ATTRACTOR DEGENERACY IN HOPFIELD NEURAL NETWORKS: MEMORY EQUIVALENCE, ORBIT CLASSIFICATION, AND DYNAMICAL REUSE

Anonymous authors
Paper under double-blind review

ABSTRACT

Hopfield neural networks (HNNs) realize associative memory through dynamics induced by memories. Different memory matrices can nevertheless generate conjugate network dynamics, a phenomenon we call *attractor degeneracy*. We study the equivalence relation generated by row and column permutations and sign flips in binary HNNs. With synchronous updates and spin retention at zero field, equivalent matrices have update maps related by a signed coordinate relabeling. This bijection preserves trajectories, periodic attractors, basin sizes, and Hamming distances. We describe the equivalence classes as group orbits, derive an exact count among all 2^{MN} binary memory matrices with M memories and N neurons, and give an orbit-recognition method that recovers the row and column transformations. The count avoids enumerating the memory space, while a recovered transformation allows the dynamics of one representative to be transferred to any equivalent memory matrix. Exact complete-space and independently sampled candidate-bank experiments preserve the reported statistics while reducing repeated HNN evaluations beyond a column-only quotient.

1 INTRODUCTION

Hopfield neural networks (HNNs) provide a classical framework for associative memory, in which memories shape the dynamical landscape and govern how network states evolve toward memories or other attractors (Hopfield, 1982). Accordingly, much of the theoretical study of HNNs has focused on the properties of a given memory system, including storage capacity, stability, retrieval, and attractor structure. These analyses typically treat the stored memory matrix as the identity of the network (Abu-Mostafa & St-Jacques, 1985; Amit et al., 1985; McEliece et al., 1987; Kanter & Sompolinsky, 1987; Storkey, 1997; Baldi & Venkatesh, 1987; Gardner, 1988). A more fundamental question, however, remains: *which transformations of the memory matrix can preserve the dynamics they induce, while ensuring a consistent relabeling of the network states?*

This question arises because a memory matrix contains representational degrees of freedom that need not correspond to distinct dynamics. For example, reordering memories or reversing the sign of an entire pattern leaves the interactions unchanged. Likewise, permuting or sign flips neuron coordinates can change the representation of network states while preserving their dynamical relationships. The final state after training is such that A and B are in one-to-one correspondence between trajectories, periodic attractors and attractor basins in the original network and the transformed network.

This observation points to a form of redundancy that is fundamentally different from conventional redundancy in stored information: it is redundancy in the *representation of the dynamical system itself*. We refer to this redundancy across memory representations as *attractor degeneracy*: distinct memory matrices can induce conjugate dynamics under a bijective relabeling of network states. Here, degeneracy concerns multiple memory descriptions of the same dynamical structure, rather than duplicate attractors within a single network. This perspective naturally raises three questions: which memory transformations guarantee such dynamical equivalence, how much redundancy do they induce in the memory space, and how can equivalent memory representations be recognized in practice?

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.

These transformations partition the memory space into equivalence classes whose members have conjugate dynamics. For any statistic that is constant within each class, one evaluation on a representative suffices for the entire class. Weighting each representative's result by its class size reconstructs the corresponding aggregate statistics over all memory matrices. In a candidate bank, an equivalent matrix can instead reuse a previously computed score.

We study this problem systematically for classical binary HNNs with synchronous dynamics and spin retention at zero local field. Our classification concerns equivalence relations generated by permutations of rows and columns and by the inversion of signs; the fact that two elements belong to the same class indicates the existence of a dynamical conjugacy relation. Our contributions are summarized as follows:

- **Dynamical characterization of attractor degeneracy.** We formalize memory equivalence under row and column permutations and sign flips, and prove that equivalent memory matrices induce conjugate synchronous dynamics. The induced state relabeling maps trajectories, periodic attractors, and basins bijectively, while preserving basin sizes and Hamming distances.

- **Exact orbit classification and counting.** We establish a correspondence between memory equivalence classes and the orbits of a valid group action, rcharacterize the representatives of complete orbits, and deduce a precise formula for the number of equivalence classes without the need to enumerate the memory space.

- **Recognition and computational reuse.** We develop an exact orbit-recognition procedure based on anchor normalization and colored bipartite graph isomorphism that returns a verifiable transformation certificate. This enables dynamical quantities to be evaluated once per equivalence class, and experiments on both complete memory spaces and independently sampled candidate banks demonstrate substantial reductions in repeated HNN evaluations.

2 Related Work

Modern associative memory. Modern HNNs and dense associative memories study capacity and retrieval under modified energy functions or update rules (Ramsauer et al., 2021; Krotov & Hopfield, 2021; Demircigil et al., 2017; Bao et al., 2022; Bao & Zhao, 2025), with complementary analyses of sparsity, computational complexity, and expressivity (Hu et al., 2023; 2024; Li et al., 2024). Continuous-time and memristive implementations, together with studies of attractor structure and nonequilibrium dynamics, further underscore HNNs as a unifying framework (Santos et al., 2025; Niu et al., 2024; McAlister et al., 2025; Behera et al., 2025). Recent work broadens their dynamics, architectures, and theoretical interpretations: universal, dense, and Bayesian adaptive associative memories extend the expressive and probabilistic view of Hopfield retrieval (Millidge et al., 2022; Mimura et al., 2026; Wang et al., 2026a), while parameter-free Hopfield attention, attractor-like vision transformers, complex retrieval trajectories, and manifold-constrained models expand their dynamical and representational scope (Masumura & Taki, 2026; Xia & Jiang, 2026; Bogakovskaya et al., 2026; Wang et al., 2026b).

Symmetry and combinatorial classification. Gauge transformations in spin-glass theory identify redundancies in parameter descriptions (Nishimori, 2001); group actions also underlie invariant and equivariant neural representations (Cohen & Welling, 2016; Bronstein et al., 2021). For binary matrices, permutation-group methods and Pólya enumeration provide tools for counting classes under permutations and complement operations (Harrison, 1973; De Bruijn, 1964; de Bruijn, 1971; Fagbile, 2025). These are the combinatorial tools used here. To interpret a matrix orbit as a family of dynamically equivalent memories, one must additionally prove compatibility with the Hebbian weights and the chosen update rule, including its zero-field convention. We make this connection explicit and pair the orbit count with a recognition method that returns the actual transformation.

3 MEMORY EQUIVALENCE AND DYNAMICAL CONJUGACY

3.1 PRELIMINARIES

Fix $M, N \in \mathbb{Z}_{>0}$. For $d \in \mathbb{Z}_{>0}$, write $[d] := \{1, \dots, d\}$ and let S_d be the group of permutations of $[d]$. Consider a discrete HNN with N neurons and M memories, stored as the rows of the bipolar matrix

$$\Xi = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_M \end{pmatrix} = \begin{pmatrix} \Xi_{11} & \Xi_{12} & \cdots & \Xi_{1N} \\ \Xi_{21} & \Xi_{22} & \cdots & \Xi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \Xi_{M1} & \Xi_{M2} & \cdots & \Xi_{MN} \end{pmatrix} \in \{-1, 1\}^{M \times N}. \tag{1}$$

Here Ξ_{ij} is the entry in row i, column j, and $\xi_i = (\Xi_{i1}, \dots, \Xi_{iN}) \in \{-1, 1\}^N$ is the ith memory. For a labeled matrix Ξ_s, its entries are written $(\Xi_s)_{ij}$; the same convention applies to other matrices. Repeated or opposite rows are allowed. The network state is a row vector $V \in \Sigma := \{-1, 1\}^N$. The symbols $\mathbf{1}_d$ and I_d denote the all-ones row vector and identity matrix of size d; $\mathbf{1}$ and 0 have the dimensions of the surrounding expression. For $V, Z \in \Sigma$, the Hamming distance is

$$d_{\mathrm{H}}(V, Z) := \big|\{j \in [N] \mid V_j \neq Z_j\}\big|. \tag{2}$$

Matrix transformations and binary encoding. A signed permutation matrix has exactly one nonzero entry, equal to 1 or -1, in each row and column. We use $P \in \mathbb{R}^{M \times M}$ for row transformations and $Q \in \mathbb{R}^{N \times N}$ for column transformations; thus $P^{-1} = P^\top$ and $Q^{-1} = Q^\top$. For counting, write $\mathbb{F}_2 = \{0, 1\}$ with arithmetic modulo two and set

$$X := \frac{\mathbf{1} - \Xi}{2} \in \Omega := \mathbb{F}_2^{M \times N}, \qquad \Xi_{ij} = (-1)^{X_{ij}}. \tag{3}$$

This encoding is a bijection between bipolar and binary matrices. Braces denote sets, parentheses denote ordered tuples, and double braces denote multisets, in which repeated elements retain their multiplicities. Ranks and dimensions of binary matrices and spaces are taken over $\mathbb{F}_2$; those of bipolar matrices are over $\mathbb{R}$.

Weight matrix. Under the Hebbian learning rule, with self-couplings removed, the synaptic weight matrix associated with Ξ is

$$W_\Xi := \Xi^\top \Xi - M I_N = \sum_{i=1}^{M} \xi_i^\top \xi_i - M I_N, \tag{4}$$

where I_N denotes the $N \times N$ identity matrix. Thus W_Ξ is symmetric and has zero diagonal.

Synchronous update rule. Given a state $V \in \Sigma$, all neurons are updated simultaneously according to the local fields VW_Ξ. Element-wise, the state-transition map is

$$F_\Xi(V)_j := \begin{cases} 1, & (VW_\Xi)_j > 0, \\ -1, & (VW_\Xi)_j < 0, \\ V_j, & (VW_\Xi)_j = 0, \end{cases} \qquad j \in [N]. \tag{5}$$

The zero-field case specifies the tie convention used throughout: a neuron with zero local field retains its current spin. With this convention, $F_\Xi : \Sigma \to \Sigma$ is well defined for every memory matrix Ξ.

Dynamical system. Starting from $V^{(0)} \in \Sigma$, the synchronous HNN evolves by

$$V^{(t+1)} = F_\Xi\big(V^{(t)}\big), \qquad V^{(t)} = F_\Xi^t\big(V^{(0)}\big), \tag{6}$$

where $t \in \mathbb{Z}_{\geq 0}$, F_Ξ^t denotes the t-fold composition, and $F_\Xi^0 = \mathrm{id}_\Sigma$. We also use the standard energy

$$E_\Xi(V) := -\frac{1}{2} V W_\Xi V^\top, \tag{7}$$

Section 4 establishes how this energy transforms under the symmetries of the model. Notation for group actions and orbit certificates is introduced as it becomes necessary in Sections 4 and 5.

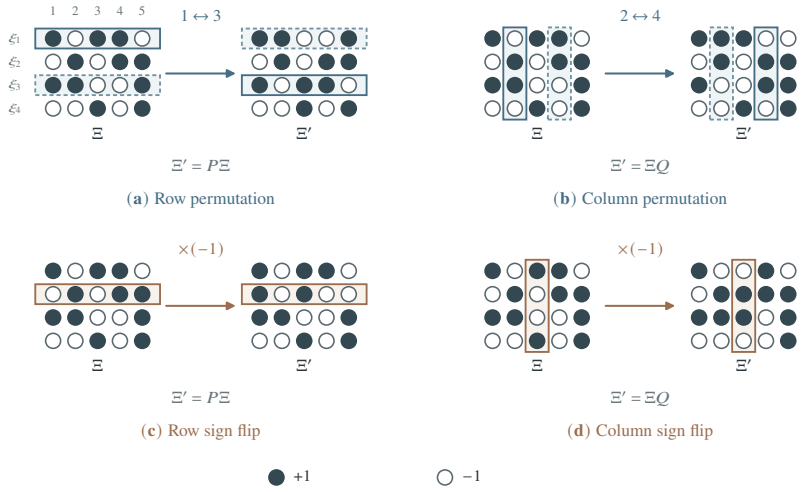


Figure 1: The four generators of memory equivalence, each applied to the same matrix. Filled and open disks denote $+1$ and -1. Blue outlines track whole rows or columns under permutation; ochre outlines mark sign flips at fixed positions. Row transformations multiply Ξ on the left, column transformations on the right.

3.2 Problem Formulation

The global dynamics of a HNN are fully determined by the synchronous state-transition map $F_\Xi : \Sigma \to \Sigma$. This map prescribes the next state for every configuration and therefore determines every trajectory, periodic attractor, and basin of attraction in the state space Σ, where $|\Sigma| = 2^N$. Here a periodic attractor is a directed cycle in the finite transition graph, and its basin is the set of states whose trajectories eventually enter that cycle. Distinct memory matrices can nevertheless induce transition graphs that become isomorphic after the states are relabeled. The resulting structural question is which natural transformations of a memory matrix guarantee preservation of the complete dynamical structure, and how much redundancy can be removed by identifying matrices related by those transformations? The relevant transformations arise from the intrinsic row and column symmetries of the Hebbian model, illustrated in Figure 1.

Definition 1 (Equivalence of memory matrices)**.** *Two memory matrices $\Xi_1, \Xi_2 \in \{-1, 1\}^{M\times N}$ are* equivalent*, denoted $\Xi_1 \sim \Xi_2$, if Ξ_2 can be obtained from Ξ_1 by a finite sequence, possibly empty, of the following operations: (i) permutation of rows of Ξ; (ii) permutation of columns of Ξ; (iii) multiplication of any row of Ξ by -1; and (iv) multiplication of any column of Ξ by -1.*

Equivalently, $\Xi_1 \sim \Xi_2$ if and only if there exist signed permutation matrices $P \in \mathbb{R}^{M\times M}$ and $Q \in \mathbb{R}^{N\times N}$ such that

$$\Xi_2 = P\Xi_1 Q. \tag{8}$$

The column transformation induces a state relabeling $\phi : \Sigma \to \Sigma$ given by $\phi(V) = VQ$. It permutes neuron indices and possibly reverses selected coordinates, while the row transformation changes only the ordering and signs of the stored memories.

Proposition 1 (Equivalence relation and classes)**.** *Definition 1 is equivalent to* (8)*. On the fixed space $\{-1, 1\}^{M\times N}$, the relation $\sim$ is reflexive, symmetric, and transitive. Its classes*

$$[\Xi] := \{\Xi' \in \{-1, 1\}^{M\times N} \mid \Xi \sim \Xi'\} \tag{9}$$

form a partition of that space.

Appendix A.1.1 proves the certificate equivalence and each relation axiom, and identifies these classes with the group orbits in Section 4.

Definition 2 (Dynamical conjugacy)**.** *Two update maps are conjugate if there exists a bijection $\phi : \Sigma \to \Sigma$ such that $F_{\Xi_2} \circ \phi = \phi \circ F_{\Xi_1}$.*

This is an equivalence relation on update maps (Appendix A.1.2). The results below prove that memory equivalence implies conjugacy, with $\phi(V) = VQ$; they do not assume the converse. The central question is therefore:

How can the equivalence classes generated by row and column permutations and sign flips be counted and recognized, and how can their structure compress the description of the memory space?

This question has three components:

(i) **Dynamical commutativity:** proving the exact relation $F_{\Xi_2} \circ \phi = \phi \circ F_{\Xi_1}$ for equivalent memory matrices;

(ii) **State-space invariance:** showing that ϕ is a Hamming isometry and a bijection of Σ, so trajectories, attractors, and basins correspond exactly;

(iii) **Orbit counting and recognition:** counting the algebraic symmetry orbits and constructing a test that recognizes orbit membership and recovers an explicit transformation certificate.

3.3 Theoretical Results

The main results connect compatibility of one-step dynamics with equivalence of the full state-transition graphs.

Theorem 1 (Dynamical Commutativity of Equivalent Matrices). *Let $\Xi_1 \sim \Xi_2$, so that $\Xi_2 = P\Xi_1 Q$ for signed permutation matrices $P \in \mathbb{R}^{M\times M}$ and $Q \in \mathbb{R}^{N\times N}$. Define $\phi : \Sigma \to \Sigma$ by $\phi(V) = VQ$. Then, for every $V \in \Sigma$,*

$$F_{\Xi_2}\big(\phi(V)\big) = \phi\big(F_{\Xi_1}(V)\big). \tag{10}$$

Indeed, $P^\top P = I_M$ gives $W_{\Xi_2} = Q^\top W_{\Xi_1} Q$ and $(VQ)W_{\Xi_2} = (VW_{\Xi_1})Q$. A signed coordinate permutation preserves all three branches of Eq. (5), including retention of the transformed spin at zero field, which yields Eq. (10). Appendix A.1.3 gives the coordinate proof. The following result describes the state relabeling itself.

Theorem 2 (State-Space Bijection and Hamming Isometry). *Under the hypotheses of Theorem 1, $\phi(V) = VQ$ is a bijection of Σ, with inverse $\phi^{-1}(Z) = ZQ^\top$. Moreover, for all $V, Z \in \Sigma$,*

$$d_{\mathrm{H}}\big(\phi(V), \phi(Z)\big) = d_{\mathrm{H}}(V, Z). \tag{11}$$

Consequently, $\phi(\Sigma) = \Sigma$ and $|\phi(\Sigma)| = |\Sigma| = 2^N$.

The proof appears in Appendix A.1.4. Together, Theorems 1 and 2 give conjugate transition graphs: ϕ carries finite trajectories, periodic attractors, and their basins to their counterparts. Orbit membership is therefore a sufficient, directly checkable certificate of conjugacy, although other equivalences may relate matrices in distinct algebraic classes. Sections 4 and 5 derive the exact class count and an orbit-recognition procedure that returns a transformation certificate.

4 Algebraic Classification of Memory Equivalence Classes

The four transformations in Definition 1 admit a compact algebraic description. This section identifies the effective group, removes the flip symmetry by normalization, counts the remaining permutation orbits, and relates those orbits to Hopfield dynamics. We use the binary encoding $X \in \Omega$ from Eq. (3). Throughout, matrices are ordered arrays; transposition is not allowed even when $M = N$.

4.1 Effective Transformation Group

In binary coordinates, sign flips add a row-constant or column-constant matrix, while permutations rearrange entries. Let $\mathcal{H} := S_M \times S_N$ be the direct product of the row and column permutation groups. For $h = (\sigma, \tau) \in \mathcal{H}$, define $(hX)_{ij} = X_{\sigma^{-1}(i),\tau^{-1}(j)}$. For $\epsilon \in \mathbb{F}_2^M$ and $\eta \in \mathbb{F}_2^N$, set

$$\delta(\epsilon,\eta)_{ij} = \epsilon_i + \eta_j, \qquad \mathcal{C} := \operatorname{im}\delta = \{\delta(\epsilon,\eta) \mid \epsilon \in \mathbb{F}_2^M,\ \eta \in \mathbb{F}_2^N\}. \tag{12}$$

The kernel of δ is $\{(0,0), (\mathbf{1}_M, \mathbf{1}_N)\}$, so $|\mathcal{C}| = 2^{M+N-1}$.

Theorem 3 (Effective group). *For $D \in \mathcal{C}$ and $h \in \mathcal{H}$, the four transformations induce the faithful action*

$$(D,h)\cdot X = D + hX, \qquad \mathcal{G} := \mathcal{C} \rtimes \mathcal{H}, \qquad |\mathcal{G}| = 2^{M+N-1}M!N!, \tag{13}$$

with multiplication $(D,h)(D',h') = (D + hD', hh')$.

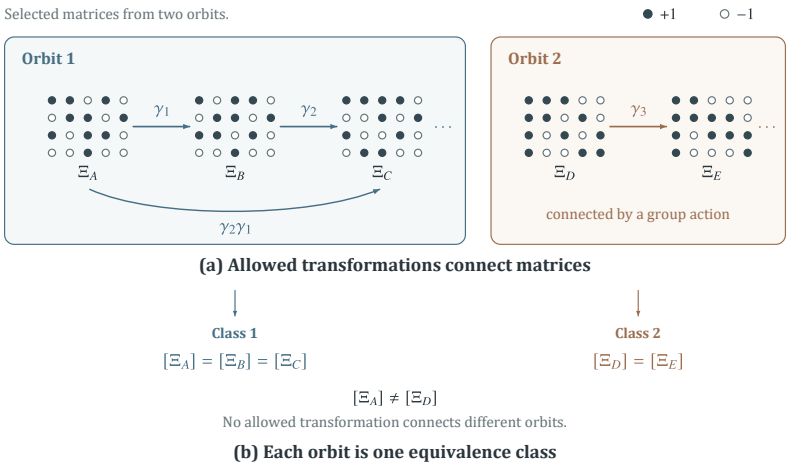


Figure 2: Group actions and the induced equivalence classes. Each disk array is a complete memory matrix; filled and open disks denote $+1$ and -1. In Orbit 1, γ_1 exchanges rows 1 and 3, and γ_2 flips column 2; applying them in that order maps Ξ_A to Ξ_C. In Orbit 2, γ_3 exchanges rows 1 and 2 and flips column 4. The lower panel identifies each orbit with one equivalence class. The two displayed orbits are distinct because their representatives have real ranks 4 and 3, which signed row and column permutations preserve. Only selected orbit members are shown.

The proof of Theorem 3 is given in Appendix A.2.1. Here $\rtimes$ is the semidirect product in which permutations act on flip masks by $D' \mapsto hD'$; its underlying set is $\mathcal{C} \times \mathcal{H}$. The factor of two comes from the kernel of the flip parameters, not from an assumption about stabilizers, which may be nontrivial. Flips can now be separated from permutations.

4.2 Normalization and Complete Orbit Representatives

Removing the flip symmetry before treating permutations assigns a unique normalized element to each flip class. Define

$$\pi(X)_{ij} = X_{ij} + X_{i1} + X_{1j} + X_{11}, \qquad U := \{Y \in \Omega \mid Y_{i1} = Y_{1j} = 0 \,\forall (i,j) \in [M] \times [N]\}. \tag{14}$$

Then $\pi(X) = \pi(X')$ exactly when $X + X' \in \mathcal{C}$; hence $\Omega/\mathcal{C} \cong U$ and $\dim U = (M-1)(N-1)$. A permutation is followed by re-normalization, so the induced action on U is $Y \mapsto \pi(hY)$. Here $\Omega/\mathcal{C}$ is the set of additive cosets $X + \mathcal{C}$; $U/\mathcal{H}$ below is the orbit set of this induced permutation action. Let $\mathcal{G}_X := \{g \in \mathcal{G} \mid g \cdot X = X\}$ be the stabilizer of X. Write $\mathcal{G} \cdot X := \{g \cdot X \mid g \in \mathcal{G}\}$ for its orbit and $\Omega/\mathcal{G} := \{\mathcal{G} \cdot X \mid X \in \Omega\}$ for the orbit set.

Theorem 4 (Orbits and complete representatives)**.** *The orbits partition Ω, and*

$$|\mathcal{G} \cdot X| = \frac{|\mathcal{G}|}{|\mathcal{G}_X|}, \qquad \Omega/\mathcal{G} \cong U/\mathcal{H}. \tag{15}$$

For any fixed total order on U,

$$\min_{h \in \mathcal{H}} \pi(hX) \tag{16}$$

is a complete representative: two matrices lie in the same orbit exactly when these minima agree.

The proof of Theorem 4 is given in Appendix A.2.3. Figure 2 illustrates the relation between successive transformations, orbits, and equivalence classes. Orbit–stabilizer determines the class size from its internal symmetries, whereas the minimum in Eq. (16) uniquely labels the class. Direct minimization examines at most $M!N!$ candidates. The count below does not require the representatives to be constructed.

4.3 Exact Number of Algebraic Classes

We apply Burnside's lemma to the induced permutation action on U (De Bruijn, 1964). Let $\lambda = (a_1, \ldots, a_r) \vdash M$ and $\mu = (b_1, \ldots, b_s) \vdash N$ be the integer partitions formed by the row- and

column-permutation cycle lengths. Here $\vdash$ means "is a partition of," while r and s are the respective numbers of cycles. Put

$$d_{uv} := \gcd(a_u, b_v), \qquad K(\lambda, \mu) := \sum_{u=1}^{r} \sum_{v=1}^{s} d_{uv}. \tag{17}$$

Define the $rs \times (r+s)$ binary matrix $T(\lambda, \mu)$ by

$$T_{(u,v),u} = \frac{b_v}{d_{uv}} \pmod 2, \qquad T_{(u,v),r+v} = \frac{a_u}{d_{uv}} \pmod 2, \tag{18}$$

with all other entries zero. We write K and T for these quantities at each pair (λ, μ). For $\nu \in \{\lambda, \mu\}$, let $m_k(\nu)$ be the number of parts of ν equal to k, and set $z_\nu = \prod_{k \geq 1} k^{m_k(\nu)} m_k(\nu)!$.

Theorem 5 (Exact orbit count)**.** *For all $M, N \geq 1$, the number $A(M, N) := |\Omega/\mathcal{G}|$ is*

$$A(M, N) = \sum_{\lambda \vdash M} \sum_{\mu \vdash N} \frac{2^{K - \operatorname{rank} T}}{z_\lambda z_\mu}, \tag{19}$$

The term $2^{K - \operatorname{rank} T}$ counts fixed flip classes, while z_λ and z_μ supply the cycle-type weights in Burnside's average. Appendix A.2.4 proves the formula, Appendix A.2.5 derives $\dim \ker T(\lambda, \mu)$, and Appendix B.1 gives an exact integer algorithm.

Define the candidate compression ratio $R(M, N) := \frac{A(M,N)}{2^{MN}}$. Its reciprocal $\frac{1}{R(M,N)} = \frac{2^{MN}}{A(M,N)}$ is the mean orbit size, hence the mean reduction when an orbit-invariant quantity is evaluated once per representative. Appendix A.2 plots $\log_{10}(1/R)$; a value of 20 means that one representative replaces 10^{20} matrices on average. At $M = N = 25$, this factor is about 1.35×10^{65}. Representative generation has a separate cost.

4.4 Connection to Hebbian Dynamics

For bipolar memories, consider the definitions of W_Ξ, F_Ξ, and E_Ξ from Section 3. Row operations vanish from the Hebbian weights, whereas column operations relabel memory coordinates and network states in the same way.

Theorem 6 (Induced dynamical equivalence)**.** *Let Ξ, Ξ' be the bipolar matrices encoded by $X, X' \in \Omega$, respectively. If X and X' lie in one $\mathcal{G}$-orbit, there exist signed permutation matrices P, Q with $\Xi' = P\Xi Q$. For every $V \in \Sigma$,*

$$\begin{aligned} W_{\Xi'} &= Q^\top W_\Xi Q, \\ F_{\Xi'}(VQ) &= F_\Xi(V) Q, \\ E_{\Xi'}(VQ) &= E_\Xi(V). \end{aligned} \tag{20}$$

The proof of Theorem 6 is given in Appendix A.2.7. Each algebraic orbit therefore supplies an explicit signed state relabeling that conjugates the corresponding Hebbian dynamics.

4.5 Discussion

Orbit membership certifies dynamical conjugacy: transition graphs, attractors, basins and energies under Eq. (20). Distinct algebraic orbits may still be dynamically equivalent through other relations. Thus $A(M, N)$ counts descriptions after removing the proven symmetries, but does not reduce the 2^N states of one representative. Section 5 gives a decision procedure and transformation certificate for matrices in the same orbit.

5 Orbit Recognition and Transformation Certificates

To use the orbit partition computationally, one must decide whether two memory matrices lie in the same class and recover the transformation relating them. Let $\Xi_1, \Xi_2 \in \{-1, 1\}^{M \times N}$. We seek

permutations $p \in S_M$, $q \in S_N$ and sign vectors $\rho = (\rho_i)_{i\in[M]} \in \{-1,1\}^M$ and $\kappa = (\kappa_j)_{j\in[N]} \in \{-1,1\}^N$ satisfying, for every $(i,j) \in [M] \times [N]$,

$$(\Xi_2)_{ij} = \rho_i (\Xi_1)_{p(i),q(j)} \kappa_j. \tag{21}$$

Thus $p(i)$ and $q(j)$ identify the source row and column used at target position (i,j). Figure 5 in Appendix A.3 summarizes the recognition workflow.

Fast rejection. Eq. (21) preserves the two multisets

$$\left\{\left\{ \left| \sum_{j=1}^{N} \Xi_{ij}\Xi_{kj} \right| \,\middle|\, 1 \le i < k \le M \right\}\right\}, \qquad \left\{\left\{ \left| \sum_{i=1}^{M} \Xi_{ij}\Xi_{ij'} \right| \,\middle|\, 1 \le j < j' \le N \right\}\right\}. \tag{22}$$

Absolute values remove the signs, and permutations merely reorder the pairs. Unequal multisets therefore certify distinct orbits. Equality is only a necessary condition and is followed by the exact test below.

Anchor reduction and exact recognition. For an anchor $(a,b) \in [M] \times [N]$, define

$$\alpha_{a,b}(\Xi)_{ij} := \Xi_{ab}\Xi_{ib}\Xi_{aj}\Xi_{ij}. \tag{23}$$

This valid combination of row and column sign transformations makes row a and column b positive. It is the bipolar counterpart of the binary reduction in Section 4: if $X = \frac{\mathbf{1}-\Xi}{2}$, then $\alpha_{1,1}(\Xi)_{ij} = (-1)^{\pi(X)_{ij}}$. Delete the positive anchor row and column, represent the remaining rows and columns by two vertex colors, and join them exactly at negative entries. Denote the resulting colored bipartite graph by $\Gamma_{a,b}(\Xi)$; isolated vertices are retained.

Theorem 7 (Exact orbit recognition)**.** *The binary encodings of Ξ_1 and Ξ_2 lie in the same $\mathcal{G}$-orbit if and only if, for some target anchor (a,b), the graphs $\Gamma_{1,1}(\Xi_1)$ and $\Gamma_{a,b}(\Xi_2)$ are isomorphic with row and column colors preserved. Equivalently, there exist permutations p, q with $p(a) = q(b) = 1$ such that*

$$\alpha_{a,b}(\Xi_2)_{ij} = \alpha_{1,1}(\Xi_1)_{p(i),q(j)}. \tag{24}$$

The proof of Theorem 7 is given in Appendix A.3.1. A single source anchor therefore suffices, together with at most MN target anchors. Preserving the colors rules out transposition, including for square memories. Retaining every vertex, including isolated ones, preserves multiplicities and covers degenerate cases such as repeated or opposite rows and columns.

Recovering the transformation. Inverting the graph's source-to-target correspondence gives p, q. The signs are then set to

$$\rho_i = (\Xi_2)_{ab}(\Xi_2)_{ib}(\Xi_1)_{11}(\Xi_1)_{p(i),1}, \tag{25}$$

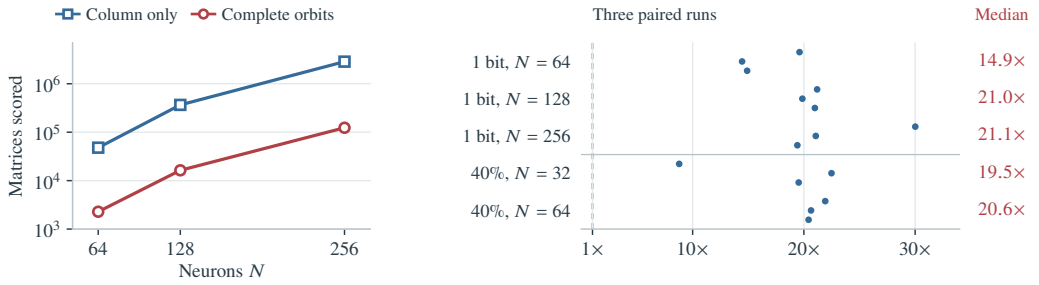
$$\kappa_j = (\Xi_1)_{1,q(j)}(\Xi_2)_{a,j}. \tag{26}$$

These choices satisfy Eq. (21). Setting $P_{i,p(i)} = \rho_i$ and $Q_{q(j),j} = \kappa_j$, with other entries zero, gives $\Xi_2 = P\Xi_1 Q$ and $\phi(V)_j = \kappa_j V_{q(j)}$. The implementation verifies both permutations, all signs, and every reconstructed entry. Appendices A.3 and B give the proof, complete search, certificate recovery, and an example. Correlation preprocessing costs $O(M^2N + MN^2)$; the exact graph search is finite, with no claimed polynomial worst-case guarantee.

6 Computational Experiments

All experiments used Python 3.12 on one Intel Core i7-12650H CPU with 16 GB RAM, one process, one numerical-library thread, and no GPU. The column-only method merges matrices related by signed column permutations; the complete method also merges signed row permutations. Both call the same scorer once per representative, and timings include generation, scoring, exact weighting, and aggregation. We report the orbit-invariant statistics u, the number of fixed stored rows, and $s_{r,\ell}$, the number of radius-r state–row pairs that reach a fixed row after ℓ updates. Their definitions and invariance proof appear in Appendix A.4.

Exact complete-space analysis. For $M = 3$, multiplicities of the four projective column types $\{c, -c\}$ generate complete-orbit representatives directly. We evaluate $(r, \ell) = (1, 1)$ and $(\lfloor 0.40N \rfloor, 8)$. Weighting every result by its orbit size reconstructs all 2^{3N} matrices exactly; the construction and weights are given in Appendix C.



(a) Network evaluations **(b) End-to-end speedup**

Figure 3: Complete-space analysis for $M = 3$. (a) Matrices scored for $s_{1,1}$ using column-only classes (blue) or complete orbits (red). (b) End-to-end speedup, defined as column-only time divided by complete-orbit time. Blue dots show three paired runs per row; red values give their median. The dashed line marks equal time. The 40% task flips 40% of the neurons, rounded down. Both methods recover the same orbit-weighted histogram.

One-bit median speedups are 14.90, 20.99, and 21.07 at $N = 64, 128, 256$. At $N = 256$, $123{,}464$ complete orbits replace $2{,}862{,}209$ column classes over all 2^{768} matrices. High-noise medians are 19.53 at $N = 32$ and 20.64 at $N = 64$; all aggregates agree exactly.

Independent candidate banks. Ten independent banks of $1{,}000$ matrices at each of $N = 64, 128$ compare exact-matrix, column-only, and complete-orbit caches without planted copies. Relative to column-only caching, the complete cache is faster by median factors of 4.48 at $N = 64$ (range 4.06–4.75; ten of ten banks) and 2.83 at $N = 128$ (range 0.85–3.21; nine of ten). All $20{,}000$ score pairs agree. Appendix C gives all timings, $M = 4$ results, and validation checks.

7 Conclusion

This work formalizes attractor degeneracy in binary HNNs through equivalence classes generated by row and column permutations and sign flips. Equivalent memories induce conjugate synchronous dynamics under an explicit state relabeling, preserving trajectories, attractors, basins, Hamming distance, and energy. The effective action gives an exact orbit count, while anchor normalization and colored graph isomorphism yield verifiable recognition certificates. Complete-space and independent-bank experiments show that one evaluation per complete orbit recovers the same integer statistics with fewer evaluations and lower wall time than a column-only quotient. The classification therefore describes memory redundancy and avoids repeated dynamical analysis of equivalent candidates.

Limitations and future work. Our analysis focuses on binary HNNs under synchronous updates with spin retention at zero field. The classification is exact for the four specified transformations, rather than a characterization of all possible dynamical conjugacies. Although the orbit count is exact, efficient representative generation becomes challenging as the number of stored memories grows. Future work will focus on scalable representative generation and orbit recognition to reduce repeated dynamical evaluations over larger candidate spaces. We will also investigate which memory transformations preserve dynamical equivalence under alternative learning rules and in modern Hopfield networks.

AI USE STATEMENT

Generative AI tools assisted with theoretical formulations and proofs, scientific figure generation, and English editing. The research motivation and approach were primarily driven by authors. The authors verified the mathematical claims and proofs. The authors reviewed the final text and take responsibility for its content.

REPRODUCIBILITY STATEMENT

The supplementary material contains the complete source code, including (i) the exact counting algorithm and (ii) the fast algorithm for deciding whether two memory matrices lie in the same orbit. Appendix A and B provides the pseudocode, correctness arguments, and complexity analysis for these algorithms. We take responsibility for the correctness of the released code and the reported results.

ACKNOWLEDGMENTS

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

REFERENCES

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

Daniel J Amit, Hanoch Gutfreund, and Haim Sompolinsky. Storing infinite numbers of patterns in a spin-glass model of neural networks. *Physical Review Letters*, 55(14):1530, 1985.

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.

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.

Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pp. 2990–2999. PMLR, 2016.

Nicolaas G de Bruijn. A survey of generalizations of pólya's enumeration theorem. *Nieuw Archief voor Wiskunde*, 3(2):89–112, 1971.

Nicolaas Govert De Bruijn. Polya's theory of counting. In *Applied combinatorical mathematics*, pp. 144–184. 1964.

Mete Demircigil, Judith Heusel, Matthias Löwe, Sven Upgang, and Franck Vermet. On a model of associative memory with huge storage capacity. *Journal of Statistical Physics*, 168(2):288–299, 2017.

Damilare Fagbile. Pólya's enumeration theorem. 2025.

Elizabeth Gardner. The space of interactions in neural network models. *Journal of physics A: Mathematical and general*, 21(1):257, 1988.

MA Harrison. On the number of classes of binary matrices. *IEEE Transactions on Computers*, 22 (12):1048–1052, 1973.

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.

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*, 2024.

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

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

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.

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.

Brendan D. McKay and Adolfo Piperno. Practical graph isomorphism, II. *Journal of Symbolic Computation*, 60:94–112, 2014. doi: 10.1016/j.jsc.2013.09.003.

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.

Hidetoshi Nishimori. *Statistical physics of spin glasses and information processing: an introduction*. Number 111. Clarendon Press, 2001.

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.

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.

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

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.

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.

APPENDIX: NOTATION

All memory matrices have M rows and N columns. States are row vectors. Bipolar products are over $\mathbb{R}$; binary sums are over $\mathbb{F}_2$. The table lists the notation shared across sections. Auxiliary variables needed only within a proof or an algorithm are defined there.

Table 1: Notation used throughout the paper.

Symbol	Meaning
$M, N;\ [d];\ S_d$	Numbers of memories and neurons; $[d] = \{1, \dots, d\}$; permutations of $[d]$.
$\Xi;\ \xi_i;\ \Xi_{ij}$	Memory matrix in $\{-1, 1\}^{M \times N}$; its ith row $\xi_i = (\Xi_{i1}, \dots, \Xi_{iN})$; its row-$i$, column-$j$ entry. For a labeled matrix Ξ_s, write $(\Xi_s)_{ij}$.
$\Sigma;\ V^{(t)}$	State space $\Sigma = \{-1, 1\}^N$; state after t synchronous updates.
$W_\Xi,\ F_\Xi,\ E_\Xi$	Hebbian weights $W_\Xi = \Xi^\top \Xi - M I_N$; update map with spin retention at zero field; energy $E_\Xi(V) = -\frac{1}{2} V W_\Xi V^\top$. See (5) for the update rule.
$P, Q;\ \phi$	Signed row and column permutation matrices in $\Xi_2 = P \Xi_1 Q$; state map $\phi(V) = VQ$. Their sizes are $M \times M$ and $N \times N$.
$p, q;\ \rho, \kappa$	Target-to-source row and column permutations and their signs: $P_{i,p(i)} = \rho_i$, $Q_{q(j),j} = \kappa_j$, with $\rho_i, \kappa_j \in \{-1, 1\}$.
$\sim;\ [\Xi]$	Memory equivalence under the four transformations; $[\Xi] = \{\Xi' \in \{-1, 1\}^{M \times N} \mid \Xi \sim \Xi'\}$.
$\Omega;\ X$	Binary matrix space $\Omega = \mathbb{F}_2^{M \times N}$, where $\mathbb{F}_2 = \{0, 1\}$ with arithmetic modulo 2; encoding $X_{ij} = \frac{1 - \Xi_{ij}}{2}$.
$\mathcal{C}, \mathcal{H}, \mathcal{G}$	Additive group of masks $D_{ij} = \epsilon_i + \eta_j$; permutation group $\mathcal{H} = S_M \times S_N$; effective group $\mathcal{G} = \mathcal{C} \rtimes \mathcal{H}$. Its action is $(D, h) \cdot X = D + hX$.
$\times;\ \rtimes$	$\times$ denotes a Cartesian product; with the componentwise group law it is a direct product, as in $\mathcal{H} = S_M \times S_N$. For matrices it separates row and column dimensions. The semidirect product $\rtimes$ has underlying set $\mathcal{C} \times \mathcal{H}$ but the law $(D, h)(D', h') = (D + hD', hh')$, where hD' permutes the rows and columns of the mask D'.
$\oplus;\ \cdot;\ \circ;\ \cong$	$U \oplus \mathcal{C}$ is an internal direct sum (unique decomposition); $g \cdot X$ is the group action; $f \circ g$ applies g then f; $\cong$ denotes a structure-preserving isomorphism appropriate to its objects.
$\Omega/\mathcal{G};\ \mid$	The set of $\mathcal{G}$-orbits in Ω (not a numerical quotient); $\mid$ means "such that" in set-builder notation.
$\pi;\ U$	Normalization $\pi(X)_{ij} = X_{ij} + X_{i1} + X_{1j} + X_{11}$; space $U \subseteq \Omega$ with first row and first column zero.
$\mathcal{G} \cdot X;\ \mathcal{G}_X$	Orbit $\{g \cdot X \mid g \in \mathcal{G}\}$; stabilizer $\{g \in \mathcal{G} \mid g \cdot X = X\}$.
$A(M, N);\ R(M, N)$	Number of orbits $\lvert\Omega/\mathcal{G}\rvert$ and candidate-count compression ratio $R(M, N) = \frac{A(M,N)}{2^{MN}}$ (compressed-to-original), respectively.
$\lambda, \mu;\ z_\lambda, z_\mu$	Cycle types $(a_1, \dots, a_r) \vdash M$ and $(b_1, \dots, b_s) \vdash N$; $\vdash$ denotes an integer partition. For multiplicities m_k, $z_\lambda = \prod_{k \geq 1} k^{m_k} m_k!$; the same convention defines z_μ.
$d_{uv}, K;\ T$	$d_{uv} = \gcd(a_u, b_v)$ (greatest common divisor) and $K = \sum_{u,v} d_{uv}$; the binary compatibility matrix $T = T(\lambda, \mu)$ defined in (18).
$\alpha_{a,b};\ \Gamma_{a,b}$	Anchor normalization $\alpha_{a,b}(\Xi)_{ij} = \Xi_{ab} \Xi_{ib} \Xi_{aj} \Xi_{ij}$; its bipartite graph after deleting anchor row a and column b. Rows and columns have different colors; negative entries are edges.
d_{H}	Hamming distance on bipolar row vectors.
$u(\Xi);\ s_{r,\ell}(\Xi)$	Number of fixed stored rows; number of state–row pairs at Hamming radius r that return to the fixed stored row after exactly ℓ synchronous updates. Repeated stored rows retain their multiplicity.

We write sets as $\{x \in S \mid P(x)\}$ and multisets as $\{\{\cdot\}\}$, retaining repetitions. The identity matrix is I_d; $\mathbf{1}$ and 0 have the dimensions specified by their context. The symbol $\perp$ denotes a failed match.

APPENDIX CONTENTS

The appendices collect proofs first, then the counting and recognition algorithms, followed by additional experimental details.

A Proofs **15**

A.1 Proofs of Memory Equivalence and Dynamical Conjugacy 15

A.2 Effective Group Action and Exact Orbit Counting 17

A.3 Exact Orbit Recognition 23

A.4 Invariance of the Named-Memory Recovery Score 25

B Algorithms for Orbit Counting and Recognition **26**

B.1 Exact integer evaluation of the orbit count 26

B.2 Exact colored graph search 27

B.3 Recognition and certificate verification 27

B.4 A four-by-five example with all four operations 28

B.5 Computational cost 29

C Additional Experimental Details **30**

C.1 Experimental protocol 30

C.2 Complete-space timings 30

C.3 Independent candidate banks 30

C.4 Validation and scope 31

A PROOFS

A.1 PROOFS OF MEMORY EQUIVALENCE AND DYNAMICAL CONJUGACY

A.1.1 MEMORY EQUIVALENCE AND ITS CLASSES

Fix $M, N \geq 1$. All memory matrices below belong to $\{-1, 1\}^{M\times N}$; equivalence is defined only between matrices of the same size.

Proof of Proposition 1. Each allowed row operation is left multiplication by a signed permutation matrix, and each column operation is right multiplication by one. Left and right multiplication commute with each other, and products of signed permutation matrices are again signed permutation matrices. Every finite sequence therefore has the form

$$\Xi_2 = P\Xi_1 Q, \qquad P \in \mathbb{R}^{M\times M}, \quad Q \in \mathbb{R}^{N\times N}. \tag{A.1}$$

The empty sequence gives $P = I_M$ and $Q = I_N$. Conversely, write the nonzero entries of P, Q as

$$P_{i,p(i)} = \rho_i, \qquad Q_{q(j),j} = \kappa_j, \qquad p \in S_M,\ q \in S_N,\ \rho_i, \kappa_j \in \{-1, 1\}. \tag{A.2}$$

Here p, q map target indices to source indices. Then

$$(P\Xi_1 Q)_{ij} = \rho_i (\Xi_1)_{p(i),q(j)} \kappa_j. \tag{A.3}$$

This expression permutes the rows and columns and flips the rows and columns whose signs are negative. Thus every certificate in (A.1) is realized by the allowed operations.

Reflexivity follows from the identity matrices. Orthogonality of P, Q gives symmetry, and closure under products gives transitivity:

$$\begin{aligned} \Xi_2 = P\Xi_1 Q \quad &\Longrightarrow \quad \Xi_1 = P^\top \Xi_2 Q^\top, \\ \Xi_2 = P\Xi_1 Q, \quad \Xi_3 = P'\Xi_2 Q' \quad &\Longrightarrow \quad \Xi_3 = (P'P)\Xi_1 (QQ'). \end{aligned} \tag{A.4}$$

Hence $\sim$ is an equivalence relation. Each matrix belongs to its own class, so the classes cover the whole space. If a matrix belongs to both $[\Xi_1]$ and $[\Xi_2]$, symmetry and transitivity give $\Xi_1 \sim \Xi_2$; every member of either class then belongs to the other. Consequently,

$$[\Xi_1] \cap [\Xi_2] \neq \varnothing \quad \Longleftrightarrow \quad \Xi_1 \sim \Xi_2 \quad \Longleftrightarrow \quad [\Xi_1] = [\Xi_2], \tag{A.5}$$

which proves that distinct classes are disjoint. □

Lemma 1 (Correspondence with binary group orbits)**.** *Let* $(X_s)_{ij} = \frac{1-(\Xi_s)_{ij}}{2}$ *for* $s \in \{1, 2\}$. *For the action* (13)*,*

$$\Xi_1 \sim \Xi_2 \quad \Longleftrightarrow \quad X_2 \in \mathcal{G} \cdot X_1. \tag{A.6}$$

Binary encoding therefore identifies memory classes with $\mathcal{G}$-orbits.

Proof. Write $\rho_i = (-1)^{\epsilon_i}$ and $\kappa_j = (-1)^{\eta_j}$, with $\epsilon_i, \eta_j \in \mathbb{F}_2$. Entrywise encoding turns (A.3) into

$$(X_2)_{ij} = \epsilon_i + (X_1)_{p(i),q(j)} + \eta_j \quad \text{in } \mathbb{F}_2. \tag{A.7}$$

With $D = \delta(\epsilon, \eta)$ and $h = (p^{-1}, q^{-1})$, this is exactly $X_2 = D + hX_1$. Conversely, any $D \in \mathcal{C}$ has this form for some ϵ, η, and any $h \in S_M \times S_N$ can be written as (p^{-1}, q^{-1}). Decoding recovers the signed certificate above. Since encoding is bijective, the correspondence holds for whole classes. □

A.1.2 DYNAMICAL CONJUGACY AND MEMORY EQUIVALENCE

Dynamical conjugacy, defined in Definition 2, allows any bijection $\phi : \Sigma \to \Sigma$ satisfying

$$F_{\Xi_2} \circ \phi = \phi \circ F_{\Xi_1}. \tag{A.8}$$

The identity map gives reflexivity. Symmetry follows by inversion:

$$F_{\Xi_2} \circ \phi = \phi \circ F_{\Xi_1} \quad \Longrightarrow \quad F_{\Xi_1} \circ \phi^{-1} = \phi^{-1} \circ F_{\Xi_2}. \tag{A.9}$$

If ϕ conjugates F_{Ξ_1} to F_{Ξ_2} and ψ conjugates F_{Ξ_2} to F_{Ξ_3}, then

$$\begin{aligned} F_{\Xi_3} \circ (\psi \circ \phi) &= \psi \circ F_{\Xi_2} \circ \phi \\ &= (\psi \circ \phi) \circ F_{\Xi_1}. \end{aligned} \tag{A.10}$$

The inverse and composite maps are bijective, proving that dynamical conjugacy is an equivalence relation.

The proofs below show that $\Xi_2 = P\Xi_1 Q$ supplies the particular conjugacy $\phi(V) = VQ$. Thus memory equivalence implies dynamical conjugacy; the converse is not asserted. Different memory classes may yield conjugate transition maps. Energy preservation under signed coordinate changes is proved in (20); it is not part of the definition of an arbitrary dynamical conjugacy.

A.1.3 Proof of theorem 1

Proof. Let $\Xi_2 = P\Xi_1 Q$, where P, Q are signed permutation matrices. Their orthogonality gives

$$\begin{aligned} W_{\Xi_2} &= Q^\top \Xi_1^\top P^\top P \Xi_1 Q - M I_N \\ &= Q^\top (\Xi_1^\top \Xi_1 - M I_N) Q = Q^\top W_{\Xi_1} Q. \end{aligned} \tag{A.11}$$

This calculation covers all four elementary operations and every finite composition. Row transformations disappear from the Gram matrix; column transformations act on the neuron coordinates.

To check the update rule, use the column convention $Q_{q(j),j} = \kappa_j$ from (A.2). For each $V \in \Sigma$,

$$(VQ)_j = \kappa_j V_{q(j)}, \qquad (VQ)W_{\Xi_2} = VW_{\Xi_1} Q. \tag{A.12}$$

In particular, $VQ \in \Sigma$. Fix $j \in [N]$ and put $u = (VW_{\Xi_1})_{q(j)}$. The transformed field is $\kappa_j u$, so the update in (5) becomes

$$F_{\Xi_2}(VQ)_j = \begin{cases} \kappa_j, & u > 0, \\ -\kappa_j, & u < 0, \\ \kappa_j V_{q(j)}, & u = 0 \end{cases} = \kappa_j F_{\Xi_1}(V)_{q(j)} = \big(F_{\Xi_1}(V)Q\big)_j. \tag{A.13}$$

At zero field, both sides retain the transformed spin $\kappa_j V_{q(j)}$. Since the identity holds in every coordinate, setting $\phi(V) = VQ$ gives

$$F_{\Xi_2}(VQ) = F_{\Xi_1}(V)Q, \qquad F_{\Xi_2} \circ \phi = \phi \circ F_{\Xi_1}. \tag{A.14}$$

□

A.1.4 Proof of theorem 2

Proof. A signed permutation preserves Σ, and so does its transpose. For $V, Z \in \Sigma$, orthogonality yields

$$(VQ)Q^\top = V, \qquad (ZQ^\top)Q = Z. \tag{A.15}$$

Thus $\phi^{-1}(Z) = ZQ^\top$ is an inverse on Σ, proving both injectivity and surjectivity. Consequently,

$$\phi(\Sigma) = \Sigma, \qquad |\phi(\Sigma)| = |\Sigma| = 2^N. \tag{A.16}$$

The same coordinate formula also proves preservation of Hamming distance. Multiplication by κ_j cannot change whether two coordinates differ, and q visits every coordinate once. Hence

$$\begin{aligned} d_{\mathrm{H}}(VQ, ZQ) &= \big|\{j \in [N] \mid \kappa_j V_{q(j)} \neq \kappa_j Z_{q(j)}\}\big| \\ &= \big|\{j \in [N] \mid V_{q(j)} \neq Z_{q(j)}\}\big| = d_{\mathrm{H}}(V, Z). \end{aligned} \tag{A.17}$$

□

A.1.5 TRAJECTORIES, PERIODIC ORBITS, AND BASINS

The one-step identity extends to every $t \in \mathbb{Z}_{\geq 0}$:

$$F^t_{\Xi_2}\big(\phi(V)\big) = \phi\big(F^t_{\Xi_1}(V)\big). \tag{A.18}$$

Indeed, it holds at $t = 0$. If it holds at t, then

$$\begin{aligned} F^{t+1}_{\Xi_2}\big(\phi(V)\big) &= F_{\Xi_2}\big(\phi\big(F^t_{\Xi_1}(V)\big)\big) \\ &= \phi\big(F^{t+1}_{\Xi_1}(V)\big), \end{aligned} \tag{A.19}$$

where the second equality uses (A.14).

Let $\Lambda \subseteq \Sigma$ be a periodic orbit of F_{Ξ_1}. Its image $\phi(\Lambda) = \{\phi(V) \mid V \in \Lambda\}$ is an orbit of F_{Ξ_2} with the same least period: for any $V \in \Lambda$ and integer $t \geq 1$, injectivity of ϕ gives

$$F^t_{\Xi_2}\big(\phi(V)\big) = \phi(V) \quad \Longleftrightarrow \quad F^t_{\Xi_1}(V) = V. \tag{A.20}$$

This includes fixed points as the period-one case. Define the basin of a periodic orbit by

$$B_\Xi(\Lambda) := \{V \in \Sigma \mid \exists\, t \in \mathbb{Z}_{\geq 0} : F^t_\Xi(V) \in \Lambda\}. \tag{A.21}$$

For every $V \in \Sigma$, (A.18) gives

$$\begin{aligned} V \in B_{\Xi_1}(\Lambda) \quad &\Longleftrightarrow \quad \exists\, t \geq 0 :\ F^t_{\Xi_1}(V) \in \Lambda \\ &\Longleftrightarrow \quad \exists\, t \geq 0 :\ F^t_{\Xi_2}(\phi(V)) \in \phi(\Lambda) \\ &\Longleftrightarrow \quad \phi(V) \in B_{\Xi_2}(\phi(\Lambda)). \end{aligned} \tag{A.22}$$

Surjectivity of ϕ therefore yields

$$\phi\big(B_{\Xi_1}(\Lambda)\big) = B_{\Xi_2}\big(\phi(\Lambda)\big). \tag{A.23}$$

The restriction of ϕ is a bijection between these basins, so their cardinalities agree. No convergence to a fixed point is assumed.

A.2 EFFECTIVE GROUP ACTION AND EXACT ORBIT COUNTING

Throughout this appendix, $M, N \geq 1$ are fixed. All binary sums are over $\mathbb{F}_2$, and $\Xi_{ij} = (-1)^{X_{ij}}$. Repeated and opposite memory rows are allowed.

A.2.1 THE FLIP KERNEL AND THE EFFECTIVE SEMIDIRECT PRODUCT

Recall that $\Omega = \mathbb{F}_2^{M \times N}$ and

$$\delta(\epsilon, \eta)_{ij} = \epsilon_i + \eta_j, \qquad (\epsilon, \eta) \in \mathbb{F}_2^M \times \mathbb{F}_2^N, \qquad \mathcal{C} = \operatorname{im} \delta. \tag{A.24}$$

Thus $\mathcal{C}$ contains exactly the masks generated by row and column flips. The redundancy in the flip parameters has a simple form.

Lemma 2 (Kernel of the flip parameters)**.** *The kernel of δ is $\{(0,0), (\mathbf{1}_M, \mathbf{1}_N)\}$. Consequently, $\dim \mathcal{C} = M + N - 1$ and $|\mathcal{C}| = 2^{M+N-1}$.*

Proof. The equation $\delta(\epsilon, \eta) = 0$ holds exactly when

$$\epsilon_i = \eta_j = c \quad \text{for every } i, j, \qquad c \in \mathbb{F}_2. \tag{A.25}$$

Hence $\ker \delta = \{(c\mathbf{1}_M, c\mathbf{1}_N) \mid c \in \mathbb{F}_2\}$. Rank–nullity gives $\dim \mathcal{C} = M + N - 1$. Each mask has exactly two parameterizations: if $D = \delta(\epsilon, \eta)$, then

$$\delta^{-1}(D) = \{(\epsilon, \eta), (\epsilon + \mathbf{1}_M, \eta + \mathbf{1}_N)\}. \tag{A.26}$$

□

For $h = (\sigma, \tau) \in \mathcal{H} = S_M \times S_N$, let $(hX)_{ij} = X_{\sigma^{-1}(i),\tau^{-1}(j)}$. This action preserves $\mathcal{C}$, since

$$h\delta(\epsilon, \eta) = \delta(\epsilon \circ \sigma^{-1}, \eta \circ \tau^{-1}). \tag{A.27}$$

It therefore defines the semidirect product $\mathcal{G} = \mathcal{C} \rtimes \mathcal{H}$, with

$$(D, h)(D', h') = (D + hD', hh'), \qquad (D, h)^{-1} = (h^{-1}D, h^{-1}). \tag{A.28}$$

The underlying set is $\mathcal{C} \times \mathcal{H}$; the symbol $\rtimes$ specifies the multiplication in (A.28). In particular, the permutation h acts on the second mask D' before that mask is added to D. Associativity follows from the common first component $D + hD' + hh'D''$ of both three-factor products; the identity is $(0, e_{\mathcal{H}})$.

Proof of Theorem 3. The affine maps satisfy

$$D + h(D' + h'X) = D + hD' + hh'X, \tag{A.29}$$

which agrees with (A.28); the identity fixes every X. A permutation is represented by $(0, h)$ and a flip mask by $(D, e_{\mathcal{H}})$. Conversely, $(D, h) = (D, e_{\mathcal{H}})(0, h)$, so these maps are precisely the compositions of the four allowed operations.

If $D + hX = X$ for every $X \in \Omega$, substitution of $X = 0$ gives $D = 0$. Applying h to a matrix whose sole nonzero entry is at (i, j) then shows that $(\sigma(i), \tau(j)) = (i, j)$ for every i, j. Thus $h = e_{\mathcal{H}}$, proving faithfulness also when $M = 1$ or $N = 1$. Finally,

$$|\mathcal{G}| = |\mathcal{C}||\mathcal{H}| = 2^{M+N-1}M!N!. \tag{A.30}$$

Faithfulness of the action does not require the stabilizer of an individual matrix to be trivial. □

A.2.2 Normalization and orbit representatives

Define

$$U = \{Y \in \Omega \mid Y_{i1} = 0 \ (i \in [M]),\ Y_{1j} = 0 \ (j \in [N])\},$$
$$\pi(X)_{ij} = X_{ij} + X_{i1} + X_{1j} + X_{11}. \tag{A.31}$$

The $(M-1)(N-1)$ entries outside the first row and column are free. Thus $\dim U = (M-1)(N-1)$, including the cases in which a dimension is one.

Lemma 3 (One normalized matrix per flip class)**.** *The map π is a linear projection onto U with kernel $\mathcal{C}$. In particular,*

$$\pi(X) = \pi(X') \iff X + X' \in \mathcal{C}, \qquad \Omega = U \oplus \mathcal{C}, \qquad \Omega/\mathcal{C} \cong U. \tag{A.32}$$

Proof. Formula (A.31) is linear and gives $\pi(X)_{i1} = \pi(X)_{1j} = 0$. If $Y \in U$, it also gives $\pi(Y) = Y$. Hence π is a projection onto U. Moreover,

$$\pi(\delta(\epsilon, \eta))_{ij} = (\epsilon_i + \eta_j) + (\epsilon_i + \eta_1) + (\epsilon_1 + \eta_j) + (\epsilon_1 + \eta_1) = 0. \tag{A.33}$$

Conversely, $\pi(X) = 0$ implies $X_{ij} = X_{i1} + X_{1j} + X_{11}$, a flip mask with parameters $\epsilon_i = X_{i1}$ and $\eta_j = X_{1j} + X_{11}$. Therefore

$$\ker \pi = \mathcal{C}. \tag{A.34}$$

By linearity, $\pi(X) = \pi(X')$ is therefore equivalent to $X + X' \in \mathcal{C}$. The decomposition $X = \pi(X) + (X + \pi(X))$ has its two terms in U and $\mathcal{C}$, respectively. It is unique because $Y \in U \cap \mathcal{C}$ implies $Y = \pi(Y) = 0$. □

A.2.3 The induced action and orbit sizes

A permutation of a normalized matrix need not retain its zero border. The induced action on U therefore includes normalization:

$$Y \longmapsto \pi(hY), \qquad h \in \mathcal{H}. \tag{A.35}$$

Because $Z + \pi(Z) \in \mathcal{C}$ and $h\mathcal{C} = \mathcal{C}$,

$$\pi\big(h\pi(Z)\big) = \pi(hZ) \qquad (Z \in \Omega). \tag{A.36}$$

In particular, $\pi(h\pi(h'Y)) = \pi(hh'Y)$ and $\pi(Y) = Y$ for $Y \in U$. These are the action and identity laws. This induced action need not be faithful for the orbit count to apply.

Proof of Theorem 4. By Lemma 3 and (A.36),

$$\begin{aligned} X' \in \mathcal{G} \cdot X &\iff \exists h \in \mathcal{H} :\ X' + hX \in \mathcal{C} \\ &\iff \exists h \in \mathcal{H} :\ \pi(X') = \pi(hX) \\ &\iff \exists h \in \mathcal{H} :\ \pi(X') = \pi(h\pi(X)). \end{aligned} \tag{A.37}$$

Thus $X \mapsto \pi(X)$ induces a bijection between $\Omega/\mathcal{G}$ and $U/\mathcal{H}$.

Fix a total order on U. The induced orbit of $\pi(X)$ is the finite set $\{\pi(hX) \mid h \in \mathcal{H}\}$. Two such sets are equal if they are in the same orbit, and disjoint otherwise. Hence

$$X' \in \mathcal{G} \cdot X \quad \iff \quad \min_{h\in\mathcal{H}} \pi(hX') = \min_{h\in\mathcal{H}} \pi(hX). \tag{A.38}$$

Here the partition property follows directly from invertibility: if $Z = gX = g'X'$, then $X' = (g')^{-1}gX$ and the two orbits coincide.

For the stabilizer $\mathcal{G}_X = \{g \in \mathcal{G} \mid gX = X\}$,

$$g\mathcal{G}_X = g'\mathcal{G}_X \iff (g')^{-1}g \in \mathcal{G}_X \iff gX = g'X. \tag{A.39}$$

Thus the left cosets correspond bijectively to orbit elements, giving

$$|\mathcal{G} \cdot X| = \frac{|\mathcal{G}|}{|\mathcal{G}_X|}. \tag{A.40}$$

No normality assumption on $\mathcal{G}_X$ is needed. □

A.2.4 Fixed flip classes and Burnside's formula

Fix $h = (\sigma, \tau) \in \mathcal{H}$ with row cycle type $\lambda = (a_1, \dots, a_r)$ and column cycle type $\mu = (b_1, \dots, b_s)$. All cycles, including fixed indices, are included. Write

$$d_{uv} = \gcd(a_u, b_v), \qquad K = \sum_{u=1}^{r}\sum_{v=1}^{s} d_{uv}. \tag{A.41}$$

A position in the block formed by cycles u, v returns after $\operatorname{lcm}(a_u, b_v)$ simultaneous permutation steps. Therefore that block has $\frac{a_u b_v}{\operatorname{lcm}(a_u,b_v)} = d_{uv}$ position cycles.

For flip parameters ϵ, η, let e_u be the sum of the ϵ_i over row cycle u, and f_v the sum of the η_j over column cycle v. These parities determine whether

$$X = \delta(\epsilon, \eta) + hX \tag{A.42}$$

has a solution.

Lemma 4 (Compatibility on a cycle block)**.** *Equation* (A.42) *is solvable exactly when*

$$\left(\frac{b_v}{d_{uv}} \bmod 2\right) e_u + \left(\frac{a_u}{d_{uv}} \bmod 2\right) f_v = 0 \qquad (u \in [r],\ v \in [s]). \tag{A.43}$$

When these conditions hold, there are 2^K solutions in Ω.

Proof. Within the block u, v, put $\ell = \operatorname{lcm}(a_u, b_v)$ and follow one position cycle, $(i_t, j_t) = (\sigma^{-t}(i_0), \tau^{-t}(j_0))$ for $0 \le t < \ell$. The fixed-point equation becomes

$$X_{i_t j_t} = \epsilon_{i_t} + \eta_{j_t} + X_{i_{t+1} j_{t+1}}, \qquad t \pmod{\ell}. \tag{A.44}$$

Summing around the cycle cancels every matrix entry twice, so consistency requires

$$0 = \sum_{t=0}^{\ell-1} (\epsilon_{i_t} + \eta_{j_t}) = \left(\frac{b_v}{d_{uv}} \bmod 2\right) e_u + \left(\frac{a_u}{d_{uv}} \bmod 2\right) f_v. \tag{A.45}$$

Indeed, each row index occurs $\frac{\ell}{a_u} = \frac{b_v}{d_{uv}}$ times and each column index occurs $\frac{\ell}{b_v} = \frac{a_u}{d_{uv}}$ times. The same condition holds for all d_{uv} position cycles in this block.

If a condition fails, the recurrence cannot close. Otherwise, either choice of the first entry determines the other entries consistently. The K position cycles are disjoint, so their initial entries are independent and yield exactly 2^K matrices. □

For these fixed types, abbreviate $T(\lambda, \mu)$ by T, where T is the coefficient matrix in (18). The number of compatible parity vectors is $2^{r+s-\operatorname{rank} T}$. Each prescribed parity vector is realized by

$$\prod_{u=1}^{r} 2^{a_u - 1} \prod_{v=1}^{s} 2^{b_v - 1} = 2^{M+N-r-s} \tag{A.46}$$

choices of the individual flip bits. Consequently,

$$\begin{aligned} &\sum_{\epsilon \in \mathbb{F}_2^M,\, \eta \in \mathbb{F}_2^N} \big|\{X \in \Omega \mid X = \delta(\epsilon, \eta) + hX\}\big| \\ &\quad = 2^{M+N-r-s}\, 2^{r+s-\operatorname{rank} T}\, 2^K = 2^{M+N+K-\operatorname{rank} T}. \end{aligned} \tag{A.47}$$

Lemma 2 gives two parameterizations per mask. Dividing by two therefore gives

$$\sum_{D \in \mathcal{C}} \big|\{X \in \Omega \mid X = D + hX\}\big| = 2^{M+N+K-\operatorname{rank} T - 1}. \tag{A.48}$$

A flip class $X + \mathcal{C}$ is fixed by h exactly when $X + hX \in \mathcal{C}$. For each X' in such a class, the unique mask satisfying $X' = D + hX'$ is $D = X' + hX'$. Thus every fixed class contributes $|\mathcal{C}|$ pairs (D, X') to (A.48), whereas a class that is not fixed contributes none. Since normalization identifies flip classes with U, we obtain

$$\begin{aligned} \big|\{Y \in U \mid \pi(hY) = Y\}\big| &= \frac{1}{|\mathcal{C}|} \sum_{D \in \mathcal{C}} \big|\{X \in \Omega \mid X = D + hX\}\big| \\ &= 2^{K - \operatorname{rank} T}. \end{aligned} \tag{A.49}$$

The divisions by two and by $|\mathcal{C}|$ remove different redundancies: flip parameterizations and representatives within a flip class.

Proof of Theorem 5. Burnside's lemma for the induced action on U gives

$$A(M, N) = \frac{1}{M!N!} \sum_{h \in \mathcal{H}} \big|\{Y \in U \mid \pi(hY) = Y\}\big|. \tag{A.50}$$

If m_k is the multiplicity of part k in $\lambda \vdash M$, then $z_\lambda = \prod_{k \geq 1} k^{m_k} m_k!$ and the number of permutations of type λ is $\frac{M!}{z_\lambda}$. Applying the corresponding formula for $\mu \vdash N$ and grouping (A.50) by cycle type yields

$$\begin{aligned} A(M, N) &= \frac{1}{M!N!} \sum_{\lambda \vdash M} \sum_{\mu \vdash N} \frac{M!}{z_\lambda} \frac{N!}{z_\mu} 2^{K - \operatorname{rank} T} \\ &= \sum_{\lambda \vdash M} \sum_{\mu \vdash N} \frac{2^{K - \operatorname{rank} T}}{z_\lambda z_\mu}. \end{aligned} \tag{A.51}$$

Here K and T are evaluated at each pair (λ, μ), as in (19). □

A.2.5 A CLOSED FORM FOR THE COMPATIBILITY NULLITY

Let $\nu_2(t)$ denote the exponent of two in a positive integer t. Set

$$a_* = \min_{u \in [r]} \nu_2(a_u), \qquad b_* = \min_{v \in [s]} \nu_2(b_v). \tag{A.52}$$

Theorem A.1 (Closed-form nullity)**.** *For the compatibility matrix $T = T(\lambda, \mu)$,*

$$\dim \ker T = \begin{cases} 1, & a_* = b_*, \\ |\{u \in [r] \mid \nu_2(a_u) < b_*\}|, & a_* < b_*, \\ |\{v \in [s] \mid \nu_2(b_v) < a_*\}|, & a_* > b_*. \end{cases} \tag{A.53}$$

The cardinalities count cycle occurrences, including repeated lengths.

Proof. Since $\nu_2(d_{uv}) = \min\{\nu_2(a_u), \nu_2(b_v)\}$, condition (A.43) reduces to

$$\begin{cases} e_u = f_v, & \nu_2(a_u) = \nu_2(b_v), \\ e_u = 0, & \nu_2(a_u) > \nu_2(b_v), \\ f_v = 0, & \nu_2(a_u) < \nu_2(b_v). \end{cases} \tag{A.54}$$

If $a_* = b_*$, choose cycles u_0, v_0 attaining this minimum. Pairing every row cycle with v_0, and every column cycle with u_0, forces

$$e_u = 0 \quad \text{when } \nu_2(a_u) > a_*, \qquad f_v = 0 \quad \text{when } \nu_2(b_v) > a_*. \tag{A.55}$$

All variables at the minimum are equal, since each minimum row cycle is paired with every minimum column cycle. Either value of their common bit satisfies all equations in (A.54); hence the nullity is one.

If $a_* < b_*$, a row cycle attaining a_* forces $f_v = 0$ for every v. A column cycle attaining b_* then forces $e_u = 0$ whenever $\nu_2(a_u) \geq b_*$. The remaining row variables, for which $\nu_2(a_u) < b_*$, occur only in equations of the form $f_v = 0$ and are free. Thus

$$\ker T = \{(e, f) \in \mathbb{F}_2^r \times \mathbb{F}_2^s \mid f = 0,\ e_u = 0 \text{ for every } u \text{ with } \nu_2(a_u) \geq b_*\}. \tag{A.56}$$

Its dimension is the number of remaining row variables. When $a_* > b_*$, interchanging rows and columns gives $e = 0$, with f_v free precisely when $\nu_2(b_v) < a_*$. This proves the third line as well. □

Rank–nullity gives the equivalent count

$$A(M, N) = \sum_{\lambda \vdash M} \sum_{\mu \vdash N} \frac{2^{K - r - s + \dim \ker T}}{z_\lambda z_\mu}. \tag{A.57}$$

The exponent is nonnegative: T has rs rows and $K = \sum_{u,v} d_{uv} \geq rs$, so $K - r - s + \dim \ker T = K - \operatorname{rank} T \geq 0$.

A.2.6 Boundary cases and a small exact check

When $M = 1$, the only row partition is (1), so $r = 1$, $a_* = 0$, and $K = s$. If $b_* = 0$, the first line of (A.53) gives nullity one; if $b_* > 0$, the second line gives the same value. Thus the exponent in (A.57) is zero and

$$A(1, N) = \sum_{\mu \vdash N} \frac{1}{z_\mu} = 1. \tag{A.58}$$

The last equality follows from $\sum_{\mu \vdash N} \frac{N!}{z_\mu} = N!$, since cycle types partition S_N. By symmetry $A(M, 1) = 1$, including $A(1, 1) = 1$. This also follows directly from $U = \{0\}$ when either dimension is one.

For a small exact check, take $M = N = 3$ and order both sets of partitions as $(1,1,1), (2,1), (3)$. Their permutation multiplicities are $(1, 3, 2)$, and the corresponding fixed-class counts form the matrix

$$\begin{pmatrix} 16 & 4 & 1 \\ 4 & 4 & 1 \\ 1 & 1 & 4 \end{pmatrix}. \tag{A.59}$$

Hence

$$A(3, 3) = \frac{(1, 3, 2) \begin{pmatrix} 16 & 4 & 1 \\ 4 & 4 & 1 \\ 1 & 1 & 4 \end{pmatrix} (1, 3, 2)^\top}{3!3!} = \frac{108}{36} = 3. \tag{A.60}$$

Both the action and the count concern a fixed matrix size; they do not identify matrices of different dimensions.

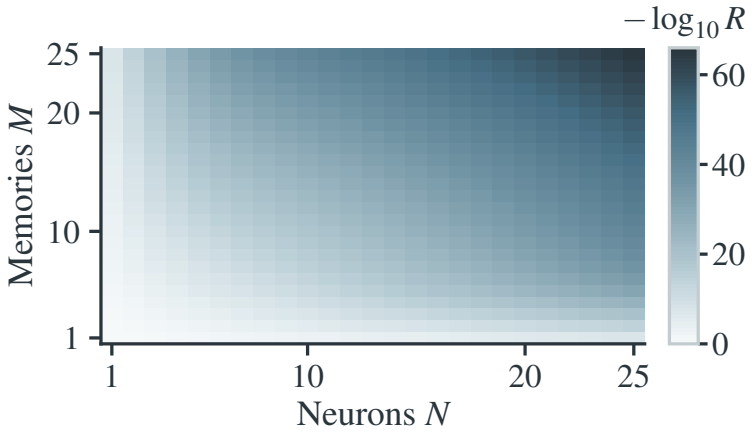


Figure 4: Candidate-space compression obtained from the exact orbit count for $1 \le M, N \le 25$. The color scale reports $\log_{10}(1/R(M,N))$, where $R(M,N) = \frac{A(M,N)}{2^{MN}}$. Thus a value c means that one orbit representative replaces 10^c original memory matrices on average.

A.2.7 Consequences for Hebbian dynamics

Proof of Theorem 6. The correspondence between binary orbits and memory equivalence gives $\Xi' = P\Xi Q$ for signed permutation matrices P, Q. The proofs of Theorems 1 and 2 give

$$W_{\Xi'} = Q^\top W_\Xi Q, \qquad F_{\Xi'}(VQ) = F_\Xi(V)Q, \tag{A.61}$$

$$\phi(V) = VQ, \qquad \phi^{-1}(V') = V'Q^\top. \tag{A.62}$$

These statements use the synchronous update that retains a neuron's state at zero local field. For the energy defined in Section 3,

$$\begin{aligned} E_{\Xi'}(VQ) &= -\frac{1}{2}(VQ)(Q^\top W_\Xi Q)(VQ)^\top \\ &= -\frac{1}{2}VW_\Xi V^\top = E_\Xi(V), \end{aligned} \tag{A.63}$$

since $QQ^\top = I_N$. The same identities hold under a common positive normalization of both weight matrices. □

Iterating the conjugacy gives $F^t_{\Xi'}(VQ) = F^t_\Xi(V)Q$ for every $t \ge 0$. Since ϕ is a bijection, it maps a periodic orbit to a periodic orbit of the same least period, and maps its basin bijectively to the corresponding basin. This includes fixed points and requires no assumption that all synchronous trajectories converge to fixed points. These memory transformations are sufficient certificates of dynamical equivalence; the converse classification is not asserted.

A.3 Exact Orbit Recognition

Throughout this appendix, $\Xi_1, \Xi_2 \in \{-1, 1\}^{M \times N}$ with $M, N \geq 1$. A transformation is recorded by the permutations and signs (p, q, ρ, κ) in (21). All comparisons below are exact.

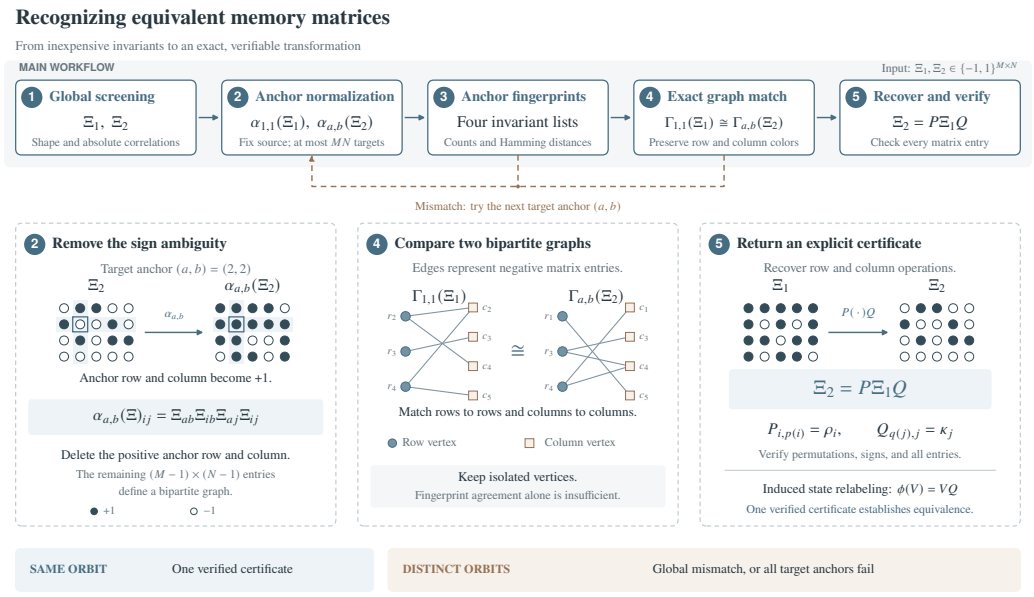


Figure 5: Exact same-orbit recognition and certificate recovery. Inexpensive invariants reject incompatible matrices before anchor normalization. The remaining candidates are compared by anchor fingerprints and a color-preserving bipartite-graph isomorphism test. A verified transformation certificate establishes equivalence; a global mismatch or failure of every target anchor establishes that the matrices lie in distinct orbits.

A.3.1 Anchor Normalization and Transformation Recovery

Lemma 5 (Normalization at a fixed anchor)**.** *The matrix $\alpha_{a,b}(\Xi)$ in* (23) *is the unique row/column sign transform of Ξ whose row a and column b are positive. It is unchanged by further row/column sign transformations of the input.*

Proof. Write $Y = \alpha_{a,b}(\Xi)$. The signs $\rho_i = \Xi_{ab}\Xi_{ib}$ and $\kappa_j = \Xi_{aj}$ give

$$Y_{ij} = \rho_i \Xi_{ij} \kappa_j, \qquad Y_{aj} = \Xi_{ab}^2 \Xi_{aj}^2 = 1, \qquad Y_{ib} = \Xi_{ab}^2 \Xi_{ib}^2 = 1. \tag{A.64}$$

For arbitrary signs ρ_i, κ_j, let $\Xi'_{ij} = \rho_i \Xi_{ij} \kappa_j$. Substitution gives

$$\alpha_{a,b}(\Xi')_{ij} = \rho_a^2 \rho_i^2 \kappa_b^2 \kappa_j^2 \Xi_{ab}\Xi_{ib}\Xi_{aj}\Xi_{ij} = \alpha_{a,b}(\Xi)_{ij}. \tag{A.65}$$

To prove uniqueness, suppose $Z_{ij} = \rho_i \Xi_{ij} \kappa_j$ and $Z_{aj} = Z_{ib} = 1$ for every i, j. Then

$$\rho_i = \Xi_{ib}\kappa_b, \qquad \kappa_j = \rho_a \Xi_{aj} = \Xi_{ab}\kappa_b \Xi_{aj}, \qquad Z_{ij} = \Xi_{ab}\Xi_{ib}\Xi_{aj}\Xi_{ij} = Y_{ij}. \tag{A.66}$$

The normalized matrix is unique, although the sign pair has the redundancy $(\rho, \kappa) \mapsto (-\rho, -\kappa)$. The anchor and row/column labels remain fixed here. □

Proof of Theorem 7. Suppose (21) holds and choose $a = p^{-1}(1)$, $b = q^{-1}(1)$. Thus $p(a) = q(b) = 1$. The signs cancel in

$$\begin{aligned} \alpha_{a,b}(\Xi_2)_{ij} &= (\Xi_2)_{ab}(\Xi_2)_{ib}(\Xi_2)_{aj}(\Xi_2)_{ij} \\ &= \rho_a^2 \rho_i^2 \kappa_b^2 \kappa_j^2 (\Xi_1)_{11}(\Xi_1)_{p(i),1}(\Xi_1)_{1,q(j)}(\Xi_1)_{p(i),q(j)} \\ &= \alpha_{1,1}(\Xi_1)_{p(i),q(j)}. \end{aligned} \tag{A.67}$$

Restricting p, q to the non-anchor indices gives bijections

$$[M] \setminus \{a\} \longrightarrow [M] \setminus \{1\}, \qquad [N] \setminus \{b\} \longrightarrow [N] \setminus \{1\}. \tag{A.68}$$

Equation (A.67) preserves negative entries, so these restrictions define a color-preserving isomorphism from the target graph to the source graph.

Conversely, let $\psi : \Gamma_{1,1}(\Xi_1) \to \Gamma_{a,b}(\Xi_2)$ be a color-preserving isomorphism. Extend its row and column maps by $1 \mapsto a$ and $1 \mapsto b$, respectively, and invert them to obtain p, q. Then

$$p(a) = q(b) = 1, \qquad \alpha_{a,b}(\Xi_2)_{ij} = \alpha_{1,1}(\Xi_1)_{p(i),q(j)} \quad (i \neq a,\ j \neq b). \tag{A.69}$$

Both normalized borders are positive, so the equality holds at the anchor row and column as well. With the signs in (25)–(26),

$$\begin{aligned} \rho_i(\Xi_1)_{p(i),q(j)}\kappa_j &= (\Xi_2)_{ab}(\Xi_2)_{ib}(\Xi_2)_{aj}\alpha_{1,1}(\Xi_1)_{p(i),q(j)} \\ &= (\Xi_2)_{ab}(\Xi_2)_{ib}(\Xi_2)_{aj}\alpha_{a,b}(\Xi_2)_{ij} \\ &= (\Xi_2)_{ij}. \end{aligned} \tag{A.70}$$

This is the required certificate. Isolated vertices are retained, so the argument also applies to positive, repeated, or opposite rows and columns. If $M = 1$ or $N = 1$, the graphs have one empty vertex side and no edges; the same argument holds, or one may use the direct certificates given below. □

A.3.2 Screening candidate anchors

The correlation tests in (22) follow from

$$\begin{aligned} \sum_j (\Xi_2)_{ij}(\Xi_2)_{kj} &= \rho_i\rho_k \sum_j (\Xi_1)_{p(i),j}(\Xi_1)_{p(k),j}, \\ \sum_i (\Xi_2)_{ij}(\Xi_2)_{i\ell} &= \kappa_j\kappa_\ell \sum_i (\Xi_1)_{i,q(j)}(\Xi_1)_{i,q(\ell)}. \end{aligned} \tag{A.71}$$

Absolute values remove the signs, and the permutations reorder the unordered pairs. Multiplicities and the distinction between rows and columns are therefore retained in the two tests.

For each anchor, four sorted lists provide a further necessary test: the negative-entry counts of the non-anchor rows and columns, and the Hamming distances between all unordered row pairs and all unordered column pairs after normalization. Distances are computed on the non-anchor coordinates; the pairs include the anchor row or column. Equal normalized matrices up to anchor-preserving permutations have the same four lists.

These lists can be computed from correlations without storing every normalized matrix. For $Y = \alpha_{a,b}(\Xi)$,

$$\begin{aligned} \big|\{j \in [N] \setminus \{b\} \mid Y_{ij} = -1\}\big| &= \frac{N - \Xi_{ab}\Xi_{ib}\sum_j \Xi_{aj}\Xi_{ij}}{2}, \\ \big|\{j \in [N] \setminus \{b\} \mid Y_{ij} \neq Y_{kj}\}\big| &= \frac{N - \Xi_{ib}\Xi_{kb}\sum_j \Xi_{ij}\Xi_{kj}}{2}. \end{aligned} \tag{A.72}$$

Indeed, $Y_{ib} = Y_{kb} = 1$ and

$$\sum_{j \neq b} Y_{ij}Y_{kj} = \Xi_{ib}\Xi_{kb}\sum_j \Xi_{ij}\Xi_{kj} - 1; \tag{A.73}$$

the second equality follows by writing this correlation as $N - 1 - 2d$, where d is the number of unequal entries. Taking $k = a$ gives the first equality. Interchanging rows and columns gives the other two lists. The row-distance list depends only on b, and the column-distance list only on a, so they are precomputed before the anchor scan. A mismatch rejects the anchor; matching lists still require an exact graph test.

A.3.3 Exactness of the colored graph search

Lemma 6 (Exactness of the graph search)**.** *Algorithm 2 terminates and returns a color-preserving isomorphism if and only if one exists.*

Proof. Let ψ preserve the current colors and adjacencies. For each source vertex u, the neighbor bijection induced by ψ gives

$$\chi_1(u) = \chi_2(\psi(u)), \qquad \{\{\chi_1(v) \mid v \text{ adjacent to } u\}\} = \{\{\chi_2(w) \mid w \text{ adjacent to } \psi(u)\}\}. \quad \text{(A.74)}$$

The joint dictionary therefore assigns equal new colors to u and $\psi(u)$. Refinement preserves every compatible isomorphism and cannot reject an isomorphic pair. In the graph stage $M, N > 1$; each graph has $n = M + N - 2$ vertices. On a nonrejected round, both graphs have the same color multiplicities, so there are at most n joint classes. Every nonterminal round splits a class, and refinement terminates.

If a nonsingleton class is selected, the algorithm tries every possible image of u with its color, including $v = \psi(u)$. Giving this pair a fresh common color preserves ψ. At least one compatible branch thus survives at each level. Individualization strictly increases the number of classes, so a path has at most $n - 2$ individualizations. The branching is finite, and the resulting search tree is finite.

At a singleton leaf, equal colors determine the only compatible bijection. The final test accepts it precisely when

$$uv \text{ is an edge of } \Gamma_1 \quad \Longleftrightarrow \quad \psi(u)\psi(v) \text{ is an edge of } \Gamma_2, \quad \text{(A.75)}$$

with row and column sides preserved. Any existing isomorphism therefore reaches an accepted leaf, and every returned map is an isomorphism. □

A.3.4 Exactness of orbit recognition

Theorem A.2 (Exactness of orbit recognition)**.** *Algorithm 3 terminates on valid finite input. It returns a certificate if and only if $\Xi_1 \sim \Xi_2$, and returns* NO *otherwise.*

Proof. Every positive answer satisfies (21): this follows directly for the identity and single-axis branches, and from (A.70) and the final check otherwise. Thus a returned certificate implies $\Xi_1 \sim \Xi_2$.

For equivalent matrices with $M, N > 1$, Theorem 7 gives an anchor (a, b) for which

$$\Gamma_{1,1}(\Xi_1) \cong \Gamma_{a,b}(\Xi_2). \quad \text{(A.76)}$$

Neither the correlation tests nor the four anchor lists reject this match, since each is preserved by the certificate. Lemma 6 then finds an isomorphism, and sign recovery returns a valid certificate. Consequently an equivalent pair cannot reach NO; by contraposition, a negative answer certifies inequivalence.

There are MN target anchors. Each screening operation is finite, and every graph search terminates by Lemma 6. The whole algorithm therefore terminates. An externally interrupted run is not a negative answer. □

A.4 Invariance of the Named-Memory Recovery Score

For $x \in \Sigma$ and $r \in \{0, \dots, N\}$, define the Hamming shell

$$L_r(x) := \{V \in \Sigma \mid d_{\mathrm{H}}(V, x) = r\}. \quad \text{(A.77)}$$

For $\ell \in \mathbb{Z}_{\geq 0}$, let

$$u(\Xi) := \big|\{i \in [M] \mid F_\Xi(\xi_i) = \xi_i\}\big|, \quad \text{(A.78)}$$

$$s_{r,\ell}(\Xi) := \sum_{\substack{i \in [M] \\ F_\Xi(\xi_i) = \xi_i}} \big|\{V \in L_r(\xi_i) \mid F_\Xi^\ell(V) = \xi_i\}\big|. \quad \text{(A.79)}$$

A nonfixed stored row is absent from the sum in (A.79), and therefore contributes no successful trials.

Lemma 7 (Orbit invariance of the experimental score)**.** *If $\Xi' = P\Xi Q$, then for every admissible r and ℓ,*

$$u(\Xi') = u(\Xi), \qquad s_{r,\ell}(\Xi') = s_{r,\ell}(\Xi). \quad \text{(A.80)}$$

Proof. First note that the update map is odd. For each $V \in \Sigma$, the local field of $-V$ is the negative of the local field of V. At a nonzero field, the updated spin therefore changes sign. At zero field, spin retention gives $F_\Xi(-V)_j = -V_j = -F_\Xi(V)_j$. Hence

$$F_\Xi(-V) = -F_\Xi(V), \qquad F_\Xi^\ell(-V) = -F_\Xi^\ell(V). \tag{A.81}$$

The second identity follows by induction on ℓ.

Use the entry convention from (A.2). Row i of Ξ' is

$$\xi'_i = \rho_i \xi_{p(i)} Q. \tag{A.82}$$

For this row, define $\theta_i : \Sigma \to \Sigma$ by $\theta_i(V) = \rho_i V Q$. Since Q is a signed permutation and $\rho_i \in \{-1, 1\}$, the map θ_i is bijective. The Hamming isometry in (A.17) gives

$$\theta_i\big(L_r(\xi_{p(i)})\big) = L_r(\xi'_i). \tag{A.83}$$

Combining conjugacy (A.18) with oddness (A.81) yields, for all $V \in \Sigma$,

$$F_{\Xi'}^\ell\big(\theta_i(V)\big) = \rho_i F_\Xi^\ell(V) Q. \tag{A.84}$$

At $\ell = 1$ and $V = \xi_{p(i)}$, this identity shows that ξ'_i is fixed if and only if $\xi_{p(i)}$ is fixed. For a fixed row, (A.82) and (A.84) give

$$F_{\Xi'}^\ell\big(\theta_i(V)\big) = \xi'_i \quad \Longleftrightarrow \quad F_\Xi^\ell(V) = \xi_{p(i)}. \tag{A.85}$$

Thus corresponding rows contribute equal shell counts. Finally, $p \in S_M$ merely reorders these contributions, proving both identities in (A.80). □

B Algorithms for Orbit Counting and Recognition

B.1 Exact integer evaluation of the orbit count

The cycle-type multiplicities $\frac{M!}{z_\lambda}$ and $\frac{N!}{z_\mu}$ are integers. Therefore the numerator

$$\sum_{\lambda \vdash M} \sum_{\mu \vdash N} \frac{M!}{z_\lambda} \frac{N!}{z_\mu} 2^{K - r - s + \dim \ker T} \tag{B.1}$$

can be accumulated with integer arithmetic. Burnside's lemma guarantees that division by $M!N!$ is exact. Algorithm 1 uses the closed form (A.53) to avoid constructing T.

Algorithm 1 Exact count $A(M, N)$ of algebraic equivalence classes

Require: Positive integers M, N; arbitrary-precision integer arithmetic
Ensure: The number $A(M, N)$ of $\mathcal{G}$-orbits in Ω

$t \leftarrow 0$
for each integer partition $\lambda = (a_1, \dots, a_r)$ of M **do**
  Compute z_λ and $\nu_2(a_1), \dots, \nu_2(a_r)$
  for each integer partition $\mu = (b_1, \dots, b_s)$ of N **do**
    Compute z_μ and $\nu_2(b_1), \dots, \nu_2(b_s)$
    $K \leftarrow \sum_{u=1}^{r} \sum_{v=1}^{s} \gcd(a_u, b_v)$
    Evaluate $\dim \ker T$ using (A.53)
    $t \leftarrow t + \frac{M!}{z_\lambda} \frac{N!}{z_\mu} 2^{K - r - s + \dim \ker T}$
  end for
end for
return $\frac{t}{M!N!}$ ▷ exact integer division

The loops visit each partition pair once. Partition multiplicities and valuations may be cached. For each pair, at most MN greatest-common-divisor evaluations are required; the cost of the remaining integer operations depends on the operand bit lengths. The algorithm enumerates cycle types, without constructing memory matrices or simulating their dynamics.

B.2 EXACT COLORED GRAPH SEARCH

We use color refinement followed by individualization and search (McKay & Piperno, 2014). Initially, row vertices have color 0 and column vertices color 1. At each refinement round, replace a vertex's color by an encoding of its current color and the multiset of its neighbors' colors. A single dictionary is shared by both graphs: two encodings agree exactly when the full descriptors agree. Unequal color multiplicities reject the pair. Otherwise, repeat until no color class splits. The current color is part of the descriptor, so distinct classes cannot merge.

Algorithm 2 specifies the remaining search recursively. Its inputs χ_1, χ_2 are the two color arrays; the graphs remain fixed throughout a recursive call. Every branch uses a copy of the parent arrays.

Algorithm 2 SEARCH: exact color-preserving graph isomorphism

Require: Graphs Γ_1, Γ_2 and color arrays χ_1, χ_2
Jointly refine the colors until stable, or reject unequal multiplicities
if refinement rejects the pair **then**
    return $\perp$
end if
if every color occurs once in each graph **then**
    Construct ψ by matching equal colors
    if ψ is bijective and preserves the row/column sides and all adjacencies **then**
        return ψ
    end if
    return $\perp$
end if
Choose a smallest nonsingleton source class and a vertex u in it
for each target vertex v with $\chi_2(v) = \chi_1(u)$ **do**
    Copy the arrays to χ_1', χ_2'; give u, v one common fresh color
    $\psi \leftarrow \text{SEARCH}(\Gamma_1, \Gamma_2, \chi_1', \chi_2')$
    if $\psi \neq \perp$ **then**
        return ψ
    end if
end for
return $\perp$

The supplied implementation uses an explicit stack for this depth-first search. A failed subproblem may be cached by its two complete color arrays in fixed vertex order, but only after its verified leaf or all its children have failed. Identical arrays impose identical constraints for the fixed graph pair; color histograms alone are not valid cache keys.

B.3 RECOGNITION AND CERTIFICATE VERIFICATION

The graph vertices retain their original row and column indices, excluding the anchor. This convention allows a graph isomorphism to be extended and inverted without ambiguity.

Algorithm 3 Recognize an orbit and recover a transformation

Require: Nonempty bipolar matrices Ξ_1, Ξ_2
Ensure: A verified certificate (p, q, ρ, κ), or NO

```
Validate entries and rectangular shape; invalid input is an error
if the shapes differ then
    return NO
end if
if Ξ1 = Ξ2 then
    return identity permutations and unit signs
end if
if M = 1 or N = 1 then
    return the direct certificate given below
end if
Compute the row and column correlations of both matrices
if either multiset in (22) differs then
    return NO
end if
Precompute the sorted distance lists from (A.72)
Compare each target distance list with the corresponding source list once
Compute the source degree lists at anchor (1, 1)
Γ1 ← Γ1,1(Ξ1)
for a = 1, ..., M and b = 1, ..., N do
    if a precompared distance list or a newly computed degree list differs then
        continue
    end if
    Γ2 ← Γa,b(Ξ2)
    Initialize χ1, χ2 with row color 0 and column color 1
    ψ ← SEARCH(Γ1, Γ2, χ1, χ2)
    if ψ ≠ ⊥ then
        Extend the two parts of ψ by 1 ↦ a and 1 ↦ b
        Invert these bijections to obtain p, q
        Compute ρ, κ by (25)–(26)
        Verify p ∈ S_M, q ∈ S_N, all signs, and (21) for every entry
        On success, return (p, q, ρ, κ); otherwise report an internal error
    end if
end for
return NO
```

When $N = 1$, identity permutations and $\rho_i = (\Xi_1)_{i1}(\Xi_2)_{i1}$, $\kappa_1 = 1$ give the certificate. Otherwise, when $M = 1$, use identity permutations and $\rho_1 = 1$, $\kappa_j = (\Xi_1)_{1j}(\Xi_2)_{1j}$. These formulas include the one-cell case. In the general case, checking the permutations as well as the reconstructed entries is necessary: repeated rows may otherwise permit a nonbijective index map to pass an entrywise test.

B.4 A FOUR-BY-FIVE EXAMPLE WITH ALL FOUR OPERATIONS

Consider the memories

$$\Xi_1 = \begin{pmatrix} 1 & -1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & 1 & 1 \\ -1 & -1 & 1 & 1 & -1 \end{pmatrix}, \qquad \Xi_2 = \begin{pmatrix} -1 & 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 \\ -1 & -1 & 1 & -1 & -1 \\ -1 & -1 & -1 & 1 & 1 \end{pmatrix}. \tag{B.2}$$

A target-to-source witness, with permutations in one-line notation, is

$$p = (3, 1, 4, 2), \qquad q = (4, 2, 5, 1, 3), \qquad \rho = (-1, 1, -1, 1), \qquad \kappa = (1, -1, 1, -1, 1). \tag{B.3}$$

Both permutations and both families of sign flips are nontrivial. Because $p(2) = q(4) = 1$, source anchor $(1, 1)$ corresponds to target anchor $(2, 4)$. The normalized matrices are

$$\alpha_{1,1}(\Xi_1) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 & -1 \end{pmatrix}, \qquad \alpha_{2,4}(\Xi_2) = \begin{pmatrix} 1 & -1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 & -1 \\ 1 & 1 & -1 & 1 & -1 \end{pmatrix}. \quad \text{(B.4)}$$

The positive anchor borders occupy different positions. Removing the two borders produces bipartite graphs with three row vertices and four column vertices. Their row and column isomorphisms are p^{-1} and q^{-1}, which match every negative interior entry. Substitution into (25)–(26) directly recovers the displayed ρ, κ. For instance, $(\Xi_2)_{12} = (-1)(\Xi_1)_{3,2}(-1) = 1$; the same identity checks all twenty entries.

The witness need not be unique. Columns 3 and 5 of Ξ_1 are opposite; the supplied solver returns the equally valid $q' = (4, 2, 3, 1, 5)$ and $\kappa' = (1, -1, -1, -1, -1)$ with the same p, ρ. Direct reconstruction verifies both certificates. Accordingly, the required output is a valid transformation rather than the particular sequence of operations used to prepare the input.

B.5 Computational cost

Computing the correlations costs $O(M^2N + MN^2)$ arithmetic operations. Sorting and precomputing the distance lists, followed by the scan of the anchor degree lists, costs

$$O\big(NM^2 \log(M+1) + MN^2 \log(N+1) + MN\big(M \log(M+1) + N \log(N+1)\big)\big). \quad \text{(B.5)}$$

Constructing each graph costs $O(MN)$, with one source graph and at most MN target graphs. Each graph has $M + N - 2$ vertices and at most $(M-1)(N-1)$ edges. Highly symmetric graphs may require many search branches; no polynomial worst-case bound is claimed for this algorithm. The stored correlations and distance lists use $O(M^2 + N^2 + M^2N + MN^2)$ integer positions, in addition to the graphs and search storage. These are arithmetic-operation and storage bounds; bit costs also depend on the integer lengths. A certificate can be checked in $O(MN)$ entry operations after validating its permutations.

C Additional Experimental Details

C.1 Experimental protocol

The complete-space experiment evaluates one representative from each memory- matrix orbit and assigns its result to every matrix in that orbit. Orbit sizes are retained during representative generation, so the weighted histogram is the exact histogram over all ordered bipolar memory matrices rather than a Monte Carlo estimate. The column-only and complete-orbit methods recover the same histogram; they differ only in how many representatives require an HNN evaluation.

For three stored memories, column-sign normalization leaves four projective column types. Sorting their multiplicities gives one representative for each complete orbit. The implementation also records the number of ordered memory matrices represented by each multiplicity pattern. Before dynamical scoring, we checked that these orbit sizes cover the full matrix space exactly.

C.2 Complete-space timings

The one-bit task perturbs one neuron and checks the next update. The high-noise task flips 40% of the neurons, rounded down, and follows the network for eight updates. Both methods call the same scorer; their only difference is the quotient used to select matrices for evaluation. Table 2 reports all paired runs used in Figure 3. Speedup is measured as column-only wall time divided by complete-orbit wall time and is reported separately from the reduction in representative count.

Table 2: All paired runs for direct three-memory complete-orbit generation. The last column reports the median and observed range of the three timing ratios.

Task	N	Column classes	Full orbits	speedups	Median; range
One-bit	64	47,905	2,280	19.61, 14.45, 14.90	14.90; 14.45–19.61
One-bit	128	366,145	16,335	21.19, 19.87, 20.99	20.99; 19.87–21.19
One-bit	256	2,862,209	123,464	30.01, 21.07, 19.41	21.07; 19.41–30.01
40% shell	32	6,545	351	08.79, 22.48, 19.53	19.53; 8.79–22.48
40% shell	64	47,905	2,280	21.92, 20.64, 20.42	20.64; 20.42–21.92

In the two high-noise configurations, the exact complete-space mean return fractions after eight updates are 0.5236122324 for 32 neurons with 12 flips and 0.7006788764 for 64 neurons with 25 flips. These are finite-space results for three stored memories; we do not infer a monotonic trend from two values.

The direct construction above uses the special four-type action for three stored memories. We also ran a generic full-orbit generator, which traverses the projective column histograms and scores one member of each signed-row orbit. Table 3 includes two four-memory configurations to check that the computational effect is not confined to the direct generator. Every paired method recovered the same integer joint histogram and total weight for the full matrix space.

Table 3: Generic complete-orbit generation against the column-only quotient. Each entry summarizes three paired runs; the interval is the observed range.

Task	Memories / neurons	Column classes	Full orbits	Median speedup; range
One-bit	3 / 64	47,905	2,280	5.96; 5.94–6.42
One-bit	3 / 128	366,145	16,335	6.41; 6.27–6.50
One-bit	4 / 20	888,030	6,271	9.60; 9.22–9.65
Two-bit	3 / 64	47,905	2,280	10.60; 9.30–10.65
Two-bit	4 / 16	245,157	1,955	26.65; 25.65–27.15

C.3 Independent candidate banks

For this experiment, every matrix in a bank was sampled independently and uniformly from bipolar matrices with three stored memories; transformed copies were not inserted. An exact-matrix cache is

the unquotiented baseline. A cache hit means that a candidate inherits a previously evaluated score because its key has already appeared. The column-only key identifies projective column histograms; the complete key also quotients the signed row action. All methods return the score to every candidate in its original order.

Table 4 reports medians and interquartile ranges across ten independently seeded banks. The timing ratio compares column-only and complete-orbit caching and includes candidate generation, key construction, exact shell evaluation, and result assignment. The full protocol also used five predeclared radii for the 100-candidate banks. Across all configurations, $30{,}000$ candidate–radius records were produced per method. All $60{,}000$ pairwise comparisons with the exact-matrix cache agreed in fixed-row count, successful return count, and evaluated-trial count. Only fixed stored rows were scored. For unconditional recovery fractions, nonfixed stored rows were retained as failures in the denominator.

Table 4: Reuse in independently sampled candidate banks. "Hits" are evaluations avoided within a bank. Speed is column-only time divided by complete-orbit time; the interval is the observed range over ten paired banks.

N	Candidates	Complete hits	Column hits	Median speed; range	Faster banks
64	100	$36.0\ [32.5, 38.5]$	$4.0\ [3.0, 5.0]$	1.58; 1.30–1.68	10 of 10
64	1,000	$822.5\ [818.2, 824.0]$	$249.0\ [243.2, 262.0]$	4.48; 4.06–4.75	10 of 10
128	100	$18.0\ [13.2, 21.2]$	$1.0\ [0.2, 1.0]$	1.23; 1.09–1.37	10 of 10
128	1,000	$672.5\ [664.0, 678.0]$	$105.0\ [102.2, 114.0]$	2.83; 0.85–3.21	9 of 10

C.4 Validation and scope

The histogram scorers were first checked against explicit Hebbian networks over complete small matrix spaces. A separate large-instance check used 54 matrices with three or four stored memories and neuron counts from 16 to 256. It matched all $14{,}256$ one-bit updates, all $1{,}176{,}264$ two-bit updates, and $1{,}080$ dense local-field probes. Production output was accepted only when every joint bin agreed across methods and the accumulated orbit weights matched the full matrix-space size.

The complete quotient does not reduce the state space of a single HNN, and representative generation has a cost. The measured savings apply when an orbit-invariant analysis is repeated over a space or a candidate bank with equivalent matrices. The direct generator is limited to three stored memories; the generic generator handles more memories, but its histogram traversal becomes expensive as the memory count grows. Finally, the shell score is an exact finite-time statistic under the stated synchronous update rule, not evidence of asymptotic convergence.