i x_i = +1, \zeta_j x_j = -1$.

For a stable state, the necessary condition for each neuron is $x_k H_k \geq 0$. By the block-wise representation of H_k, for any $k \in A_p$,

$$x_k H_k = \frac{1}{N}(-M + K_p \zeta_k x_k) \geq 0. \tag{D.2}$$

Thus, for neuron i,

$$K_p \geq M, \tag{D.3}$$

whereas for neuron j,

$$K_p \leq -M. \tag{D.4}$$

These two conditions are **mutually incompatible** for $M > 0$. Hence, such a pair of neurons cannot exist, and therefore

$$\zeta_i x_i = \zeta_j x_j, \qquad \forall\, i, j \in A_p. \tag{D.5}$$

The result follows.

□

E Proofs of the Stability Criterion and the Fixed-Point Proposition

E.1 Proof of Theorem 4.4

Proof. At a stable state, the Block-Uniform Flip Property gives $\zeta_i x_i = \sigma_p$ for all $i \in A_p$. Hence $x_i H_i = \frac{1}{N}(\sigma_p K_p - M)$, and the stability condition $x_i H_i \geq 0$ is equivalent to $\sigma_p K_p \geq M$, which must hold for every block $p = 1, \ldots, P$. □

E.2 Proof of Proposition 4.6

Proof. By Eq. 10, an update of neuron i leaves y_i unchanged if and only if $y_i(K_p - M y_i) \geq 0$, that is, if and only if $y_i K_p \geq M$. Hence $\boldsymbol{x}$ is a fixed point exactly when this inequality holds for every neuron. Since $M > 0$, it requires $y_i = \mathrm{sign}(K_p)$ for all $i \in A_p$ together with $|K_p| \geq M$, so that the alignment is uniform within each block with $\sigma_p = \mathrm{sign}(K_p)$ and $\sigma_p K_p = |K_p| \geq M$. Conversely, if $y_i = \sigma_p$ and $\sigma_p K_p \geq M$ for every p, then $\sigma_p = \mathrm{sign}(K_p)$ and $y_i K_p = \sigma_p K_p \geq M$, so no neuron changes. The same inequality characterizes the fixed points of the synchronous map T of Section 3, so that the two update schedules share the same fixed points. □

F Formal Definitions for the Block Partition of Section 4.1

F.1 Structure of the Interaction Matrix

Lemma F.1 (Structure of the interaction matrix)**.** *Let $s_p \in \{\pm 1\}^M$ be the signature of block A_p, that is, $s_p^m = \xi_i^m/\zeta_i$ for $i \in A_p$. Then*

$$\Gamma_{pq} = M - 2\, d_H(s_p, s_q)\,, \tag{F.1}$$

where d_H is the Hamming distance on $\{\pm 1\}^M$. In particular $\Gamma_{pq} = 0$ if and only if $d_H(s_p, s_q) = M/2$, which requires M even; hence for odd M the support graph of $\mathbf{\Gamma}$ is complete.

Proof. Each of the $M - d_H(s_p, s_q)$ coordinates in which s_p and s_q agree contributes $+1$ to $\Gamma_{pq} = \sum_{m=1}^{M} s_p^m s_q^m$, and each of the $d_H(s_p, s_q)$ coordinates in which they differ contributes -1. The remaining claims are immediate from $\Gamma_{pp} = M$. □

Remark F.2. Lemma F.1 shows that $\mathbf{\Gamma}$ is an affine function of the Hamming distances between block signatures; it also implies the closed form used in Appendix G and in Appendix I, where $\Gamma_{pq} = M - 2(t_p + t_q - 2x_0)$ for blocks A_p, A_q with minority sizes t_p, t_q and x_0 the number of minority indices shared by the two blocks.

Lemma F.3 (Parity of the external support)**.** *For a block configuration $\boldsymbol{\sigma} \in \{\pm 1\}^P$, write*

$$E_p \;\coloneqq\; \sum_{q \neq p} \Gamma_{pq} \sigma_q L_q \tag{F.2}$$

for the signed external support of block A_p. Then

$$E_p \;\equiv\; M\,(N - L_p) \pmod 2, \tag{F.3}$$

independently of $\boldsymbol{\sigma}$.

Proof. Every entry of $\mathbf{\Gamma}$ is a sum of M terms equal to ± 1, so $\Gamma_{pq} = M - 2\, d_H(s_p, s_q) \equiv M \pmod 2$ by Lemma F.1, while $\sigma_q = \pm 1 \equiv 1 \pmod 2$. Hence

$$E_p \;\equiv\; \sum_{q \neq p} M\, L_q \;=\; M\,(N - L_p) \pmod 2, \tag{F.4}$$

and the right-hand side does not involve $\boldsymbol{\sigma}$. □

Remark F.4. The parity of E_p is thus a property of the block sizes rather than of the configuration. For even M every external support is even, and the value $E_p = 0$ — the case in which a block carries no signed external support at all — can occur; for odd M one has $E_p \equiv N - L_p \pmod 2$, so every block with $N - L_p$ odd has a nonvanishing external support. Together with the slack form of the criterion, Eq. 12, this pins down when the constraint on a small block is rigid: a singleton block is stable exactly when $\sigma_p E_p \geq 0$, so its sign is forced, $\sigma_p = \operatorname{sign}(E_p)$, whenever $E_p \neq 0$, whereas the value $E_p = 0$ leaves both signs admissible.

F.2 Spectrum and Gram Structure of $\mathbf{\Gamma}$

The defining relation $\Gamma_{pq} = \sum_{m=1}^{M} s_p^m s_q^m$ of Eq. 4 exhibits $\mathbf{\Gamma}$ as a Gram matrix of the block signatures. We record the consequences of this observation, and the evaluation scheme for a single candidate configuration that it provides.

Lemma F.5 (Gram structure)**.** *Let $S \in \{\pm 1\}^{P \times M}$ be the block-signature matrix, whose row p is the signature s_p of block A_p, that is, $S_{pm} = s_p^m$. Then*

$$\mathbf{\Gamma} = S S^{\mathsf{T}}, \qquad \mathbf{\Gamma}' \coloneqq S^{\mathsf{T}} S \in \mathbb{Z}^{M \times M}, \tag{F.5}$$

where $\Gamma'_{mn} = \sum_p s_p^m s_p^n$ and $\Gamma'_{mm} = P$. Consequently $\operatorname{rank}(\mathbf{\Gamma}) = \operatorname{rank}(\mathbf{\Gamma}') \leq \min(P, M)$; in particular, if $P > M$ then $\mathbf{\Gamma}$ is singular and has at least $P - M$ zero eigenvalues.

Proof. The entries are immediate from the definition of S and of $\mathbf{\Gamma}$:

$$\left(SS^{\mathsf{T}}\right)_{pq} = \sum_m s_p^m s_q^m = \Gamma_{pq}, \qquad \left(S^{\mathsf{T}}S\right)_{mm} = \sum_p \left(s_p^m\right)^2 = P. \tag{F.6}$$

The matrices SS^{T} and $S^{\mathsf{T}}S$ have the same rank as S, which is at most $\min(P, M)$; a $P \times P$ matrix of rank at most M has at least $P - M$ zero eigenvalues. □

Remark F.6. The $M \times M$ matrix $\mathbf{\Gamma}'$ admits a direct interpretation in terms of the stored memories. Choosing one representative neuron $i_p \in A_p$ per block, and using $\zeta_{i_p}^2 = 1$,

$$\Gamma'_{mn} = \sum_p s_p^m s_p^n = \sum_p \xi_{i_p}^m \xi_{i_p}^n, \tag{F.7}$$

so that $\mathbf{\Gamma}'$ is the overlap matrix of the stored memories evaluated on one representative per block. It is independent of the chosen representatives, and, unlike $\mathbf{\Gamma}$, it is unchanged both by a block flip and by the change of reference of Proposition F.10.

Lemma F.7 (Block flips act by orthogonal similarity)**.** *For every block configuration $\boldsymbol{\sigma} \in \{\pm 1\}^P$, with $D_{\boldsymbol{\sigma}} \coloneqq \mathrm{diag}(\boldsymbol{\sigma})$,*

$$\mathbf{\Gamma}^{\boldsymbol{\sigma}} = D_{\boldsymbol{\sigma}} \mathbf{\Gamma} D_{\boldsymbol{\sigma}} = \left(D_{\boldsymbol{\sigma}} S\right)\left(D_{\boldsymbol{\sigma}} S\right)^{\mathsf{T}}, \qquad \left(D_{\boldsymbol{\sigma}} S\right)^{\mathsf{T}}\left(D_{\boldsymbol{\sigma}} S\right) = S^{\mathsf{T}} S = \mathbf{\Gamma}'. \tag{F.8}$$

Since $D_{\boldsymbol{\sigma}}$ is a signed permutation matrix and hence orthogonal, all 2^P matrices $\mathbf{\Gamma}^{\boldsymbol{\sigma}}$ are orthogonally similar to $\mathbf{\Gamma}$: for every $\boldsymbol{\sigma}$,

$$\mathrm{spec}(\mathbf{\Gamma}^{\boldsymbol{\sigma}}) = \mathrm{spec}(\mathbf{\Gamma}), \qquad \mathrm{rank}(\mathbf{\Gamma}^{\boldsymbol{\sigma}}) = \mathrm{rank}(\mathbf{\Gamma}), \qquad \left(\mathbf{\Gamma}^{\boldsymbol{\sigma}}\right)_{pp} = \Gamma_{pp} = M. \tag{F.9}$$

Proof. The first identity is the definition of Eq. 8; substituting $\mathbf{\Gamma} = SS^{\mathsf{T}}$ from Lemma F.5 gives $\mathbf{\Gamma}^{\boldsymbol{\sigma}} = D_{\boldsymbol{\sigma}} SS^{\mathsf{T}} D_{\boldsymbol{\sigma}} = \left(D_{\boldsymbol{\sigma}} S\right)\left(D_{\boldsymbol{\sigma}} S\right)^{\mathsf{T}}$, and $\left(D_{\boldsymbol{\sigma}} S\right)^{\mathsf{T}}\left(D_{\boldsymbol{\sigma}} S\right) = S^{\mathsf{T}} D_{\boldsymbol{\sigma}}^2 S = S^{\mathsf{T}} S$. A signed permutation matrix satisfies $D_{\boldsymbol{\sigma}}^{\mathsf{T}} = D_{\boldsymbol{\sigma}}^{-1}$, so that $D_{\boldsymbol{\sigma}} \mathbf{\Gamma} D_{\boldsymbol{\sigma}}$ is a similarity by an orthogonal matrix, which preserves the spectrum and the rank; the diagonal entries are unchanged because $\sigma_p^2 = 1$. □

Remark F.8 (Certifying a configuration without forming $\mathbf{\Gamma}^{\boldsymbol{\sigma}}$). Because $\mathbf{\Gamma}^{\boldsymbol{\sigma}} \boldsymbol{L} = D_{\boldsymbol{\sigma}} SS^{\mathsf{T}} D_{\boldsymbol{\sigma}} \boldsymbol{L}$, the criterion $\mathbf{\Gamma}^{\boldsymbol{\sigma}} \boldsymbol{L} \succeq M \mathbf{1}_P$ of Eq. 8 can be evaluated directly from S, $\boldsymbol{L}$ and $\boldsymbol{\sigma}$, **without materializing either $\mathbf{\Gamma}$ or $\mathbf{\Gamma}^{\boldsymbol{\sigma}}$**: form $\boldsymbol{w} = S^{\mathsf{T}}(\boldsymbol{\sigma} \odot \boldsymbol{L}) \in \mathbb{R}^M$ and test $\boldsymbol{\sigma} \odot (S\boldsymbol{w}) \succeq M \mathbf{1}_P$. The cost is $\mathcal{O}(PM)$, against $\mathcal{O}(P^2 M)$ for the explicit construction of $\mathbf{\Gamma}$ and $\mathcal{O}(P)$ for a lookup once $\mathbf{\Gamma}$ is stored; a single candidate is therefore certified at the cost of one multiplication by S and one by S^{T}.

Remark F.9 (What the spectral structure does and does not provide). Two consequences should be kept apart. On the computational side, $\mathrm{rank}(\mathbf{\Gamma}) \leq M$ compresses the network to an M-dimensional signature space, and Eq. F.8 shows that forming $\mathbf{\Gamma}^{\boldsymbol{\sigma}}$ is never necessary, so that the criterion of any candidate configuration is a single multiplication by S and S^{T}. On the structural side, neither the spectrum nor the rank can decide stability: by Eq. F.9 the spectrum is the same for every block configuration, whereas the stable set $\mathcal{S}$ of Eq. 8 contains some configurations and not others. Stability is therefore not a spectral property of $\mathbf{\Gamma}$, but a property of the signed data $(S, \boldsymbol{L}, \boldsymbol{\sigma})$ entering the inequality $\mathbf{\Gamma}^{\boldsymbol{\sigma}} \boldsymbol{L} \succeq M \mathbf{1}_P$, and the signatures s_p^m have to be retained for that purpose.

F.3 Gauge Invariance of the Reference Centroid

The block decomposition of Section 4.1 is stated relative to a reference sequence, taken in the main text to be the symbolic centroid $\boldsymbol{\zeta}$ of Eq. 3. We now show that this choice carries no physical content: two references that differ by a block-uniform sign factor define signatures, an interaction matrix and a criterion that are related by an orthogonal similarity, hence describe the same stability problem.

Proposition F.10 (Gauge invariance)**.** *Let $\boldsymbol{\zeta}$ be a reference for which $\xi_i^m \zeta_i = s_p^m$ is well defined on every block, and let $\boldsymbol{\eta} \in \{\pm 1\}^N$ be block-uniform with block values η_p, that is, $\eta_i = \eta_p$ for $i \in A_p$. Put $\boldsymbol{\zeta}' = \boldsymbol{\eta} \odot \boldsymbol{\zeta}$, and let $S', \mathbf{\Gamma}'$ and $\boldsymbol{\sigma}', \boldsymbol{K}'$ with $K'_p = \sum_q \Gamma'_{pq} \sigma'_q L_q$ be the quantities built from $\boldsymbol{\zeta}'$ in place of $\boldsymbol{\zeta}$ in Eqs. 4 and 6. Then*

$$S' = D_{\boldsymbol{\eta}} S, \qquad \mathbf{\Gamma}' = D_{\boldsymbol{\eta}} \mathbf{\Gamma} D_{\boldsymbol{\eta}}, \qquad \boldsymbol{\sigma}' = D_{\boldsymbol{\eta}} \boldsymbol{\sigma}, \qquad \boldsymbol{K}' = D_{\boldsymbol{\eta}} \boldsymbol{K}, \tag{F.10}$$

while the stability input $\sigma_p K_p$ is unchanged block by block, so that $\sigma'_p K'_p \geq M$ if and only if $\sigma_p K_p \geq M$; equivalently, $\boldsymbol{\sigma}' \in \mathcal{S}'$ if and only if $D_{\boldsymbol{\eta}} \boldsymbol{\sigma}' \in \mathcal{S}$ for the stable set of Eq. 8. The local field H_i of Eq. 7, the spectrum $\mathrm{spec}(\mathbf{\Gamma})$, the products of $\mathbf{\Gamma}$ along any closed walk, and the diagonal $\Gamma_{pp} = M$ are likewise invariant, and $\mathbf{\Gamma}'^{\boldsymbol{\sigma}'} = \mathbf{\Gamma}^{\boldsymbol{\sigma}}$.

Proof. For $i \in A_p$ the signature of block A_p under the new reference is $s'^m_p = \xi^m_i \zeta'_i = \eta_p \xi^m_i \zeta_i = \eta_p s^m_p$, which is $S' = D_{\boldsymbol{\eta}} S$ and gives $\Gamma'_{pq} = \sum_m s'^m_p s'^m_q = \eta_p \eta_q \Gamma_{pq}$, that is, $\boldsymbol{\Gamma}' = D_{\boldsymbol{\eta}} \boldsymbol{\Gamma} D_{\boldsymbol{\eta}}$. The flip symbol is defined by $x_i = \zeta'_i \sigma'_p$ for $i \in A_p$, so $\sigma'_p = \zeta'_i \zeta_i \sigma_p = \eta_p \sigma_p$; substituting both relations into $K'_p = \sum_q \Gamma'_{pq} \sigma'_q L_q$ yields $K'_p = \eta_p \sum_q \Gamma_{pq} \sigma_q L_q = \eta_p K_p$, whence $\sigma'_p K'_p = \sigma_p K_p$ block by block, and the criterion is satisfied by the same configurations. For the local field, Eq. 7 gives

$$\frac{\zeta'_i}{N} \left(K'_p - M \zeta'_i x_i \right) = \frac{\eta_p \zeta_i}{N} \left(\eta_p K_p - M \eta_p \zeta_i x_i \right) = \frac{\zeta_i}{N} \left(K_p - M \zeta_i x_i \right) = H_i, \tag{F.11}$$

so that H_i is unchanged neuron by neuron. Since $D_{\boldsymbol{\eta}}$ is orthogonal, $\boldsymbol{\Gamma}'$ is orthogonally similar to $\boldsymbol{\Gamma}$ and the spectrum is preserved; along a closed walk every vertex contributes its value of $\boldsymbol{\eta}$ to exactly two edges, so that each factor η_p is paired with itself and every walk product is invariant. Finally,

$$\boldsymbol{\Gamma}'^{\boldsymbol{\sigma}'} = D_{\boldsymbol{\sigma}'} \boldsymbol{\Gamma}' D_{\boldsymbol{\sigma}'} = D_{\boldsymbol{\eta}} D_{\boldsymbol{\sigma}} D_{\boldsymbol{\eta}} \boldsymbol{\Gamma} D_{\boldsymbol{\eta}} D_{\boldsymbol{\sigma}} D_{\boldsymbol{\eta}} = D_{\boldsymbol{\sigma}} \boldsymbol{\Gamma} D_{\boldsymbol{\sigma}} = \boldsymbol{\Gamma}^{\boldsymbol{\sigma}}, \tag{F.12}$$

because diagonal matrices commute, so the matrix entering the criterion is the same in both references. □

Corollary F.11 (The memory combinations form one gauge orbit)**.** *Take $\boldsymbol{\zeta}$ to be the centroid of the stored memories, and for a sign combination $\boldsymbol{\varepsilon} \in \{\pm 1\}^M$ let $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}}$ be the centroid of $(\varepsilon_1 \xi^1, \ldots, \varepsilon_M \xi^M)$ formed with the convention of Eq. 3. Then $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}}$ induces the same block partition as $\boldsymbol{\zeta}$, and the ratio $\eta_i = \zeta^{\boldsymbol{\varepsilon}}_i / \zeta_i$ is block-uniform. Consequently the 2^M combinations form a single gauge orbit: their interaction matrices are related by Eq. F.10 and are therefore orthogonally similar, and the status of $\boldsymbol{\zeta}^{\boldsymbol{\varepsilon}}$ is decided by the criterion of Eq. 8 evaluated at the configuration $\boldsymbol{\sigma} = \boldsymbol{\eta}$ of the reference gauge, the pairs $\boldsymbol{\varepsilon}$ and $-\boldsymbol{\varepsilon}$ corresponding to $\boldsymbol{x}$ and $-\boldsymbol{x}$. In particular the eight combinations of the $M = 3$ analysis are not eight independent stability problems.*

Proof. The partition of Definition 4.1 depends on the stored memories only: two neurons lie in the same block exactly when their memory-alignment patterns $(\xi^1_i, \ldots, \xi^M_i)$ agree up to an overall sign, a relation that is unaffected by negating any subset of the memories. For $i \in A_p$ the combination sum factorizes as

$$\sum_{m=1}^{M} \varepsilon_m \xi^m_i = \zeta_i \, \alpha_p, \qquad \alpha_p \coloneqq \sum_{m=1}^{M} \varepsilon_m s^m_p. \tag{F.13}$$

If $\alpha_p \neq 0$, then $\zeta^{\boldsymbol{\varepsilon}}_i = \operatorname{sign}(\zeta_i \alpha_p) = \zeta_i \operatorname{sign}(\alpha_p)$; if $\alpha_p = 0$, which requires even M, the convention assigns the value of the first memory of the combination, $\zeta^{\boldsymbol{\varepsilon}}_i = \varepsilon_1 \xi^1_i = \varepsilon_1 s^1_p \zeta_i$. In both cases η_i is constant on A_p, so that Proposition F.10 applies to the pair $(\boldsymbol{\zeta}, \boldsymbol{\eta})$. Negating the combination negates α_p and ε_1 alike, hence reverses $\boldsymbol{\eta}$ and with it the state. □

Remark F.12 (Role of the tie-breaking rule). A tie, $\sum_m \xi^m_i = 0$, can occur only for even M, and Corollary F.11 relies on the convention of Eq. 3, which resolves it by the value of the first memory of the combination. Such a convention is well behaved precisely because the value of a stored memory is itself block-uniform up to the signature s^1_p: the ratio η_i then remains constant on each block even on tied neurons. A convention that does not refer to the memories — assigning, for instance, a fixed value to every tie — can instead make η_i vary between neurons of one and the same block when that block contains two neurons whose memory patterns are negatives of each other, in which case the signature s^m_p and hence $\boldsymbol{\Gamma}$ would no longer be well defined. The convention adopted in Section 3 is therefore not a mere bookkeeping choice, but the one that keeps the block description valid for every combination.

F.4 Block Counting

Each block is determined by the signature $s \in \{\pm 1\}^M$ of its neurons, and a signature is realized only if it is compatible with the centroid construction: either $\sum_m s^m > 0$, or $\sum_m s^m = 0$ together with $s^1 = +1$ under the tie-breaking rule of the symbolic centroid (Eq. 3). Writing n_t for the number of blocks whose signature has minority size $t = \#\{m : s^m = -1\}$,

$$n_t = \begin{cases} 1, & t = 0, \\ \binom{M}{t}, & 2t < M, \\ \frac{1}{2}\binom{M}{t}, & 2t = M \text{ (even } M \text{ only)}. \end{cases} \tag{F.14}$$

The last line accounts for the tie-breaking rule, which identifies the two orientations of a half–half split. Summing equation F.14 over t gives

$$P = \sum_t n_t = \sum_{t<M/2} \binom{M}{t} + \tfrac{1}{2}\binom{M}{M/2} = 2^{M-1} \qquad \text{(the tie term being absent for odd } M\text{)}, \tag{F.15}$$

and therefore $P \le \min(2^{M-1}, N)$. No contiguity or ordering of the minority indices is assumed: each block corresponds to an arbitrary subset of memories of size t (with the tie-breaking rule above).

F.5 Worked Instance of the Block Decomposition

With $M = 3$ and $N = 5$, the memories $\boldsymbol{\xi}^1 = (+,-,+,+,+)$, $\boldsymbol{\xi}^2 = (+,+,+,-,+)$ and $\boldsymbol{\xi}^3 = (+,+,-,+,+)$ (writing $\pm$ for ± 1) have $\boldsymbol{\zeta} = \mathbf{1}_5$ and agree at neurons 1 and 5, which therefore share the signature $(+,+,+)$ and form the block A_1 of length $L_1 = 2$, whereas neurons $2, 3, 4$ carry the distinct signatures $(-,+,+), (+,+,-), (+,-,+)$ and form singleton blocks, so that $P = 4$ and $\boldsymbol{L} = (2,1,1,1)^\mathsf{T}$.

F.6 Definition of Bit-level Signature

Definition F.13 (Bit-level Signature)**.** For each neuron i, its *bit-level signature* is the minority index set

$$\mathcal{M}_i \coloneqq \{m \in \{1,\dots,M\} : \xi_i^m \zeta_i = -1\}, \tag{F.16}$$

that is, the set of memories in which neuron i disagrees with the symbolic centroid $\boldsymbol{\zeta}$ of Eq. 3.

Two neurons i and j belong to the same block if and only if their bit-level signatures are identical, i.e., $\mathcal{M}_i = \mathcal{M}_j$.

The proof proceeds as follows:

- If i and j are in the same block, they share the same signature $s_p \in \{\pm 1\}^M$, so they disagree with the reference centroid on exactly the same memories and $\mathcal{M}_i = \mathcal{M}_j$.
- If $\mathcal{M}_i = \mathcal{M}_j$, then the memory patterns of the two neurons agree up to the common factor $\zeta_i \zeta_j$, so they lie in the same block by Definition 4.1.

G COMPLETE EXAMPLE FOR $M = 4$

This section provides a **closed-form minimal example** for $M = 4$ (4 stored binary memories) to verify the applicability of the block decomposition, the block counting of Appendix F.4, the $\mathbf{\Gamma}$-matrix construction, and the stability criterion framework for $M > 3$. All definitions, notations, and formulas strictly follow the main text.

G.1 MINIMAL EXPERIMENTAL SETUP

- Number of stored memories: $M = 4$

- Number of neurons: $N = 8$ (matches the maximal block count for $M = 4$)

- State space: $\mathcal{X} = \{-1, +1\}^8$

- Block-size vector: $\boldsymbol{L} = [1, 1, 1, 1, 1, 1, 1, 1]^\top$ (each block contains exactly 1 neuron)

- Symbolic centroid (Eq. 3): computed below; it is the reference relative to which the block partition and all block quantities are evaluated.

- Stored memories: $\xi^1, \xi^2, \xi^3, \xi^4 \in \mathcal{X}$ (designed to make all blocks non-empty), listed below their centroid

$$\begin{aligned} \xi^1 &= [1,\ 1,\ 1,\ 1,\ -1,\ 1,\ 1,\ 1], \\ \xi^2 &= [1,\ 1,\ 1,\ -1,\ 1,\ 1,\ -1,\ -1], \\ \xi^3 &= [1,\ 1,\ -1,\ 1,\ 1,\ -1,\ 1,\ -1], \\ \xi^4 &= [1,\ -1,\ 1,\ 1,\ 1,\ -1,\ -1,\ 1]. \end{aligned} \tag{G.1}$$

The element-wise sums of the four memories are $[4, 2, 2, 2, 2, 0, 0, 0]$. For an even number of memories the majority vote can tie, as it does here at neurons 6–8; the tie-breaking rule of Eq. 3 then assigns to such a neuron the value of the first memory, giving $\zeta_6 = \zeta_7 = \zeta_8 = +1$ and hence the symbolic centroid

$$\boldsymbol{\zeta} = [1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1,\ 1]^\top = \mathbf{1}_8. \tag{G.2}$$

G.2 BLOCK PARTITION VIA BIT-LEVEL SIGNATURE

Following Definition F.13 and the block consistency condition (Definition 4.1), neurons are partitioned into $P = 8$ disjoint blocks, indexed by the minority index set of their bit-level signature:

$$\begin{aligned} A_1 &= A_{1234}, \quad A_2 = A_{234|1}, \quad A_3 = A_{134|2}, \quad A_4 = A_{124|3}, \\ A_5 &= A_{123|4}, \quad A_6 = A_{12|34}, \quad A_7 = A_{13|24}, \quad A_8 = A_{14|23}. \end{aligned} \tag{G.3}$$

Here A_{1234} collects the neurons whose signature is empty, $A_{pqr|s}$ those whose signature is the singleton $\{s\}$, and $A_{pq|rs}$ those with a two-element signature; the notation lists the majority memories before the bar. By the block definition, neurons within each block share identical update dynamics.

G.3 $\mathbf{\Gamma}$-MATRIX CONSTRUCTION

The interaction coefficient follows from its definition $\Gamma_{pq} = \sum_{m=1}^{M} s_p^m s_q^m$ (Eq. 4); by Lemma F.1 it is equivalently the Hamming form $\Gamma_{pq} = M - 2\, d_H(s_p, s_q)$, and expanding the distance through the minority sizes gives

$$\Gamma_{pq} = M - 2\,(t_p + t_q - 2x_0)\,, \tag{G.4}$$

where $t_p = |\mathcal{M}_p|$ is the minority size of block A_p (Appendix F.4) and $x_0 = |\mathcal{M}_p \cap \mathcal{M}_q|$ denotes the number of minority indices shared by the two blocks, that is, the positions where both signatures take -1. The symmetric 8×8 $\mathbf{\Gamma}$-matrix is constructed as:

$$\mathbf{\Gamma} = \begin{pmatrix} 4 & 2 & 2 & 2 & 2 & 0 & 0 & 0 \\ 2 & 4 & 0 & 0 & 0 & -2 & -2 & -2 \\ 2 & 0 & 4 & 0 & 0 & -2 & 2 & 2 \\ 2 & 0 & 0 & 4 & 0 & 2 & -2 & 2 \\ 2 & 0 & 0 & 0 & 4 & 2 & 2 & -2 \\ 0 & -2 & -2 & 2 & 2 & 4 & 0 & 0 \\ 0 & -2 & 2 & -2 & 2 & 0 & 4 & 0 \\ 0 & -2 & 2 & 2 & -2 & 0 & 0 & 4 \end{pmatrix}. \tag{G.5}$$

Since every block has size 1 and $\boldsymbol{\sigma} = \mathbf{1}_8$ holds for the centroid configuration, K_p of Eq. 6 reduces to the row sums of $\boldsymbol{\Gamma}$, $\boldsymbol{K} = [12,\ 0,\ 8,\ 8,\ 8,\ 4,\ 4,\ 4]^\top$.

The candidate under test is the symbolic centroid $\boldsymbol{\zeta} = \mathbf{1}_8$ itself, of block configuration $\boldsymbol{\sigma} = \mathbf{1}_8$. Substituting into the stability criterion $\boldsymbol{\Gamma}^{\boldsymbol{\sigma}} \boldsymbol{L} \succeq 4 \cdot \mathbf{1}_8$ of Eq. 8, we find that it **does not** hold: block $A_{234|1}$ (neuron 5) gives $\sigma_2 K_2 = 0 < 4$ for $M = 4$, so the symbolic centroid is *not* a stable state of this network. Evaluating Eq. 7 block by block gives

$$H = \frac{1}{N}\left(\boldsymbol{\zeta} \odot (\boldsymbol{K} - M\boldsymbol{\zeta} \odot \boldsymbol{x})\right) = [1.0,\ 0.5,\ 0.5,\ 0.5,\ -0.5,\ 0.0,\ 0.0,\ 0.0]^\top, \tag{G.6}$$

which agrees component-wise with the direct evaluation $W\boldsymbol{x}$ of the weight matrix. In particular $H_5 = -0.5$ opposes $x_5 = +1$, so neuron 5 flips and $\boldsymbol{\zeta}$ is not a fixed point, in agreement with the criterion above.

The four stored memories are representatives of four of the eight blocks: ξ^1 has minority set $\{4\}$ and belongs to $A_{123|4}$, ξ^2 has $\{3\}$ and belongs to $A_{124|3}$, ξ^3 has $\{2\}$ and belongs to $A_{134|2}$, and ξ^4 has $\{1\}$ and belongs to $A_{234|1}$. Each of them is a fixed point of the network, whereas the symbolic centroid fails the criterion above, so the example separates the stored attractors from a non-stored centroid configuration within one and the same block structure.

H EXACT ELIMINATION OF A SPURIOUS ATTRACTOR

As an application of the criterion, this section removes a certified spurious attractor from the weight matrix on the two concrete cases $M = 3$ and $M = 4$. Once Eq. 8 has identified the spurious configurations, the corresponding Hebbian term is subtracted in a targeted way, and the effect on every other configuration is again predicted in closed form by the same block quantities; the strength is confined to an admissible interval obtained by the same enumeration that computes $\mathcal{S}_{\mathrm{sp}}$. Removals of several spurious attractors at once, whose corrections interact, are left to future work. All claims below are verified against exhaustive enumeration of the 2^N states and against both update schedules.

A note on the asynchronous counts is in order, because they are not schedule-free. Throughout this section, the asynchronous dynamics updates one unstable neuron at a time, namely the unstable neuron of smallest index, and recomputes the local fields after every update, with a neuron whose local field is zero (Section 3) left untouched. The pre-intervention basin of the spurious pair depends on this choice: it holds 616 of the 2^{11} initial states at $M = 3$ under the rule above, against 148, 778, and 884 under a reverse-index rule, a cyclic sweep, and a rule that always flips the most strongly opposing neuron, and 1124 at $M = 4$ against 272, 1108, and 722. After the correction the dependence disappears: the spurious pair is no longer a fixed point, every fair schedule terminates at a fixed point, and the post-deletion fixed points are exactly the stored states, so all 2048 initial states reach a stored memory under each of these rules as well. The asynchronous rows of Tables 4 and 5, and the basin sizes of Fig. 3, are those of the index-ordered rule.

H.1 REMOVING ONE SPURIOUS ATTRACTOR

Let $\{A_p\}_{p=1}^{P}$, $\boldsymbol{L}$, $\boldsymbol{\Gamma}$ be the block structure induced by the stored memories, and let $\boldsymbol{x}^* \in \mathcal{S}_{\mathrm{sp}}$ have block configuration $\boldsymbol{\sigma}^*$, i.e., $x_i^* = \zeta_i \sigma_p^*$ for $i \in A_p$. For $\varepsilon > 0$, define the modified weight matrix

$$W(\varepsilon) \coloneqq W - \frac{\varepsilon}{N}\left(\boldsymbol{x}^* \boldsymbol{x}^{*\mathsf{T}} - I_N\right), \tag{H.1}$$

so that the Hebbian contribution of $\boldsymbol{x}^*$ is subtracted with strength ε; $\varepsilon = 1$ removes this contribution completely, and $\varepsilon = 0$ recovers the original network. This is the unlearning update (Hopfield et al., 1983), applied here to a single certified state and with the strength fixed by the criterion rather than tuned empirically; Appendix H.4 compares the two settings.

Proposition H.1. *A configuration $x_i = \zeta_i \sigma_p$ for $i \in A_p$, $\boldsymbol{\sigma} \in \{-1, +1\}^P$, is a fixed point of $W(\varepsilon)$ if and only if*

$$\sigma_p \left[\boldsymbol{\Gamma}\left(\boldsymbol{\sigma} \odot \boldsymbol{L}\right)\right]_p \;\geq\; M - \varepsilon + \varepsilon\, \sigma_p \sigma_p^*\, S(\boldsymbol{\sigma}, \boldsymbol{\sigma}^*), \qquad p = 1, \ldots, P, \tag{H.2}$$

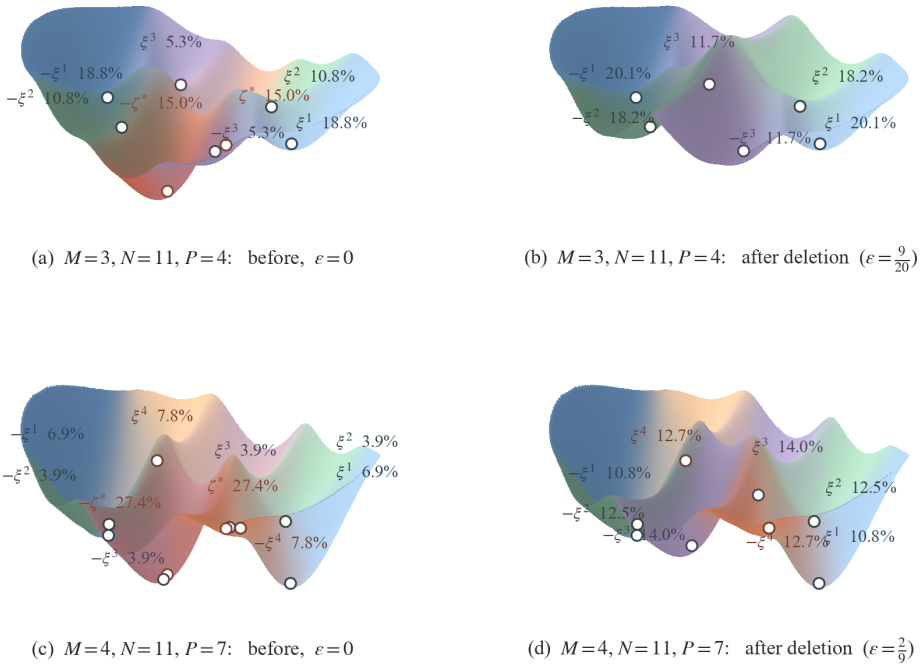
where $S(\boldsymbol{\sigma}, \boldsymbol{\sigma}^*) \coloneqq \sum_{q=1}^{P} L_q \sigma_q \sigma_q^* = \langle \boldsymbol{x}, \boldsymbol{x}^* \rangle \in [-N, N]$ is the signed overlap between the two states.

Proof. For $i \neq j$, the modified weights read $w_{ij}(\varepsilon) = (\zeta_i \zeta_j / N)\Gamma'_{pq}$ with $\Gamma' = \boldsymbol{\Gamma} - \varepsilon \boldsymbol{\sigma}^* \boldsymbol{\sigma}^{*\mathsf{T}}$, while the $-\varepsilon I_N$ normalization keeps the diagonal of $W(\varepsilon)$ at zero, so that the $j = i$ term contributes $+\varepsilon x_i / N$ to the local field. Repeating the local-field decomposition of Eq. 5 with $\boldsymbol{\Gamma}'$ in place of $\boldsymbol{\Gamma}$ gives $x_i H_i' = \frac{1}{N}(y_i K_p' - (M - \varepsilon))$ with $y_i = \zeta_i x_i$ and $K_p' = \sum_q \Gamma'_{pq}(L_q - 2d_q)$. Since $M - \varepsilon > 0$, the argument of Theorem 4.2 applies verbatim, so every fixed point of $W(\varepsilon)$ is block-uniform; substituting $L_q - 2d_q = \sigma_q L_q$ yields Eq. H.2. □

Two consequences are used below. First, $\boldsymbol{x}^*$ itself survives the intervention if and only if $\sigma_p^*[\boldsymbol{\Gamma}(\boldsymbol{\sigma}^* \odot \boldsymbol{L})]_p \geq M + \varepsilon(N - 1)$ for all p; since the left-hand side is bounded by MN, this fails for every ε exceeding the critical strength

$$\varepsilon_c \coloneqq \frac{1}{N-1} \min_p \left(\sigma_p^* \left[\boldsymbol{\Gamma}\left(\boldsymbol{\sigma}^* \odot \boldsymbol{L}\right)\right]_p - M\right), \tag{H.3}$$

which is computable in $O(P)$ from the block quantities alone, and by symmetry $-\boldsymbol{x}^*$ is eliminated together with $\boldsymbol{x}^*$. Second, Eq. H.2 is linear in ε for each of the 2^P configurations, so the set of deletion strengths that remove the spurious pair while leaving the stored states untouched and creating no further stable configuration is an interval $(\varepsilon_{\mathrm{lo}}, \varepsilon_{\mathrm{hi}})$ obtained by the same variable-assignment enumeration used to compute S_{sp}. In both cases below the full-strength correction $\varepsilon = 1$ lies outside this interval: it would leave 6 and 16 additional stable block configurations (the $M = 3$ and $M = 4$ cases, respectively) beyond the stored set, so the strength has to be chosen inside the interval. The two cases use a verified strength strictly inside it.



(a) $M=3, N=11, P=4$: before, $\varepsilon=0$

(b) $M=3, N=11, P=4$: after deletion ($\varepsilon=\frac{9}{20}$)

(c) $M=4, N=11, P=7$: before, $\varepsilon=0$

(d) $M=4, N=11, P=7$: after deletion ($\varepsilon=\frac{2}{9}$)

Figure 3: Exact basins of attraction and exact removal of a spurious attractor, for the two networks of Eqs. H.4 and H.5 ($N = 11$). Each landscape carries one well per attractor, its floor marked by a small circle: the depth of a well is set by the exact energy $E(\boldsymbol{x}^*)$ of its attractor, and the label reports the exact basin size, obtained by following all 2^{11} initial states under the asynchronous dynamics. (a,b) $M = 3$, $P = 4$. Before the intervention the spurious centroid pair $\pm\boldsymbol{\zeta}$ holds 15.0% of the state space each; after the rank-one correction $W(\varepsilon)$ with $\varepsilon = 9/20$ those wells are gone and their basin is redistributed over the six stored states. (c,d) $M = 4$, $P = 7$, $\varepsilon = 2/9$. The spurious pair holds 27.4% each, and after the correction all 2048 initial states terminate at a stored memory. Percentages and depths are exact; the placement of the wells in the plane is schematic.

H.2 Case $M = 3$: $N = 11$, $P = 4$

We store three memories at $N = 11$,

$$\begin{aligned} \xi^1 &= [+1,+1,+1,+1,+1,+1,+1,+1,+1,-1,-1], \\ \xi^2 &= [+1,+1,+1,+1,+1,+1,+1,-1,-1,+1,+1], \\ \xi^3 &= [+1,+1,+1,+1,+1,-1,-1,+1,+1,+1,+1], \end{aligned} \tag{H.4}$$

for which $\zeta = \mathbf{1}$ and the blocks are A_{123} (neurons 1–5, $L = 5$) and $A_{12|3}, A_{13|2}, A_{23|1}$ (two neurons each); the interaction matrix is the $M = 3$ matrix of Section B and $K(\boldsymbol{\zeta}) = (7, 7, 7, 21)$ with margins $(4, 4, 4, 18)$. The variable-assignment enumeration over the $2^4 = 16$ block configurations returns $\mathcal{S}_{\mathrm{sp}} = \{\boldsymbol{\zeta}, -\boldsymbol{\zeta}\}$: at most two spurious attractors exist, and the enumeration reproduces the 8 fixed points of the network found by exhaustive simulation over all $2^{11} = 2048$ states, under both synchronous and asynchronous updating.

We delete $\boldsymbol{x}^* = \boldsymbol{\zeta}$, i.e., $\boldsymbol{\sigma}^* = \mathbf{1}_P$; Eq. H.3 gives $\varepsilon_c = 4/10 = 2/5$, and the admissible interval is $(2/5, 1/2)$. With $\varepsilon = 9/20$, the post-deletion fixed-point set predicted by Eq. H.2 consists of the six stored states $\{\pm\xi^1, \pm\xi^2, \pm\xi^3\}$ and nothing else, and this coincides exactly with the simulated fixed-point set; in particular $\pm\boldsymbol{x}^*$ are no longer fixed points and no new attractor is created. Table 4 reports the basins.

Table 4: Exact elimination of the spurious centroid for the $M = 3$ case of Eq. H.4 ($N = 11$, $\varepsilon = 9/20$). Basins are counted over all 2^{11} initial states; each probe is a stored state with a single bit flipped (Hamming radius $r = 1$) and is followed under the synchronous dynamics.

	Before ($\varepsilon = 0$)	After ($\varepsilon = 9/20$)
spurious attractors	$\{\boldsymbol{\zeta}, -\boldsymbol{\zeta}\}$	none
synth. basin of $\pm\boldsymbol{\zeta}$	836 (40.8%)	0
synth. basin of each stored state	190 (9.3%)	182 (8.9%)
asynch. basin of $\pm\boldsymbol{\zeta}$	616 (30.1%)	0
asynch. basin of stored states	1432 (69.9%)	2048 (100%)
single-bit probes retrieved ($r = 1$)	54/66 (81.8%)	66/66 (100%)

Two points are worth noting. First, the intervention is surgical in the intended sense: every initial state that the asynchronous dynamics previously drove into the spurious pair now terminates at a stored memory, and no stored memory loses its basin. Second, under synchronous updating the same states may fall into period-2 orbits rather than fixed points (Section 4.5); this is a property of the synchronous schedule, not of the deletion, and it does not arise under the asynchronous rule, which provably terminates at a fixed point.

H.3 Case $M = 4$: $N = 11$, $P = 7$

We store four memories at $N = 11$,

$$\begin{aligned}
\xi^1 &= [+1,+1,-1,-1,+1,+1,+1,+1,+1,+1,+1],\\
\xi^2 &= [+1,+1,+1,+1,-1,+1,+1,+1,+1,+1,-1],\\
\xi^3 &= [+1,+1,+1,+1,+1,-1,+1,+1,+1,-1,+1],\\
\xi^4 &= [+1,+1,+1,+1,+1,+1,-1,-1,-1,-1,-1],
\end{aligned} \tag{H.5}$$

for which $\boldsymbol{\zeta} = \mathbf{1}$ and the block decomposition yields $P = 7$ blocks with $\boldsymbol{L} = (2,1,1,1,1,3,2)$, interaction matrix

$$\boldsymbol{\Gamma} = \begin{pmatrix}
4 & -2 & 0 & -2 & 0 & 0 & 2\\
-2 & 4 & 2 & 0 & -2 & 2 & 0\\
0 & 2 & 4 & -2 & 0 & 0 & 2\\
-2 & 0 & -2 & 4 & 2 & 2 & 0\\
0 & -2 & 0 & 2 & 4 & 0 & 2\\
0 & 2 & 0 & 2 & 0 & 4 & 2\\
2 & 0 & 2 & 0 & 2 & 2 & 4
\end{pmatrix}, \tag{H.6}$$

and $K(\boldsymbol{\zeta}) = (8,6,8,6,8,20,22)$ with margins $(4,2,4,2,4,16,18)$. Enumeration of the $2^7 = 128$ block configurations again returns $\mathcal{S}_{\mathrm{sp}} = \{\boldsymbol{\zeta}, -\boldsymbol{\zeta}\}$, in exact agreement with the 10 fixed points found by exhaustive simulation of all 2^{11} states.

Deleting $\boldsymbol{x}^* = \boldsymbol{\zeta}$ gives $\varepsilon_c = 2/10 = 1/5$ and admissible interval $(1/5, 1/4)$; we use $\varepsilon = 2/9$. The predicted post-deletion fixed-point set is exactly the eight stored states $\{\pm\xi^m\}_{m=1}^4$, again matching simulation, and Table 5 summarizes the basins. Here the spurious pair captures more than half of the state space under asynchronous updating; after the single rank-one correction all 2048 initial states converge to a stored memory.

Table 5: Exact elimination of the spurious centroid for the $M = 4$ case of Eq. H.5 ($N = 11$, $\varepsilon = 2/9$). Basins are counted over all 2^{11} initial states; each probe is a stored state with a single bit flipped (Hamming radius $r = 1$) and is followed under the synchronous dynamics.

	Before ($\varepsilon = 0$)	After ($\varepsilon = 2/9$)
spurious attractors	$\{\boldsymbol{\zeta}, -\boldsymbol{\zeta}\}$	none
synth. basin of $\pm\boldsymbol{\zeta}$	764 (37.3%)	0
asynch. basin of $\pm\boldsymbol{\zeta}$	1124 (54.9%)	0
asynch. basin of stored states	924 (45.1%)	2048 (100%)
single-bit probes retrieved ($r = 1$)	56/88 (63.6%)	72/88 (81.8%)

Taken together, the two cases illustrate the intended use of the framework: the block-level constants certify a priori that a candidate is spurious, give the exact strength ε_c at which a rank-one correction removes it, and predict the complete post-intervention attractor set through the same enumeration that produced $\mathcal{S}_{\mathrm{sp}}$. The intervention is thus not a heuristic perturbation but a verified edit of the stored set. Extending the procedure to the successive removal of several spurious attractors, and to memories whose block structure is not known in advance, is left for future work.

H.4 Relation to Existing Unlearning Schemes

Unlearning was introduced as a way of removing spurious attractors rather than as an edit of a chosen state: the rule of Eq. H.1 is applied iteratively to states visited during a stochastic unlearning phase, so which states are removed and which stored memories are weakened follows a posteriori from the visited sequence, and the unlearning strength is a tuned parameter (Hopfield et al., 1983; Van Hemmen, 1997). The Hebbian-unlearning and dreaming schemes that followed retain this iterative form, and their effect is characterized in the statistical-mechanical setting as an average over the memory ensemble (Fachechi et al., 2019); suppression methods of the same period act on the energy function itself, through penalty terms for constraint violations, rather than on one stored configuration (Abe, 1993). The present procedure differs in what is edited, how strongly, and with what guarantee. The target is a state certified spurious by Eq. 8 rather than a state sampled by the dynamics, ε is not tuned but read off the block quantities through Eq. H.3, and the consequence is exact rather than average: Eq. H.2 predicts the entire post-intervention stable set, which the two cases above verify against exhaustive enumeration. The difference is also quantitative. Both cases confirm, as a computed rather than an empirical statement, that an unlearning step which is too strong damages what it does not target: at full strength $\varepsilon = 1$ leaves 6 and 16 additional stable block configurations, whereas inside the admissible interval the spurious pair disappears, no new stable configuration appears, and under the asynchronous dynamics every initial state now reaches a stored memory, the fraction rising from 69.9% to 100% at $M = 3$ and from 45.1% to 100% at $M = 4$ (Tables 4 and 5). The correction removes one certified attractor at a time; several deletions interact through Eq. H.2 and are left to future work.

I Computational Complexity and Algorithm

I.1 Computational Complexity Analysis

The cost of building the block description and certifying candidates is $\mathcal{O}(MN + P^2M)$, where:

1. Bit-level signature computation and block partitioning: $\mathcal{O}(MN)$, linear in the number of neurons and stored memories.
2. $\mathbf{\Gamma}$-matrix construction: $\mathcal{O}(P^2M)$, since each of the P^2 entries is an inner product of two M-dimensional signatures (Lemma F.5); when the M memory coordinates are compared in constant time, or the signature inner products are precomputed, this reads $\mathcal{O}(P^2)$, the form quoted in the main text.
3. Block stability constant calculation and inequality verification: $\mathcal{O}(P)$ given $\mathbf{\Gamma}$, or $\mathcal{O}(PM)$ for a single candidate evaluated directly from the signature matrix S, without forming $\mathbf{\Gamma}$ or $\mathbf{\Gamma}^{\boldsymbol{\sigma}}$ (Remark F.8).
4. Enumerate all candidates: $\mathcal{O}(2^P P)$.

Since the number of blocks satisfies $P \leq \min(2^{M-1}, N)$ (Appendix F.4), building the block description and certifying a candidate are far cheaper than the exponential $\mathcal{O}(2^N)$ of exhaustive dynamic simulation, and require no simulation of the dynamics. The complete spurious set, however, is obtained by enumerating the 2^P block configurations, which remains exponential in P in the worst case: the reduction shrinks the space to be explored, from the 2^N neuron-level states to those configurations, and is decisive whenever the memory set keeps P small, but it does not by itself make the complete list polynomial-time.

I.2 Algorithmic Characterization

We present a $\mathbf{\Gamma}$-matrix algorithm (via the Hamming-distance form of Lemma F.1) to check whether a symbolic centroid is a spurious attractor.

Input: Stored memories $\{\xi^1, \ldots, \xi^M\}$

Procedure:

1. Compute the signatures $s_p \in \{\pm 1\}^M$, partition the neurons into blocks, and record the minority size $t_p = \#\{m : s_p^m = -1\}$ of each block.
2. For each pair of blocks set $\Gamma_{pq} = M - 2\, d_H(s_p, s_q)$ (Lemma F.1); if the closed form $\Gamma_{pq} = M - 2(t_p + t_q - 2x_0)$ is used, $x_0 = |\mathcal{M}_p \cap \mathcal{M}_q|$ counts the minority indices shared by the two blocks.
3. Build block-size vector $\boldsymbol{L}$ and verify $\mathbf{\Gamma}^{\boldsymbol{\sigma}} \boldsymbol{L} \succeq M \mathbf{1}_P$.

Output: The symbolic centroid is a spurious attractor iff the matrix inequality holds and its block configuration does not correspond to a stored memory.