MULTI-ATTRACTOR GNNS: SET-VALUED EXPRESSIVITY BEYOND UNIQUE EQUILIBRIA

Anonymous authors
Paper under double-blind review

ABSTRACT

Recurrent and equilibrium graph neural networks (GNNs) are often designed to have a unique fixed point or trained against a single target per graph. Yet many combinatorial and scientific problems are inherently set-valued: several valid solutions coexist for the same graph, and nothing in the problem singles one out. Training against a single designated target then makes the model fit an artifact of how that target was picked, rather than the structure of the problem. For example, when a task is invariant to node relabeling, a symmetric graph has a symmetric solution set, yet may admit no symmetric solution; isolating a single target then imposes an arbitrary symmetry-breaking choice.

In this paper, we show that multiple equilibria of recurrent GNNs are a resource rather than a defect: one weight-tied message-passing GNN can represent such set-valued equivariant maps through its attractor landscape, where different random initializations converge to different valid solutions. We establish this expressive power in two steps. First, we prove the existence of globally Lipschitz, permutation-equivariant multi-attractor dynamics that converge almost surely to valid solutions while reaching every solution branch with positive probability. Second, we establish a realization theorem proving that a recurrent message-passing GNN can approximate these dynamics and their limits with arbitrarily small error and arbitrarily high probability. This goes beyond standard universality arguments, since message passing cannot by itself distinguish symmetric nodes: we show that the evolving state keeps the symmetry broken at every step, without auxiliary node identifiers. In practice, such dynamics can be learned without solution labels from problem-specific energies. On three scientific tasks (ground states of Ising models, structural module detection in protein graphs, and steady states of chemical reaction networks), the learned updates reach multiple distinct, high-quality solutions through numerically convergent trajectories, outperforming matched unique-equilibrium, single-target, and feedforward baselines, while remaining competitive with far larger diffusion-based solvers.

1 INTRODUCTION

Recurrent graph neural networks (GNNs) repeatedly apply a shared message-passing update, allowing computation to deepen without introducing new parameters at every step Li et al. (2015). This iterative structure provides a natural mechanism for capturing long-range dependencies, while equilibrium-based variants support implicit differentiation without storing the full sequence of intermediate states Gu et al. (2020); Baker et al. (2023); Yang et al. (2025).

Consider a graph $G = (A, X)$ with n nodes, where A encodes connectivity and edge attributes, and X_i is the feature vector attached to node i. For a node-level task, a prediction $y \in \mathbb{R}^{n\times q}$ assigns a q-dimensional output y_i to each node. A feedforward GNN directly computes $y = \mathrm{GNN}_\theta(G)$. A recurrent predictor instead maintains an evolving node state $y_t \in \mathbb{R}^{n\times q}$ and repeatedly updates it:

$$y_{t+1} = \mathrm{GNN}_\theta(G, y_t), \qquad t = 0, 1, 2, \ldots,$$

starting from a deterministic or random initialization y_0. At iteration t, the GNN receives the pair $(X_i, (y_t)_i)$ at each node i and exchanges messages along graph edges to produce the updated states $(y_{t+1})_i$. The graph $G = (A, X)$ remains fixed while y_t evolves, with parameters θ shared across iterations. The usual goal is convergence to a limit y_∞ approximating a prescribed target $f(G)$.

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.

Representative implicit GNNs enforce equilibrium uniqueness through contraction or monotonicity Gu et al. (2020); Baker et al. (2023). Single-target training encourages one designated solution per graph without guaranteeing uniqueness. Global contraction imposes a stronger restriction: different initializations for a fixed graph cannot converge to different fixed points, whether training uses labels or an unsupervised objective.

Many graph or combinatorial tasks admit multiple valid solutions for the same input. Selecting one target can discard meaningful alternatives or conflict with input symmetries. For example:

- **Two coloring.** Consider proper two-coloring of the four-node cycle $1-2-3-4-1$, with identical node features: each node receives a color in $\{0,1\}$, and adjacent nodes must have different colors. There are exactly two valid assignments, $(0,1,0,1)$ and $(1,0,1,0)$, which are exchanged by a one-step rotation that leaves the input graph unchanged. A vanilla GNN receiving only the graph must assign identical outputs to all nodes and therefore cannot produce either coloring. Symmetry breaking can distinguish these nodes and enable a valid coloring, but single-target training still arbitrarily favors one equally valid solution.
- **Nonlinear equilibria.** Chemical reaction networks are represented as bipartite graphs connecting species to participating reactions, with reaction parameters as attributes; the task is to predict stationary species concentrations. Even one dynamic species can admit multiple solutions for the same graph and parameters: a Schlögl model has steady-state equation $0 = 8 - 14c + 7c^2 - c^3$ and three positive solutions $c \in \{1,2,4\}$. All three satisfy the same physical equation, which selects no preferred target. Discovering multiple steady states can reveal coexisting stable states and help explain switching between them.

These examples motivate recovering multiple valid solutions per graph without a preferred target. Recurrent GNNs provide a natural mechanism: initializations y_0 and y_0' can converge to different limits y_∞ and y_∞' under the same learned update (Figure 1). Only the initial node states change between runs; G and θ remain fixed. Random restarts thus explore multiple solutions with one model. We seek convergence to a valid solution within each run and access to the graph's different solutions across runs.

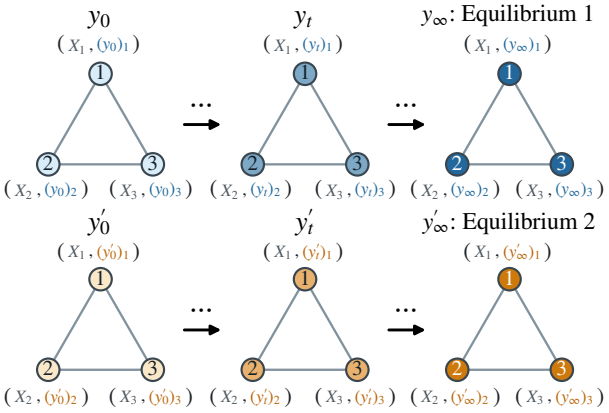


Figure 1: Different initializations can lead to different equilibria under the same recurrent GNN update, with the graph and model parameters held fixed.

We study what a standard recurrent, weight-tied GNN can represent and learn when multiple equilibria are allowed. Our contributions address three questions.

- **Can one shared update represent many solutions?** Under the stated assumptions, we prove that a single recurrent message-passing GNN can converge arbitrarily close to the solution set, with arbitrarily high probability over graphs and initializations (Theorems 1 and 2).
- **Is a unique equilibrium merely less diverse?** We uncover a more fundamental limitation: a unique equilibrium can miss every valid solution. Specifically, under suitable conditions, every continuous, equivariant, contractive update has a unique equilibrium separated from all valid solutions with positive probability. This obstruction applies to message-passing GNNs of arbitrary finite depth and width, regardless of how they are trained, and can persist even when sampled graphs are almost surely asymmetric (Theorem 3).
- **Can such dynamics be learned without solution labels, and do they pay off?** On Ising ground states, protein module detection, and chemical steady states, training on problem-specific energies or residuals from random initializations yields high numerical convergence rates and multiple distinct, high-quality solutions. The learned models achieve better average solution quality than the tested single-target, unique-equilibrium, and feedforward baselines, while remaining competitive with much larger diffusion-based solvers.

2 MAIN RESULTS

Existing GNN expressivity theory asks which single-valued invariant or equivariant maps $G \mapsto f(G)$ a GNN can approximate across a graph family (e.g., Xu et al., 2018; Keriven & Peyré, 2019; Abboud et al., 2020). We seek one GNN whose recurrent limits represent each graph's entire solution set, with random initialization selecting the solution reached. Does such a GNN exist? We prove existence (Section 2.1), illustrate it on a toy problem (Section 2.2), and show that a unique equilibrium can miss every valid solution (Section 2.3).

2.1 CAN RECURRENT GNNS REPRESENT SET-VALUED EQUIVARIANT MAPS?

To make the question precise, let $Y_n = \mathbb{R}^{n\times q}$ be the space of node-level outputs. Let $\mathcal{G}$ be a family of n-node graphs that is closed under node permutations. A permutation π acts on a graph by relabeling its nodes, and on Y_n by permuting rows. A finite solution map assigns to each $G \in \mathcal{G}$ a set

$$\mathcal{S}(G) = \{f_1(G), \dots, f_M(G)\} \subseteq Y_n,$$

where the branches f_a may coincide on some G. We require the map to be *set-valued equivariant*,

$$\mathcal{S}(\pi G) = \pi\, \mathcal{S}(G).$$

Note that *individual branches need not be equivariant.* In the two-coloring example of Section 1, rotating the cycle by one node leaves the input graph unchanged but exchanges the two valid colorings, $(0, 1, 0, 1)$ and $(1, 0, 1, 0)$. The solution set remains unchanged, as equivariance requires, but selecting either coloring as a single output would violate equivariance, since that coloring changes under the same rotation.

A recurrent GNN represents solutions through the limits of its update $y_{t+1} = \mathrm{GNN}_\theta(G, y_t)$ with $y_0 \sim \mu_n$, where μ_n is an initialization distribution on Y_n with a density and full support, such as i.i.d. standard Gaussian entries. We collect the limits reached with positive probability in the attractor set

$$\mathcal{A}_\theta(G) := \left\{ s \in Y_n : \mathbb{P}_{y_0\sim\mu_n}\left[\lim_{t\to\infty} \mathrm{GNN}_\theta^t(G, y_0) = s\right] > 0 \right\},$$

where $\mathrm{GNN}_\theta^t(G, \cdot)$ denotes t repeated applications of the block. Ideally, one shared update satisfies, for every G: (i) almost every trajectory $\{y_t\}_t$ converges to a limit in $\mathcal{S}(G)$; and (ii) every $s \in \mathcal{S}(G)$ is reached with positive probability. Together, these ensure

$$\mathcal{A}_\theta(G) = \mathcal{S}(G),$$

with no probability mass on nonconvergent trajectories or invalid limits.

We establish this in two steps. First, Theorem 1 constructs an ideal update: a globally Lipschitz, permutation-equivariant operator that satisfies (i) and (ii) exactly, on graphs where the solution branches are suitably regular. Second, Theorem 2 shows that a single recurrent message-passing GNN approximates this operator along its trajectories. As a result, the GNN's limits come arbitrarily close to $\mathcal{S}(G)$ with arbitrarily high probability over graphs and initializations, and every solution branch is still reached. We begin with the ideal operator.

Theorem 1 (Existence of equivariant multi-attractor dynamics)**.** *Let $\mathcal{G}$ be a permutation-invariant family of n-node graphs, and let $\mathcal{S}(G) = \{f_1(G), \dots, f_M(G)\} \subseteq Y_n = \mathbb{R}^{n\times q}$ be set-valued equivariant. Let $\mathcal{G}^\star$ consist of the graphs where the branches are locally Lipschitz and which branches coincide does not change under sufficiently small graph perturbations.*

Then there exists a single operator $T : (G, y) \mapsto T_G(y)$ that is jointly globally Lipschitz and permutation equivariant, $T_{\pi G}(\pi y) = \pi T_G(y)$. For every $G \in \mathcal{G}^\star$, the iteration $y_{t+1} = T_G(y_t)$, initialized from any distribution μ_n with a density and full support on Y_n, satisfies

$$\begin{aligned} y_t \to y_\infty \in \mathcal{S}(G) \qquad & \textit{almost surely}, \\ \mathbb{P}_{y_0\sim\mu_n}(y_\infty = s) > 0 \quad & \textit{for every } s \in \mathcal{S}(G). \end{aligned}$$

Thus a globally regular, equivariant update can support multiple solutions while individual trajectories converge (Appendix A). We next realize this behavior through a shared message-passing GNN.

Message-passing GNNs. For a graph $G = (A, X)$ and state $y \in Y_n$, define a GNN block $\mathrm{GNN}_\theta(G, y)$ with finitely many layers and finite hidden dimensions:

$$
\begin{aligned}
h_i^{(0)} &= \mathrm{enc}(X_i, y_i), \\
h_i^{(\ell+1)} &= U_\ell\left(h_i^{(\ell)}, \sum_{j=1}^{n} M_\ell(h_i^{(\ell)}, h_j^{(\ell)}, A_{ji})\right), \\
[\mathrm{GNN}_\theta(G,y)]_i &= O\left(h_i^{(L)}, \sum_{j=1}^{n} R_L(h_j^{(L)})\right),
\end{aligned}
\tag{1}
$$

where $\ell = 0, \dots, L-1$ and $i = 1, \dots, n$. The encoder and component maps M_ℓ, U_ℓ, R_L, O are arbitrary continuous maps between finite-dimensional Euclidean spaces, shared across nodes. The edge features A_{ji} encode edge attributes and edge presence; messages can be chosen to vanish on absent edges. Our theory uses this nonparametric formulation, which leaves the component maps unspecified and encompasses many concrete message-passing architectures.

Theorem 2 (Approximate realization by a recurrent MP-GNN). *Let $\mathcal{S}$ and $\mathcal{G}^\star$ be as in Theorem 1, and assume $q \geq 2$. Let G follow a tight Borel probability on $\mathcal{G}^\star$, and independently draw $y_0 \sim \mu_n$, where μ_n has a density and full support on Y_n. Let T_G be the ideal operator in Theorem 1.*

For any $\varepsilon, \rho > 0$ and $\delta \in (0,1)$, there exists a single GNN of the form (1) such that, with probability at least $1-\delta$ over (G, y_0), the iteration $y_{t+1} = \mathrm{GNN}_\theta(G, y_t)$ converges to a limit y_∞ satisfying

$$
\sup_{t \geq 0} \left\| \mathrm{GNN}_\theta(G, y_t) - T_G(y_t) \right\| < \rho. \quad \textit{and} \quad \mathrm{dist}(y_\infty, \mathcal{S}(G)) < \varepsilon.
$$

Moreover, every solution branch is reached within ε with positive joint probability:

$$
\mathbb{P}(y_t \to y_\infty,\ \|y_\infty - f_a(G)\| < \varepsilon) > 0, \qquad a = 1, \dots, M.
$$

This result requires more than standard GNN universality. Vanilla GNNs cannot universally approximate arbitrary continuous equivariant operators (Xu et al., 2018). Random node initialization can overcome this limitation by providing ***unique node identifiers*** with high probability (Abboud et al., 2020). In recurrent dynamics, however, subsequent states depend on the graph and one another, even when y_0 has i.i.d. node states. Do these evolving states still distinguish nodes?

Our proof establishes that independence is unnecessary: the ideal trajectories $z_t := T_G^t(y_0)$ retain distinct node states at every finite time,

$$
\mathbb{P}(z_{t,i} \neq z_{t,j} \text{ for all } i \neq j \text{ and all } t \in \mathbb{N}_0) = 1.
$$

Thus, each node's evolving state $z_{t,i}$ can serve as a unique identifier at every finite step. This property enables a recurrent message-passing block to approximate the ideal dynamics; see Appendix B.

2.2 Toy Example: Single-Source Graph Diffusion

On a problem with exactly three solutions, we test whether one trained recurrent GNN approaches different solutions from different random initial states and recovers all three across runs. We compare against contraction and single-target supervision.

Problem. The input is a three-node undirected graph $G = (A, 0)$ with independent edge weights $w_{12}, w_{23}, w_{31} \sim \mathrm{Unif}[0.5, 1.5]$ and no distinguishing static node features. Here A is the symmetric weighted adjacency matrix with zero diagonal. Define $L_G = \mathrm{diag}(A\mathbf{1}) - A$ and $H_G = I_3 + L_G$. We seek a source indicator $b \in \{0,1\}^3$ selecting exactly one node, $\mathbf{1}^\top b = 1$, and its diffusion field $u \in \mathbb{R}^3$ satisfying $H_G u = b$. Since H_G is positive definite, each source choice determines one diffusion field, giving exactly three solutions:

$$
\mathcal{S}(G) = \left\{ [H_G^{-1} e_k \ \ e_k] : k = 1, 2, 3 \right\} \subset \mathbb{R}^{3\times 2}, \tag{2}
$$

where e_k is the kth standard basis vector. The source is an output to discover, not an input supplied to the model. Figure 2 illustrates one graph and its three solutions.

Model and training. The recurrent state $y_t = [u_t \ b_t] \in \mathbb{R}^{3\times 2}$ stores two values per node. From six independent standard Gaussian entries, we iterate $y_{t+1} = F_\theta(G, y_t)$ using a shared block with two message-passing layers, following (1). For each fixed graph, we seek *trajectories approaching a member of $\mathcal{S}(G)$, with positive probability of reaching each solution; fresh initial states can reveal different solutions without changing the graph or model.* We train using the energy function

$$
E_G(u, b) = \frac{1}{3}\|H_G u - b\|_2^2 + \sum_{i=1}^{3} b_i(1 - b_i) + (\mathbf{1}^\top b - 1)^2, \qquad b \in [0,1]^3. \tag{3}
$$

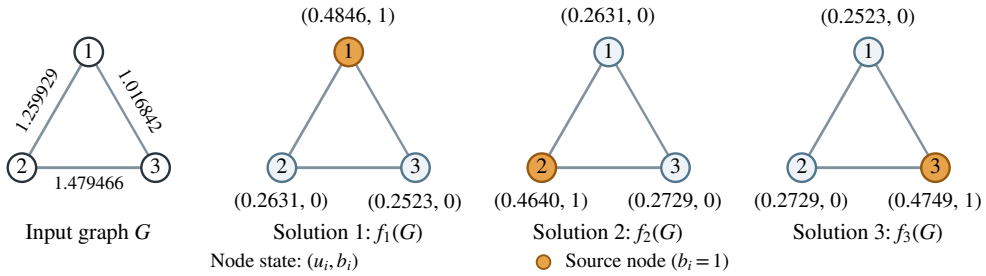


Figure 2: One graph, three valid solutions to single-source diffusion. Left: the input graph, with edge weights labeled. Right: each choice of source gives a solution $f_k(G) = [H_G^{-1} e_k \;\; e_k]$. Node labels show (u_i, b_i), with diffusion values rounded to four decimal places; orange marks the selected source ($b_i = 1$). All three outputs are valid for the same input, which specifies no source.

The three terms are all nonnegative and $E_G(u, b) = 0$ holds exactly on $\mathcal{S}(G)$. Training uses eight random starts per graph and minimizes the terminal energy $\mathbb{E}_{G,y_0}[E_G(u_T, b_T)]$ with $T = 10$. Such an unsupervised training allows the model to automatically explore/discover multiple solutions.

Baselines. We compare two controls with the same architecture, changing either the update constraint or training target. The *contractive GNN* uses the same energy objective but projects its parameters throughout training to enforce $\mathrm{Lip}_y(F_\theta(G, \cdot)) \le 0.9$, ensuring a unique fixed point for each graph. The *single-target-supervised GNN* has no contraction constraint. For each training graph, we uniformly select one of its three exact solutions and retain it across all initializations and training visits. Squared-error training asks all starts to predict that target without guaranteeing a unique equilibrium. Details are in Appendix D.

Results. Figure 3 tracks the source coordinates b_t of 128 trajectories on one held-out graph, with identical initial states across all three models. The multi-attractor model forms three clusters near the valid source indicators e_1, e_2, e_3, whereas the supervised and contractive controls each concentrate near a single point, visibly separated from all three. The diffusion coordinates u_t exhibit the same qualitative pattern (Appendix D, Figure 4). Thus energy training can produce multiple attractors near different solutions, consistent with Theorems 1 and 2, while contraction or single-target supervision can concentrate trajectories far from every valid solution.

2.3 Can a Unique Equilibrium Miss Every Valid Solution in Theory?

Do these failures reflect training or a structural limitation? A unique equilibrium cannot represent multiple solutions through different initializations, but can it always recover one valid solution? The following theorem answers no for continuous, permutation-equivariant, globally contractive updates: even with independently sampled edge weights, their equilibrium remains separated from every valid solution with positive probability, regardless of training or initialization.

Theorem 3. *Let $\mathcal{G}_\triangle$ be the family of weighted triangles in (2), with edge weights in $[w_-, w_+]$, where $0 < w_- < w_+ < \infty$. Let ν be the graph law induced by sampling the three edge weights independently and uniformly from $[w_-, w_+]$. Consider any jointly continuous, permutation-equivariant operator $F : \mathcal{G}_\triangle \times Y_3 \to Y_3$, where $Y_3 = \mathbb{R}^{3\times 2}$, satisfying*

$$\|F_G(y) - F_G(z)\|_* \le \kappa \|y - z\|_* \qquad (G \in \mathcal{G}_\triangle,\ y, z \in Y_3) \tag{4}$$

for some fixed norm $\|\cdot\|_*$ and $\kappa \in [0, 1)$. This includes any MP-GNN of (1) with continuous component maps satisfying (4), regardless of its finite depth and width.

For every $0 < \varepsilon < \sqrt{2/3}$, there exists $\delta_{F,\varepsilon} > 0$ such that, under any joint law of (G, Y_0) with graph marginal ν, the recurrence $Y_{t+1} = F_G(Y_t)$ converges to a limit Y_∞ and

$$\mathbb{P}\big(\mathrm{dist}\big(Y_\infty, \mathcal{S}(G)\big) > \varepsilon\big) \ge \delta_{F,\varepsilon},$$

where distance is measured in the Euclidean norm. In particular, no such operator converges to a valid solution almost surely over independently weighted graphs.

Random initialization distinguishes nodes even on symmetric graphs; Figure 3 shows its influence merely in the early iterations. Contraction eventually erases these differences: all trajectories approach one equilibrium. On the symmetric triangle, equivariance forces this equilibrium to preserve

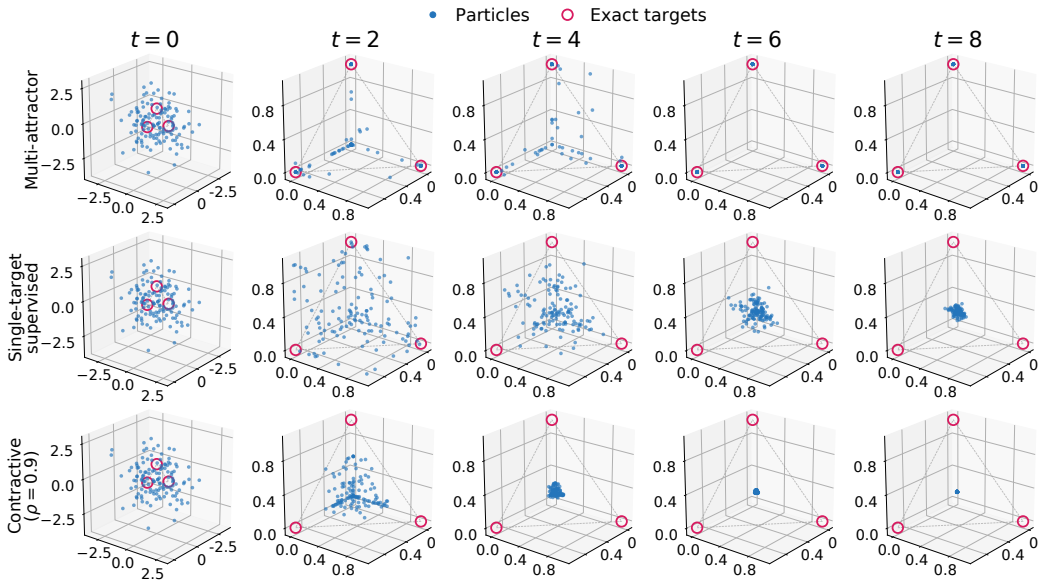


Figure 3: Different initial states can approach different solutions under the same learned update. Each blue point represents one of 128 trajectories, plotted in source coordinates $((b_t)_1, (b_t)_2, (b_t)_3)$; magenta circles mark the three valid source indicators e_1, e_2, e_3. Columns show recurrent steps $t = 0, 2, 4, 6, 8$. The multi-attractor model (top) forms three clusters near these targets, whereas the single-target-supervised (middle) and contractive (bottom) controls each concentrate away from them. All rows use the same held-out graph (test graph 0), training seed 1, and initial states.

the graph's symmetries, preventing single-source selection. For independently weighted graphs, the equilibrium need not be symmetric, but its continuous dependence on the graph extends the failure to nearby graphs with positive probability. The proof and extension are given in Appendix C.

3 Scientific Applications

We now examine whether training without solution labels yields convergent dynamics and multiple high-quality solutions in three scientific tasks.

3.1 Ising Models

Problem. The Ising model is a classical model of interacting magnetic spins and is a standard discrete optimization problem. In the zero-field setting, each node of the weighted graph $G = (V, E, J)$ carries a spin $s_i \in \{-1, +1\}$, with energy and ground states

$$\mathcal{S}(G) = \underset{s \in \{-1,+1\}^{|V|}}{\arg\min} E_G(s), \qquad E_G(s) = -\sum_{\{i,j\} \in E} J_{ij} s_i s_j. \tag{5}$$

We use open triangular lattices with antiferromagnetic couplings ($J_{i,j} < 0$), which favor opposite neighboring spins. These preferences conflict on triangles, creating competing low-energy configurations; global spin reversal also preserves energy. This multiplicity naturally tests set-valued prediction: we seek diverse low-energy configurations per graph rather than one arbitrarily selected target.

Models and baselines. We train the shared recurrent update $y_{t+1} = \mathrm{GNN}_\theta(G, y_t)$ from $y_0 \sim \mathcal{N}(0, I)$, refining continuous spin estimates and applying the sign function to y_T to predict discrete spins $\hat{s}$. Training minimizes $\mathbb{E}[E_G(y_T)]$ with $T = 10$ shared updates, encouraging low energy without prescribing a solution for each initialization. Testing uses 200 updates.

We compare four baseline groups. *(A) Single-target supervision* minimizes MSE to one Gurobi solution per graph using the same recurrent architecture, testing whether a designated target impedes accuracy and diversity. *(B) Unique-equilibrium models* are energy-trained IGNNs (Gu et al., 2020) with contraction constraints and increasing hidden widths, testing whether uniqueness compromises quality and multiplicity even with greater capacity. *(C) Feedforward models* stack 1, 3, or

Table 1: Ising results on 500 test graphs with 20 candidates per graph (mean $\pm$ standard deviation over three training seeds). $\bar{e}$: mean energy per node (lower is better); Hit10: percentage of candidates within 10% of the certified minimum energy; C_{200}: percentage of trajectories converged by step 200; D_{20}: mean number of distinct optimal configurations found per graph; G_{20}: percentage of graphs with at least two such configurations, up to numerical tolerance. Details in Appendix E.

Method	$\bar{e}\downarrow$	Hit10 $\uparrow$	$C_{200}\uparrow$	$D_{20}\uparrow$	$G_{20}\uparrow$
Ours	-1.0794 ± 0.0028	83.98 ± 0.93	99.96 ± 0.02	2.16 ± 0.23	68.73 ± 5.19
Single-target Supervision	0.2077 ± 0.5878	9.45 ± 4.37	10.11 ± 3.38	0.10 ± 0.05	0.80 ± 0.99
IGNN adapter (H16)	-0.4581 ± 0.2538	5.20 ± 3.31	100.00 ± 0.00	0.03 ± 0.02	0.00 ± 0.00
IGNN adapter (H32)	-0.7516 ± 0.0185	8.00 ± 0.71	100.00 ± 0.00	0.06 ± 0.00	0.00 ± 0.00
IGNN adapter (H64)	-0.7954 ± 0.0143	13.20 ± 2.14	100.00 ± 0.00	0.08 ± 0.01	0.00 ± 0.00
Feedforward ($\times 1$)	-0.9720 ± 0.0035	36.13 ± 0.75	–	1.31 ± 0.03	36.67 ± 1.23
Feedforward ($\times 3$)	-1.0515 ± 0.0016	69.07 ± 0.66	–	0.95 ± 0.95	22.40 ± 31.68
Feedforward ($\times 10$)	-1.0538 ± 0.0012	70.93 ± 1.52	–	0.28 ± 0.01	0.00 ± 0.00
Mean-field GNN	-0.9736 ± 0.0026	36.71 ± 0.78	–	1.34 ± 0.06	37.80 ± 1.31
Annealed Bernoulli GNN	-0.9766 ± 0.0011	37.71 ± 0.86	–	1.37 ± 0.03	38.27 ± 0.66
DIFUSCO (matched)	-0.9033 ± 0.0023	37.64 ± 0.23	–	1.56 ± 0.03	45.87 ± 0.09
DIFUSCO (large)	-1.0362 ± 0.0024	69.82 ± 1.64	–	2.60 ± 0.10	68.00 ± 2.95
VAG (matched)	-0.8893 ± 0.0075	17.18 ± 2.02	–	0.65 ± 0.11	17.40 ± 4.98
VAG (large)	-0.8814 ± 0.0547	17.34 ± 5.69	–	0.56 ± 0.17	14.00 ± 5.74

10 independently parameterized copies of our block with the same energy objective, testing whether depth and parameters can replace shared recurrence. *(D) Distributional models* learn sampling distributions or generative processes, whereas ours learns a shared update with initialization-dependent solutions. We compare quality and diversity with mean-field GNNs, annealed Bernoulli GNNs, DIFUSCO, and VAG-CO (Karalias & Loukas, 2020; Sun et al., 2022; Sun & Yang, 2023; Sanokowski et al., 2023) as alternative approaches to set-valued prediction. Appendix E provides all details.

Quality and multiple solutions. Our model achieves the lowest mean energy and highest Hit10 (Table 1): 83.98% of candidates are within 10% of optimum. It finds 2.16 distinct optima per graph on average and at least two on 68.73% of graphs. Numerical convergence reaches 99.96% by step 200 and 100% by step 500 for the same trajectories (Appendix E). Single-target supervision has substantially poorer quality and diversity; IGNN converges reliably but returns one configuration per graph with low Hit10. Deeper untied models improve quality but lag in Hit10 and distinct-optimum discovery. Large DIFUSCO finds more optima ($D_{20} = 2.60$), but has worse mean energy despite approximately 606 times as many parameters. Consistent with our theory, *one shared update trained on energy can converge to multiple high-quality solutions*, offering a favorable combination of convergence, quality, and diversity over single-target training or enforced uniqueness in these experiments.

3.2 Structural module detection in protein graphs

Problem. We seek structural modules in protein graphs: groups more internally connected than expected from their node degrees. We replace graph classification on the 1,113 PROTEINS graphs (Borgwardt et al., 2005; Morris et al., 2020) with instance-wise modularity maximization (Newman, 2006). For an undirected graph $G = (V, E)$ with node degree d_i and node assignments $z_i \in \{1, \dots, K\}$, the objective and solution set are

$$Q_G(z) = \frac{1}{|E|} \sum_{\{i,j\} \in E} \mathbf{1}[z_i = z_j] - \sum_{k=1}^{K} \left(\frac{\sum_i d_i \mathbf{1}[z_i = k]}{2|E|} \right)^2, \quad \mathcal{S}(G) = \underset{z \in \{1,\dots,K\}^{|V|}}{\arg\max} Q_G(z).$$

We allow at most four modules ($K = 4$, with empty modules permitted). Partitions can have similar energies even modulo module-label permutations, motivating distinct low-energy predictions without one selected target.

Models and baselines. Our shared recurrent update $P_{t+1} = \mathrm{GNN}_\theta(G, P_t)$ refines a randomly initialized *soft partition*, where $P_{t,i} \in \mathbb{R}^K$ contains node i's assignment weights over K modules.

Table 2: Structural module detection on 111 PROTEINS test graphs with 20 candidates per graph. $\overline{Q}_G$ averages modularity over candidates and then graphs. For optimal modularity $Q_G^\star$, Hit10 is the percentage of candidates with relative gap $(Q_G^\star - Q_G(z))/|Q_G^\star| \leq 0.10$. D_{20}^{10}: mean number of distinct partitions per graph with absolute gap $Q_G^\star - Q_G(z) \leq 0.10$; G_{20}^{10}: percentage of graphs with at least two such partitions; C_{200}: percentage of trajectories numerically converged at $t = 200$. Results: mean $\pm$ standard deviation over training seeds 0/1/2 on the same test set. Full protocol in Appendix F.

Method	$\overline{Q}_G \uparrow$	Hit10 $\uparrow$	$C_{200} \uparrow$	$D_{20}^{10} \uparrow$	$G_{20}^{10} \uparrow$
Ours	0.5560 ± 0.0024	77.49 ± 2.06	88.05 ± 2.23	5.83 ± 0.38	80.78 ± 0.42
Single-target Supervision	0.4911 ± 0.0114	41.28 ± 7.16	50.81 ± 10.84	1.11 ± 0.08	33.03 ± 2.25
IGNN (H16)	0.4147 ± 0.0079	8.11 ± 2.65	100.00 ± 0.00	0.20 ± 0.04	0.00 ± 0.00
IGNN (H32)	0.4196 ± 0.0044	11.71 ± 2.21	100.00 ± 0.00	0.22 ± 0.03	0.00 ± 0.00
IGNN (H64)	0.4235 ± 0.0121	8.41 ± 2.36	100.00 ± 0.00	0.22 ± 0.06	0.00 ± 0.00
Feedforward ($\times 1$)	0.4436 ± 0.0048	16.52 ± 2.22	–	1.35 ± 0.23	30.93 ± 6.26
Feedforward ($\times 2$)	0.4846 ± 0.0156	29.50 ± 3.61	–	2.09 ± 0.26	53.15 ± 6.62
Feedforward ($\times 3$)	0.5088 ± 0.0108	39.92 ± 5.17	–	2.95 ± 0.62	71.47 ± 9.71
Feedforward ($\times 4$)	0.5285 ± 0.0022	51.35 ± 1.68	–	4.24 ± 0.51	82.88 ± 6.62
Feedforward ($\times 5$)	0.5305 ± 0.0044	54.97 ± 3.59	–	4.32 ± 0.47	78.68 ± 4.49
Mean-field GNN	0.4020 ± 0.0122	8.53 ± 2.01	–	0.69 ± 0.05	16.22 ± 0.00
Annealed Bernoulli GNN	0.4131 ± 0.0103	9.91 ± 2.35	–	0.80 ± 0.01	18.32 ± 1.53
DIFUSCO (matched)	0.5115 ± 0.0127	41.85 ± 7.97	–	8.47 ± 1.20	97.30 ± 1.27
DIFUSCO (large)	0.5515 ± 0.0014	74.49 ± 1.66	–	11.02 ± 0.03	97.00 ± 0.42
VAG (matched)	0.3952 ± 0.0024	8.05 ± 0.14	–	0.57 ± 0.01	12.01 ± 0.85
VAG (large)	0.2577 ± 0.0237	0.44 ± 0.31	–	0.35 ± 0.15	7.21 ± 1.47

After T iterations, each node selects its highest-weight module, $z_i = \arg\max_k (P_{T,i})_k$. Training minimizes $\mathbb{E}[e_G(P_T)]$ with $T = 5$ over training graphs and initializations, where e_G is a differentiable soft-partition energy derived from modularity. At test time, we apply the learned update for 200 steps from 20 independent initializations per graph.

The four baseline groups (Section 3.1) test training objectives, equilibrium uniqueness, weight sharing, and alternative solution generation. Appendix F details the soft energy, architectures, training, and evaluation protocols.

Results. Our model has the highest mean modularity (0.5560) and Hit10 (77.49%) in Table 2, finding 5.83 distinct qualified partitions per graph on average and at least two on 80.78% of graphs. Single-target supervision and unique-equilibrium IGNNs have poorer quality and diversity. Untied depth improves performance, but five independent blocks still lag in mean modularity, Hit10, and partition discovery. Large DIFUSCO finds more qualified partitions ($D_{20}^{10} = 11.02$) but has lower mean modularity with approximately 155 times as many parameters. Together with the convergence results in Appendix F (Table 10), these findings support our theoretical premise: energy training without prescribed targets lets one shared update recover multiple high-quality solutions from different initializations.

3.3 Chemical reaction-network multistationarity

Problem. A chemical reaction network has s species i and R reactions r. The matrices $\alpha, \beta \in \mathbb{Z}_{\geq 0}^{R \times s}$ give the numbers of molecules of species i consumed (α_{ri}) and produced (β_{ri}) by reaction r; $\kappa \in \mathbb{R}_{>0}^{R}$ contains rate constants. The graph $G = (\alpha, \beta, \kappa)$ has one node per species and reaction, connected bidirectionally when $\alpha_{ri} + \beta_{ri} > 0$. Edges carry $(\alpha_{ri}, \beta_{ri})$; reaction nodes encode $\log \kappa_r$. Under mass-action kinetics, the species concentrations $c \in \mathbb{R}_{>0}^{s}$ evolve according to

$$\dot{c} = f_G(c) = N v(c), \qquad N_{ir} = \beta_{ri} - \alpha_{ri}, \qquad v_r(c) = \kappa_r \prod_{i=1}^{s} c_i^{\alpha_{ri}}.$$

We seek stationary solutions $\mathcal{S}(G) = \{c \in \mathbb{R}_{>0}^{s} : f_G(c) = 0\}$. Multiple positive solutions for one G constitute multistationarity, a set-valued prediction task. The solver seeks roots of f_G without requiring physical stability under the chemical ODE. Our kinetic instances use published network structures Yao et al. (2025), with two numerically verified stationary witnesses per instance.

Table 3: Chemistry results on 236 test graphs with 20 candidates per graph. $\overline{E}_{\mathrm{res}}$: mean Huber penalty on the mass-action residual (lower is better); $\mathrm{Hit}_{10^{-3}}$: percentage of final states with normalized physical residual $R \leq 10^{-3}$; C: percentage of recurrent trajectories with next-step log-concentration RMS change below 10^{-3} at $T = 200$; D_{20}^{raw}: mean number of distinct accepted prediction clusters per graph; G_{20}^{raw}: percentage of graphs with at least two. Entries: mean $\pm$ population standard deviation over training seeds 0/1/2.

Method	$\overline{E}_{\mathrm{res}} \downarrow$	$\mathrm{Hit}_{10^{-3}} \uparrow$	$C \uparrow$	$D_{20}^{\mathrm{raw}} \uparrow$	$G_{20}^{\mathrm{raw}} \uparrow$
Ours	0.0188 ± 0.0046	83.77 ± 3.44	91.99 ± 0.35	3.506 ± 0.314	41.67 ± 17.01
Single-target Supervis.	0.0373 ± 0.0059	18.78 ± 3.88	88.48 ± 4.29	1.371 ± 0.137	11.30 ± 0.72
IGNN (H16)	0.1214 ± 0.0823	0.00 ± 0.00	100.00 ± 0.00	0.000 ± 0.000	0.00 ± 0.00
IGNN (H32)	0.1180 ± 0.0709	0.00 ± 0.00	100.00 ± 0.00	0.000 ± 0.000	0.00 ± 0.00
IGNN (H64)	0.0960 ± 0.0056	0.00 ± 0.00	100.00 ± 0.00	0.000 ± 0.000	0.00 ± 0.00
Feedforward ($\times 1$)	0.1082 ± 0.0081	0.00 ± 0.00	—	0.000 ± 0.000	0.00 ± 0.00
Feedforward ($\times 2$)	0.0586 ± 0.0050	0.01 ± 0.02	—	0.003 ± 0.004	0.14 ± 0.20
Feedforward ($\times 3$)	0.0441 ± 0.0065	0.11 ± 0.10	—	0.021 ± 0.019	0.28 ± 0.20
Feedforward ($\times 4$)	0.0295 ± 0.0016	0.42 ± 0.35	—	0.076 ± 0.061	0.42 ± 0.35
Feedforward ($\times 5$)	0.0255 ± 0.0022	0.58 ± 0.41	—	0.110 ± 0.078	0.71 ± 0.53
DDPM / matched	0.0576 ± 0.0047	0.03 ± 0.01	—	0.006 ± 0.002	0.00 ± 0.00
DDIM / matched	0.0732 ± 0.0055	0.01 ± 0.01	—	0.003 ± 0.002	0.00 ± 0.00
DIS / matched	0.9315 ± 0.0007	0.00 ± 0.00	—	0.000 ± 0.000	0.00 ± 0.00
DDPM / large	0.1197 ± 0.0040	0.01 ± 0.01	—	0.003 ± 0.002	0.00 ± 0.00
DDIM / large	0.1317 ± 0.0030	0.00 ± 0.00	—	0.000 ± 0.000	0.00 ± 0.00
DIS / large	0.0463 ± 0.0003	0.02 ± 0.02	—	0.004 ± 0.003	0.00 ± 0.00

Models and baselines. Species log concentrations follow the shared update $z_{t+1} = F_\theta(G, z_t)$, with output $c_T = \exp z_T$. Training minimizes equation residuals without prescribed targets, using randomized unrolls of at most 32 steps; the main comparison uses $T = 200$. The four baseline groups (Section 3.1) are single-target MSE, physics-trained contractive IGNNs, untied feedforward depth controls, and distributional models. For continuous states, the latter comprise DDPM, DDIM, and DIS with matched and larger upstream-recipe graph adaptations; details are in Appendix G.

Results. Our model achieves 83.77% candidate hits and 3.506 accepted prediction clusters per graph on average, with at least two on 41.67% of graphs (Table 3). Numerical settling increases from 91.99% at $T = 200$ to 96.15% at $T = 5{,}000$; remaining oscillations and unresolved cases appear in Table 13. Single-target supervision reaches only 18.78% hits; IGNNs settle in their hidden-state operators but yield no accepted predictions, illustrating that convergence alone does not ensure validity. Untied depth improves accuracy, but five independent stages reach only 0.58% hits; tested distributional models also have low hit rates. These results support our theoretical premise: on this benchmark, one shared update trained on residuals without prescribed targets combines accuracy, diversity, and numerical convergence across initializations.

4 Conclusions

Our results establish multi-attractor dynamics as a mechanism for representing set-valued graph tasks with standard recurrent, weight-tied GNNs. Under the stated assumptions, one shared update can converge arbitrarily close to different valid solutions, while enforcing uniqueness through contraction can obstruct recovery of even one valid solution. Experiments on three scientific tasks show that training on problem-specific energies or residuals can produce numerically convergent trajectories and multiple high-quality predictions without solution labels. These findings motivate preserving access to multiple equilibria when the task admits multiple answers. Our work provides a general framework for learning multiple solutions across graph tasks, opening opportunities for a wide range of applications; realizing this potential requires suitable objectives and effective optimization tailored to each problem. Understanding how training shapes convergence and solution coverage is an important direction for future work.

AI USE STATEMENT

We used generative AI tools to assist with drafting and polishing the manuscript, retrieving and identifying related literature, and supporting research ideation and execution. The authors made the final research and editorial decisions and take full responsibility for the paper's content, including its theoretical claims, experimental results, interpretations, and citations.

REFERENCES

Ralph Abboud, Ismail Ilkan Ceylan, Martin Grohe, and Thomas Lukasiewicz. The surprising power of graph neural networks with random node initialization. *arXiv preprint arXiv:2010.01179*, 2020.

Lada A Adamic and Eytan Adar. Friends and neighbors on the web. *Social networks*, 25(3):211–230, 2003.

Cem Anil, Ashwini Pokle, Kaiqu Liang, Johannes Treutlein, Yuhuai Wu, Shaojie Bai, J Zico Kolter, and Roger B Grosse. Path independent equilibrium models can better exploit test-time computation. *Advances in Neural Information Processing Systems*, 35:7796–7809, 2022.

Waiss Azizian and marc lelarge. Expressive power of invariant and equivariant graph neural networks. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=lxHgXYN4bwl.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. Deep equilibrium models. *Advances in neural information processing systems*, 32, 2019.

Justin Baker, Qingsong Wang, Cory D Hauck, and Bao Wang. Implicit graph neural networks: A monotone operator viewpoint. In *International conference on machine learning*, pp. 1521–1548. PMLR, 2023.

Arpit Bansal, Avi Schwarzschild, Eitan Borgnia, Zeyad Emam, Furong Huang, Micah Goldblum, and Tom Goldstein. End-to-end algorithm synthesis with recurrent networks: Extrapolation without overthinking. *Advances in Neural Information Processing Systems*, 35:20232–20242, 2022.

Julius Berner, Lorenz Richter, and Karen Ullrich. An optimal control perspective on diffusion-based generative modeling. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=oYIjw37pTP.

Karsten M. Borgwardt, Cheng Soon Ong, Stefan Schönauer, S. V. N. Vishwanathan, Alex J. Smola, and Hans-Peter Kriegel. Protein function prediction via graph kernels. *Bioinformatics*, 2005.

Paul Breiding and Sascha Timme. HomotopyContinuation.jl: A package for homotopy continuation in Julia. In *Mathematical Software – ICMS 2018*, volume 10931 of *Lecture Notes in Computer Science*, 2018.

Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=F72ximsx7C1.

Chen Cai and Yusu Wang. A simple yet effective baseline for non-attributed graph classification. *arXiv preprint arXiv:1811.03508*, 2018.

Qian Chen, Tianjian Zhang, Linxin Yang, Qingyu Han, Akang Wang, Ruoyu Sun, Xiaodong Luo, and Tsung-Hui Chang. Symilo: A symmetry-aware learning framework for integer linear optimization. *Advances in Neural Information Processing Systems*, 37:24411–24434, 2024.

Zhengdao Chen, Soledad Villar, Lei Chen, and Joan Bruna. On the equivalence between graph isomorphism testing and function approximation with gnns. *Advances in neural information processing systems*, 32, 2019.

Kyunghyun Cho, Bart Van Merriënboer, Çağlar Gulçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1724–1734, 2014.

Gabriele Corso, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Veličković. Principal neighbourhood aggregation for graph nets. *Advances in neural information processing systems*, 33:13260–13271, 2020.

Hanjun Dai, Zornitsa Kozareva, Bo Dai, Alex Smola, and Le Song. Learning steady-states of iterative algorithms over graphs. In *International conference on machine learning*, pp. 1106–1114. PMLR, 2018.

Vijay Prakash Dwivedi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Graph neural networks with learnable structural and positional representations. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=wTTjnvGphYj.

Nadav Dym, Hannah Lawrence, and Jonathan W. Siegel. Equivariant frames and the impossibility of continuous canonicalization. In *International Conference on Machine Learning*, 2024.

Patrick E. Farrell, Ásgeir Birkisson, and Simon W. Funke. Deflation techniques for finding distinct solutions of nonlinear partial differential equations. *SIAM Journal on Scientific Computing*, 37 (4):A2026–A2045, 2015.

Floris Geerts and Juan L Reutter. Expressiveness and approximation properties of graph neural networks. *arXiv preprint arXiv:2204.04661*, 2022.

Abhinav Goel, Derek Lim, Hannah Lawrence, Stefanie Jegelka, and Ningyuan Huang. Any-subgroup equivariant networks via symmetry breaking. *arXiv preprint arXiv:2603.19486*, 2026.

Lukas Gonon, Thilo Meyer-Brandis, and Niklas Weber. Universality and approximation rates of graph neural networks with random features. *arXiv preprint arXiv:2607.26699*, 2026.

Fangda Gu, Heng Chang, Wenwu Zhu, Somayeh Sojoudi, and Laurent El Ghaoui. Implicit graph neural networks. *Advances in neural information processing systems*, 33:11984–11995, 2020.

Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2026. URL https://www.gurobi.com.

Aric Hagberg, Pieter J Swart, and Daniel A Schult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Laboratory (LANL), 2007.

Fleur Hendriks, Ondřej Rokoš, Martin Doškář, Marc GD Geers, and Vlado Menkovski. Equivariant flow matching for symmetry-breaking bifurcation problems. *arXiv preprint arXiv:2509.03340*, 2025.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

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.

Caleb Jore and Jialin Liu. Bifurcation models: Learning set-valued solution maps with weight-tied dynamics. *arXiv preprint arXiv:2605.07277*, 2026.

Sékou-Oumar Kaba and Siamak Ravanbakhsh. Symmetry breaking and equivariant neural networks. *arXiv preprint arXiv:2312.09016*, 2023.

Nikolaos Karalias and Andreas Loukas. Erdos goes neural: an unsupervised learning framework for combinatorial optimization on graphs. *Advances in Neural Information Processing Systems*, 33: 6659–6672, 2020.

Nicolas Keriven and Gabriel Peyré. Universal invariant and equivariant graph neural networks. *Advances in neural information processing systems*, 32, 2019.

Hannah Lawrence, Vasco Portilheiro, Yan Zhang, and Sékou-Oumar Kaba. Improving equivariant networks with probabilistic symmetry breaking. In *International Conference on Learning Representations*, 2025.

Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*, 2015.

Zhuwen Li, Qifeng Chen, and Vladlen Koltun. Combinatorial optimization with graph convolutional networks and guided tree search. *Advances in neural information processing systems*, 31, 2018.

David Liben-Nowell and Jon Kleinberg. The link prediction problem for social networks. In *Proceedings of the twelfth international conference on Information and knowledge management*, pp. 556–559, 2003.

Jialin Liu, Lisang Ding, Stanley J Osher, and Wotao Yin. Expressive power of implicit models: Rich equilibria and test-time scaling. In *International Conference on Learning Representations*, volume 2026, pp. 17164–17212, 2026.

Juncheng Liu, Kenji Kawaguchi, Bryan Hooi, Yiwei Wang, and Xiaokui Xiao. Eignn: Efficient infinite-depth graph neural networks. *Advances in Neural Information Processing Systems*, 34: 18762–18773, 2021.

Andreas Loukas. What graph neural networks cannot learn: depth vs width. *arXiv preprint arXiv:1907.03199*, 2019.

Haggai Maron, Heli Ben-Hamu, Hadar Serviansky, and Yaron Lipman. Provably powerful graph networks. *Advances in neural information processing systems*, 32, 2019.

Christopher Morris, Martin Ritzert, Matthias Fey, William L Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 4602–4609, 2019.

Christopher Morris, Nils M Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. Tudataset: A collection of benchmark datasets for learning with graphs. *arXiv preprint arXiv:2007.08663*, 2020.

James R. Munkres. *Topology*. Pearson Education Limited, Harlow, England, 2 edition, 2014. ISBN 978-1-292-02362-5.

Yatin Nandwani, Deepanshu Jindal, Parag Singla, et al. Neural learning of one-of-many solutions for combinatorial problems in structured output spaces. *arXiv preprint arXiv:2008.11990*, 2020.

Mark EJ Newman. Modularity and community structure in networks. *Proceedings of the national academy of sciences*, 103(23):8577–8582, 2006.

Junyoung Park, Jinhyun Choo, and Jinkyoo Park. Convergent graph solvers. *arXiv preprint arXiv:2106.01680*, 2021.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Thomas Adler, Lukas Gruber, Markus Holzleitner, Milena Pavlović, Geir Kjetil Sandve, et al. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*, 2020.

Federico Ricci-Tersenghi. The bethe approximation for solving the inverse ising problem: a comparison with other inference methods. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(08):P08015, 2012.

Eran Rosenbluth and Martin Grohe. Repetition makes perfect: Recurrent graph neural networks match message passing limit. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

Sebastian Sanokowski, Wilhelm Berghammer, Sepp Hochreiter, and Sebastian Lehner. Variational annealing on graphs for combinatorial optimization. *Advances in Neural Information Processing Systems*, 36:63907–63930, 2023.

Ryoma Sato, Makoto Yamada, and Hisashi Kashima. Random features strengthen graph neural networks. In *Proceedings of the 2021 SIAM international conference on data mining (SDM)*, pp. 333–341. SIAM, 2021.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

Martin JA Schuetz, J Kyle Brubaker, and Helmut G Katzgraber. Combinatorial optimization with physics-inspired graph neural networks. *Nature Machine Intelligence*, 4(4):367–377, 2022.

Daniel Selsam, Matthew Lamm, Benedikt Bünz, Percy Liang, Leonardo de Moura, and David L Dill. Learning a sat solver from single-bit supervision. *arXiv preprint arXiv:1802.03685*, 2018.

Tess E. Smidt, Mario Geiger, and Benjamin Kurt Miller. Finding symmetry breaking order parameters with Euclidean neural networks. *Physical Review Research*, 3:L012002, 2021.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=St1giarCHLP.

Haoran Sun, Etash K Guha, and Hanjun Dai. Annealed training for combinatorial optimization on graphs. *arXiv preprint arXiv:2207.11542*, 2022.

Zhiqing Sun and Yiming Yang. Difusco: Graph-based diffusion solvers for combinatorial optimization. *Advances in neural information processing systems*, 36:3706–3731, 2023.

Hao Tang, Zhiao Huang, Jiayuan Gu, Bao-Liang Lu, and Hao Su. Towards scale-invariant graph-related problem solving by iterative homogeneous gnns. *Advances in Neural Information Processing Systems*, 33:15811–15822, 2020.

Jan Toenshoff, Martin Ritzert, Hinrikus Wolf, and Martin Grohe. Graph neural networks for maximum constraint satisfaction. *Frontiers in artificial intelligence*, 3:580607, 2021.

Petar Veličković, Rex Ying, Matilde Padovano, Raia Hadsell, and Charles Blundell. Neural execution of graph algorithms. *arXiv preprint arXiv:1910.10593*, 2019.

Ezra Winston and J Zico Kolter. Monotone operator equilibrium networks. *Advances in neural information processing systems*, 33:10718–10728, 2020.

Dian Wu, Lei Wang, and Pan Zhang. Solving statistical mechanics using variational autoregressive networks. *Physical review letters*, 122(8):080602, 2019.

YuQing Xie and Tess Smidt. Equivariant symmetry breaking sets. *Transactions on Machine Learning Research*, 2024.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

Yongyi Yang, Tang Liu, Yangkun Wang, Zengfeng Huang, and David Wipf. Implicit vs unfolded graph neural networks. *Journal of Machine Learning Research*, 26(82):1–46, 2025.

Shenghao Yao, AmirHosein Sadeghimanesh, and Matthew England. Understanding multistationarity of fully open reaction networks: S. yao, ah sadeghimanesh, m. england. *Bulletin of Mathematical Biology*, 87(12):176, 2025.

Dinghuai Zhang, Hanjun Dai, Nikolay Malkin, Aaron C Courville, Yoshua Bengio, and Ling Pan. Let the flows tell: Solving graph combinatorial problems with gflownets. *Advances in neural information processing systems*, 36:11952–11969, 2023.

Tao Zhou, Linyuan Lü, and Yi-Cheng Zhang. Predicting missing links via local information. *The European Physical Journal B*, 71(4):623–630, 2009.

A IDEAL OPERATOR EXISTENCE

Fix the number of nodes n. We represent a graph instance as

$$G = (A, X) \in \mathcal{E}_n := \mathbb{R}^{n\times n\times d_e} \times \mathbb{R}^{n\times d_v},$$

where

$$A = (A^{(1)}, \dots, A^{(d_e)})$$

contains the edge features and X contains the node features. The output space is

$$\mathcal{Y}_n = \mathbb{R}^{n\times q} \cong \mathbb{R}^m.$$

Both spaces are equipped with their Euclidean norms.

For a permutation π of the n nodes, let P_π denote its permutation matrix, with $P_\pi e_i = e_{\pi(i)}$. Relabeling the nodes of a graph and an output is defined by

$$\pi G := \left(\left(P_\pi A^{(r)} P_\pi^\top\right)_{r=1}^{d_e}, P_\pi X \right), \qquad \pi y := P_\pi y.$$

Since permutation matrices are orthogonal, these relabelings preserve Euclidean distances.

Throughout, $\mathcal{G} \subseteq \mathcal{E}_n$ is closed under node relabeling:

$$G \in \mathcal{G} \quad \Longrightarrow \quad \pi G \in \mathcal{G}$$

for every node permutation π. The domain $\mathcal{G}$ need not be bounded, closed, connected, or compact.

A.1 BASIC CONCEPTS AND MAIN RESULT

Definition 1 (Permutation-equivariant finite-branch solution map)**.** *A finite-branch solution map on $\mathcal{G}$ is represented, for some integer $M \geq 1$, by labeled branches*

$$f_1, \dots, f_M : \mathcal{G} \to \mathcal{Y}_n$$

through

$$S(G) := \{f_1(G), \dots, f_M(G)\}.$$

Distinct branch indices are allowed to have coincident values. Since $S(G)$ is a set, if $f_i(G) = f_j(G)$ for $i \neq j$, their common value appears only once in $S(G)$.

We say that this representation is permutation equivariant if, for every node permutation π, there exists a permutation τ_π of the branch indices $\{1, \dots, M\}$ such that

$$f_{\tau_\pi(i)}(\pi G) = \pi f_i(G) \qquad \text{for every } G \in \mathcal{G} \text{ and } i \in \{1, \dots, M\}.$$

Thus, relabeling the nodes may relabel the solution branches, but it does not change the underlying solution set. In particular,

$$S(\pi G) = \pi S(G) := \{\pi s : s \in S(G)\}.$$

Definition 2 (Regular and stable graph instances)**.** *For each branch, let*

$$D_i := \{G \in \mathcal{G} : f_i \text{ is not locally Lipschitz at } G\}, \qquad D := \bigcup_{i=1}^{M} D_i.$$

A graph G_0 has a stable collapse pattern *if some relatively open neighborhood V of G_0 admits a partition*

$$\{1, \dots, M\} = C_1 \,\dot\cup \cdots \dot\cup\, C_r$$

such that, for every $G \in V$,

$$f_i(G) = f_j(G) \quad \Longleftrightarrow \quad i, j \in C_\alpha \text{ for some } \alpha.$$

Let U be the set of graphs without a stable collapse pattern and define

$$\mathcal{G}_\star := \mathcal{G} \setminus (D \cup U).$$

We call $\mathcal{G}_\star$ the regular stable domain*. Both D and U are relatively closed in $\mathcal{G}$. No emptiness, finiteness, or measure-zero assumption is imposed on D.*

Remark (Equivariant branch labels). At the level of the underlying set-valued map, the fixed label permutations in Definition 1 impose no additional restriction. Indeed, given any finite representation $S(G) = \{f_1(G), \dots, f_M(G)\}$ satisfying $S(\pi G) = \pi S(G)$, define

$$\widetilde{f}_{(i,\sigma)}(G) := \sigma f_i(\sigma^{-1} G), \qquad \widetilde{\tau}_\pi(i, \sigma) := (i, \pi\sigma), \qquad \sigma \in \mathfrak{S}_n.$$

These $Mn!$ labels represent the same solution set and satisfy Definition 1. If D and $\mathcal{G}_\star$ are computed from the original labels, the symmetrized representation has

$$\widetilde{D} = \bigcup_{\sigma \in \mathfrak{S}_n} \sigma D, \qquad \widetilde{\mathcal{G}}_\star = \bigcap_{\sigma \in \mathfrak{S}_n} \sigma \mathcal{G}_\star.$$

For the second identity, a stable partition restricts to each σ-block; conversely, local continuity and separation of the distinct solution values make matches between these blocks locally constant. Thus symmetrization may shrink the regular stable domain, and probability assumptions on that domain must be rechecked. If the original representation already satisfies Definition 1, both domains are invariant by Lemma 1 and remain unchanged.

Definition 3 (Switching set and strict basins)**.** *For $G \in \mathcal{G}_\star$, the* reduced switching set *is*

$$\Sigma_G^\circ := \left\{ y \in \mathcal{Y}_n : \#\operatorname*{argmin}_{s \in S(G)} \|y - s\| \geq 2 \right\},$$

where coincident branch values are counted only once. For $s \in S(G)$, its strict Voronoi basin is

$$V_s(G) := \{ y \in \mathcal{Y}_n : \|y - s\| < \|y - r\| \text{ for every } r \in S(G) \setminus \{s\} \}.$$

Theorem 4 (Jointly globally Lipschitz equivariant multi-attractor dynamics)**.** *Let S be an equivariant finite-branch solution map in the sense of Definition 1, with regular stable domain and switching geometry given by Definitions 2 and 3. Then there exists*

$$T : \mathcal{G} \times \mathcal{Y}_n \to \mathcal{Y}_n, \qquad (G, y) \mapsto T_G(y),$$

with the following properties.

1. ***Regularity and symmetry.*** *The map T is jointly globally Lipschitz and permutation equivariant. In particular, for some $L_T < \infty$,*

$$\|T_G(y) - T_H(z)\| \leq L_T \sqrt{\|G - H\|^2 + \|y - z\|^2}, \qquad T_{\pi G}(\pi y) = \pi T_G(y). \tag{6}$$

2. ***Exceptional configurations.*** *One may choose $T_G(y) = y$ whenever $G \in D \cup U$, or whenever $G \in \mathcal{G}_\star$ and $y \in \Sigma_G^\circ$.*

3. ***Basin-wise convergence.*** *Fix $G \in \mathcal{G}_\star$ and $y_0 \notin \Sigma_G^\circ$, and let*

$$s := \operatorname*{argmin}_{r \in S(G)} \|y_0 - r\|.$$

Then $y_{t+1} = T_G(y_t)$ converges to s, and there is a trajectory-dependent constant $c(G, y_0) \in (0, 1)$ such that

$$\|y_t - s\| \leq \big(1 - c(G, y_0)\big)^t \|y_0 - s\|. \tag{7}$$

No uniform positive lower bound on $c(G, y_0)$ is claimed.

4. ***Almost-everywhere recovery.*** *For every $G \in \mathcal{G}_\star$, $\mathcal{L}^m(\Sigma_G^\circ) = 0$. Let μ be a Borel probability measure on $\mathcal{Y}_n$ with $\mu \ll \mathcal{L}^m$, and draw $y_0 \sim \mu$. Then the iteration y_t converges to $S(G)$ for μ-almost every initialization. If in addition $\mu(V_s(G)) > 0$ for every $s \in S(G)$, then*

$$\operatorname{supp} \mathcal{L}(y_\infty \mid G) = S(G), \tag{8}$$

where y_∞ denotes the almost-sure iteration limit.

A.2 BAD SET AND DISPLACEMENT FUNCTIONS

Before the proof, we state some definitions and lemmas.

Definition 4 (Joint bad set)**.** *Let $Z := \mathcal{G} \times \mathcal{Y}_n$ carry the Euclidean product metric, and define*

$$\mathbf{\Sigma}^\circ := \{(G, y) : G \in \mathcal{G}_\star,\ y \in \Sigma_G^\circ\}.$$

The joint bad set *and its complement are*

$$B := \big((D \cup U) \times \mathcal{Y}_n\big) \cup \overline{\mathbf{\Sigma}^\circ}^{\,Z}, \qquad \Omega := Z \setminus B. \tag{9}$$

Here, B contains singular and unstable graph instances and the closed joint switching graph.

Definition 5 (Selector and normalized displacement)**.** *Define $d : Z \to [0, 1]$ by*

$$d(G, y) := \begin{cases} \min\{1, \operatorname{dist}((G, y), B)\}, & B \neq \varnothing, \\ 1, & B = \varnothing. \end{cases}$$

For $(G, y) \in \Omega$, the nearest distinct solution is unique; define the branch selector P and normalized displacement H as follows:

$$P(G, y) := \operatorname*{argmin}_{s \in S(G)} \|y - s\|, \qquad H(G, y) := \frac{P(G, y) - y}{d(G, y)}. \tag{10}$$

First, let's consider the invariance/equivariance properties of the above definitions.

Lemma 1. *The bad set B and its complement Ω are both permutation invariant.*

Proof. Fix a node permutation π. Its action on $\mathcal{G}$ is an isometric bijection and therefore maps relatively open neighborhoods to relatively open neighborhoods. Definition 1 gives

$$f_{\tau_\pi(i)} = \pi \circ f_i \circ \pi^{-1}.$$

Consequently, f_i is locally Lipschitz at G if and only if $f_{\tau_\pi(i)}$ is locally Lipschitz at πG. Since τ_π is a bijection of the labels, $\pi D = D$. If V and $\{C_\alpha\}$ give a stable collapse pattern at G_0, then πV and $\{\tau_\pi(C_\alpha)\}$ give one at πG_0. Applying the same argument to π^{-1} yields $\pi U = U$. In particular, $\pi\mathcal{G}_\star = \mathcal{G}_\star$, and

$$\pi((D \cup U) \times \mathcal{Y}_n) = (D \cup U) \times \mathcal{Y}_n.$$

Moreover, solution-set equivariance and distance preservation imply $\pi(\mathbf{\Sigma}^\circ) = \mathbf{\Sigma}^\circ$. Indeed, let $(G, y) \in \mathbf{\Sigma}^\circ$. Then there exist distinct $s_1, s_2 \in S(G)$ such that

$$\|y - s_1\| = \|y - s_2\| = \operatorname{dist}(y, S(G)).$$

Since $S(\pi G) = \pi S(G)$ and the permutation action is an isometry, πs_1 and πs_2 are distinct nearest elements of $S(\pi G)$ to πy. Since $\pi G \in \mathcal{G}_\star$, it follows that

$$\pi(G, y) = (\pi G, \pi y) \in \mathbf{\Sigma}^\circ.$$

Thus

$$\pi(\mathbf{\Sigma}^\circ) \subseteq \mathbf{\Sigma}^\circ.$$

Applying the same argument to π^{-1} gives the reverse inclusion and therefore

$$\pi(\mathbf{\Sigma}^\circ) = \mathbf{\Sigma}^\circ.$$

Since the permutation mapping is a homeomorphism,

$$\pi\left(\overline{\mathbf{\Sigma}^\circ}^{\,Z}\right) = \overline{\pi(\mathbf{\Sigma}^\circ)}^{\,Z} = \overline{\mathbf{\Sigma}^\circ}^{\,Z}.$$

Using $B = ((D \cup U) \times \mathcal{Y}_n) \cup \overline{\mathbf{\Sigma}^\circ}^{\,Z}$ now gives $\pi B = B$. In addition,

$$\pi\Omega = \pi(Z \setminus B) = \pi Z \setminus \pi B = Z \setminus B = \Omega.$$

which shows the invariance of Ω and finishes the proof. □

Lemma 2. *Function d is permutation invariant, and P and H are permutation equivariant.*

Proof. Since the permutation action is an isometry and $\pi B = B$,

$$d(\pi G, \pi y) = d(G, y).$$

Let $s_\star = P(G, y)$. By solution-set equivariance,

$$S(\pi G) = \pi S(G).$$

Thus every $\widetilde{s} \in S(\pi G)$ has the form $\widetilde{s} = \pi s$ for some $s \in S(G)$. Since permutations preserve distances,

$$\|\pi y - \widetilde{s}\| = \|y - s\| \geq \|y - s_\star\| = \|\pi y - \pi s_\star\|.$$

Hence $\pi s_\star$ is a nearest solution to πy. Its uniqueness on Ω gives

$$P(\pi G, \pi y) = \pi P(G, y).$$

Finally, by Definition 5,

$$H(\pi G, \pi y) = \frac{P(\pi G, \pi y) - \pi y}{d(\pi G, \pi y)} = \pi \frac{P(G, y) - y}{d(G, y)} = \pi H(G, y),$$

which finishes the proof. □

To further establish the properties of displacement H, we state and use the following lemma.

Lemma 3 ((Liu et al., 2026, Theorem A.4)). *Let X be an arbitrary subset of a Euclidean space, equipped with the induced metric, and let $h : X \to \mathbb{R}$ be locally Lipschitz intrinsically on X. Then there exists a function $\varepsilon : X \to (0, 1)$ such that both ε and εh are globally Lipschitz on X.*

Here intrinsic local Lipschitzness means Lipschitzness on a relative neighborhood $X \cap B(x, r)$ of each $x \in X$. The cited theorem requires no openness, closedness, boundedness, or local compactness of X, so it applies to $X = \Omega$.

We first extend this result to the vector-valued and equivariant form required below. Consider a general group setting where Γ is a finite group with orthogonal representations:

$$\rho_X : \Gamma \to \mathrm{O}(p), \qquad \rho_Y : \Gamma \to \mathrm{O}(m).$$

For brevity, given $\gamma \in \Gamma$, $x \in \mathbb{R}^p$, and $v \in \mathbb{R}^m$, we write:

$$\gamma x := \rho_X(\gamma)x, \qquad \gamma v := \rho_Y(\gamma)v.$$

The previously defined permutation π is an example of such an action; it applies to both G and y, acting as an orthogonal transformation on each domain.

Lemma 4 (Equivariant bounded flattening). *Let $X \subseteq \mathbb{R}^p$ be Γ-invariant. Suppose that $H : X \to \mathbb{R}^m$ is locally Lipschitz and Γ-equivariant. Then there exists a Γ-invariant function $\beta : X \to (0, 1)$ such that both β and βH are bounded and globally Lipschitz on X. Moreover, βH is Γ-equivariant.*

Proof. Write

$$H = (H_1, \ldots, H_m).$$

For every coordinate H_j, apply Lemma 3 to obtain a function

$$\varepsilon_j : X \to (0, 1)$$

such that ε_j and

$$u_j := \varepsilon_j H_j$$

are globally Lipschitz. Define

$$a_j(x) := \frac{\varepsilon_j(x)}{1 + |u_j(x)|}.$$

The scalar map $r \mapsto (1 + |r|)^{-1}$ is bounded and globally Lipschitz. Since ε_j is also bounded and globally Lipschitz, a_j is bounded and globally Lipschitz, and

$$0 < a_j < 1.$$

Moreover,

$$a_j H_j = \frac{u_j}{1 + |u_j|}.$$

Because $r \mapsto r/(1+|r|)$ is bounded and globally Lipschitz, $a_j H_j$ is bounded and globally Lipschitz as well. Set

$$a_0(x) := \prod_{j=1}^{m} a_j(x).$$

A finite product of bounded globally Lipschitz functions is bounded and globally Lipschitz. For every coordinate j,

$$a_0 H_j = (a_j H_j) \prod_{k \neq j} a_k,$$

which is again bounded and globally Lipschitz. Hence $a_0 H$ is bounded and globally Lipschitz.

Finally, define

$$\beta(x) := \prod_{\gamma \in \Gamma} a_0(\gamma x).$$

Since X is Γ-invariant, each $a_0 \circ \gamma$ is well defined on X. Since the representation ρ_X is orthogonal,

$$\mathrm{Lip}(a_0 \circ \gamma) \leq \mathrm{Lip}(a_0), \qquad \|a_0 \circ \gamma\|_\infty \leq \|a_0\|_\infty.$$

The finiteness of Γ therefore implies that β is bounded and globally Lipschitz.

For every $\gamma_0 \in \Gamma$,

$$\beta(\gamma_0 x) = \prod_{\gamma \in \Gamma} a_0(\gamma \gamma_0 x) = \prod_{\gamma \in \Gamma} a_0(\gamma x) = \beta(x),$$

where the second equality follows because right multiplication by γ_0 permutes the elements of Γ. Thus β is Γ-invariant. Furthermore,

$$\beta(x) H(x) = a_0(x) H(x) \prod_{\substack{\gamma \in \Gamma \\ \gamma \neq e}} a_0(\gamma x),$$

which is a finite product of bounded globally Lipschitz factors. Thus βH is bounded and globally Lipschitz. Moreover, the equivariance of βH follows from the invariance of β and equivariance of H. This proves the lemma. □

A.3 Proof of the main result

Proof of Theorem 4. We construct T and verify the four asserted properties.

Step 1: The set B is closed and permutation-invariant.

For each branch, the locus on which it is locally Lipschitz is relatively open. Therefore D is relatively closed in $\mathcal{G}$. The set of points having a stable collapse pattern is also relatively open by definition, so U is relatively closed. It follows that

$$(D \cup U) \times \mathcal{Y}_n$$

is relatively closed in Z. The set $\overline{\mathbf{\Sigma}^\circ}^{\,Z}$ is closed by definition. Hence B is closed in Z.

In addition, Lemma 1 provides the permutation-invariance of the bad set B and its complement.

Step 2: The nearest-solution selector is locally Lipschitz on $Z \setminus B$.

Fix $(G_0, y_0) \in \Omega$. Then $G_0 \notin D \cup U$, the collapse pattern is stable near G_0, and y_0 has a unique nearest distinct branch value. Let $C_\star$ be the stable cluster realizing this nearest value, and choose any representative $i_\star \in C_\star$. Branches in $C_\star$ agree identically throughout some neighborhood of G_0.

If $C_\star$ is the only stable cluster, then all branch values agree throughout that neighborhood, and hence

$$P(G, y) = f_{i_\star}(G)$$

there; the local Lipschitz conclusion follows immediately.

Otherwise, choose one representative i_α from every other stable cluster. Since the nearest branch value at (G_0, y_0) is unique, the margin

$$\delta := \min_{\alpha \neq \star} \left(\|y_0 - f_{i_\alpha}(G_0)\| - \|y_0 - f_{i_\star}(G_0)\| \right)$$

is strictly positive. All regular branches are continuous near G_0. After shrinking the neighborhood of (G_0, y_0) if necessary, $C_\star$ remains the unique nearest cluster throughout that neighborhood. Hence

$$P(G, y) = f_{i_\star}(G)$$

there. Because $G_0 \notin D$, the representative branch $f_{i_\star}$ is locally Lipschitz near G_0. Thus P is jointly locally Lipschitz near (G_0, y_0). Since (G_0, y_0) was arbitrary,

$$P : \Omega \to \mathcal{Y}_n$$

is jointly locally Lipschitz. This is the same local mechanism used in (Jore & Liu, 2026, Lemma A.12), but no boundedness or compactness of the graph domain is required here.

Step 3: Construction of a damping factor that removes the singularities.

Since B is relatively closed in Z, for every $z \in Z \setminus B$ we have $\operatorname{dist}(z, B) > 0$; otherwise $z \in \overline{B}^Z = B$. Hence it holds that

$$d(z) > 0 \quad \text{for } z \in \Omega, \qquad d(z) = 0 \quad \text{for } z \in B.$$

Moreover, d is bounded, 1-Lipschitz, and permutation-invariant. Since d is 1-Lipschitz and strictly positive on Ω, it is locally bounded away from zero there. Hence d^{-1} is locally Lipschitz on Ω. Together with the local Lipschitzness of P, this implies that $H = (P - y)/d$ is locally Lipschitz. In addition, thanks to Lemma 2, H is permutation equivariant.

Apply Lemma 4 on Ω and permutation action group. There exists a permutation-invariant function

$$\beta : \Omega \to (0, 1)$$

such that

$$A := \beta H$$

is bounded and globally Lipschitz on Ω. Define

$$q(G, y) := \begin{cases} d(G, y)A(G, y), & (G, y) \in \Omega, \\ 0, & (G, y) \in B. \end{cases}$$

For $(G, y) \in \Omega$,

$$q(G, y) = \beta(G, y)\big(P(G, y) - y\big).$$

Finally, define

$$T_G(y) := y + q(G, y).$$

Equivalently,

$$T_G(y) = \begin{cases} (1 - \beta(G, y))y + \beta(G, y)P_G(y), & (G, y) \notin B, \\ y, & (G, y) \in B. \end{cases} \tag{11}$$

The freezing property on B is immediate from this definition.

Step 4: Joint global Lipschitz continuity of T.

Let

$$\|A\|_\infty \leq M_A, \qquad \operatorname{Lip}(A) \leq L_A.$$

For any $z, z' \in \Omega$,

$$\begin{aligned} \|q(z) - q(z')\| &= \|d(z)A(z) - d(z')A(z')\| \\ &\leq |d(z) - d(z')| \, \|A(z)\| + d(z')\|A(z) - A(z')\| \\ &\leq (M_A + L_A)\|z - z'\|, \end{aligned}$$

where we used $0 \le d \le 1$ (This follows from Definition 5) and the 1-Lipschitz property of d.

If $z \in \Omega$ and $z' \in B$, then

$$\|q(z) - q(z')\| = \|q(z)\| = d(z)\|A(z)\| \le M_A \operatorname{dist}(z, B) \le M_A \|z - z'\|.$$

If both points lie in B, the difference is zero. Thus q is globally Lipschitz on all of Z. Since the coordinate projection $(G, y) \mapsto y$ is 1-Lipschitz and

$$T_G(y) = y + q(G, y),$$

T is jointly globally Lipschitz. In particular, one may take

$$L_T \le 1 + M_A + L_A.$$

Step 5: Permutation equivariance.

The functions d and β are invariant, whereas H is equivariant. Therefore

$$A(\pi G, \pi y) = \pi A(G, y)$$

on Ω. Since B is invariant, it follows that

$$q(\pi G, \pi y) = \pi q(G, y)$$

on all of Z. Consequently,

$$T_{\pi G}(\pi y) = \pi y + q(\pi G, \pi y) = \pi y + \pi q(G, y) = \pi T_G(y).$$

Step 6: On a regular stable graph, taking the closure does not enlarge the switching section.

Fix $G \in \mathcal{G}_\star$. We claim that

$$\{y : (G, y) \in B\} = \Sigma^\circ_G. \tag{12}$$

The inclusion "$\supseteq$" follows immediately from the definition of the closure. For the reverse inclusion, it suffices to show that any point in the G-section of $\overline{\mathbf{\Sigma}^\circ}^{\,Z}$ belongs to Σ°_G. Suppose that

$$(G_k, y_k) \in \mathbf{\Sigma}^\circ, \qquad (G_k, y_k) \to (G, y).$$

For every k, choose two labels whose distinct branch values attain the common minimum distance from y_k. Because there are only finitely many pairs of labels, after passing to a subsequence we may assume that these labels are fixed, say $i \ne j$.

Since G is a stable-collapse point, if i and j belonged to the same stable cluster, then the two branches would agree identically throughout a neighborhood of G. For all sufficiently large k, this would contradict the fact that they represent distinct branch values at G_k. Therefore i and j belong to different stable clusters, and in particular

$$f_i(G) \ne f_j(G).$$

Because $G \notin D$, every branch is continuous at G. Passing to the limit in the equal-minimum relations gives

$$\|y - f_i(G)\| = \|y - f_j(G)\| = \min_\ell \|y - f_\ell(G)\|.$$

Thus $y \in \Sigma^\circ_G$, proving the reverse inclusion.

Step 7: Invariance of the strict Voronoi segment.

Fix

$$G \in \mathcal{G}_\star, \qquad y_0 \notin \Sigma^\circ_G,$$

and let

$$s = P_G(y_0).$$

For any other distinct solution $r \in S(G) \setminus \{s\}$,

$$\|y_0 - s\| < \|y_0 - r\|.$$

For $a \in [0,1]$, define

$$z_a = (1-a)y_0 + as.$$

Then

$$\|z_a - s\| = (1-a)\|y_0 - s\|.$$

On the other hand, the reverse triangle inequality yields

$$\begin{aligned}\|z_a - r\| &\geq \|y_0 - r\| - \|z_a - y_0\| \\ &= \|y_0 - r\| - a\|y_0 - s\| \\ &> (1-a)\|y_0 - s\| \\ &= \|z_a - s\|.\end{aligned}$$

Therefore

$$P_G(z_a) = s \qquad \text{and} \qquad z_a \notin \Sigma_G^\circ$$

for every $a \in [0,1]$. Hence

$$P_G(z) = s \qquad \text{for every } z \in [y_0, s],$$

and Step 6 implies

$$\{G\} \times [y_0, s] \subseteq \Omega.$$

Step 8: Convergence.

The set $\{G\} \times [y_0, s]$ is a compact subset of Ω. Since β is continuous and strictly positive on Ω,

$$c(G, y_0) := \min_{z \in [y_0, s]} \beta(G, z) > 0.$$

Furthermore, $0 < \beta < 1$. If $y_t \in [y_0, s]$, then Step 7 gives $P_G(y_t) = s$, and therefore

$$y_{t+1} = (1 - \beta(G, y_t))y_t + \beta(G, y_t)s \in [y_t, s] \subseteq [y_0, s].$$

By induction, every iterate remains on $[y_0, s]$. Moreover,

$$y_{t+1} - s = \big(1 - \beta(G, y_t)\big)(y_t - s). \tag{13}$$

It follows that

$$\|y_{t+1} - s\| \leq \big(1 - c(G, y_0)\big)\|y_t - s\|.$$

Iterating this estimate gives

$$\|y_t - s\| \leq \big(1 - c(G, y_0)\big)^t \|y_0 - s\|.$$

Hence $y_t \to s$.

Step 9: Almost-everywhere convergence and full support.

Fix $G \in \mathcal{G}_\star$. For any two distinct solutions $s, r \in S(G)$, the equidistance set

$$\{y : \|y - s\| = \|y - r\|\}$$

is an affine hyperplane, because the defining equality is equivalent to

$$2\langle y, r - s\rangle = \|r\|^2 - \|s\|^2.$$

Therefore

$$\Sigma_G^\circ \subseteq \bigcup_{\substack{s,r \in S(G) \\ s \neq r}} \{y : \|y - s\| = \|y - r\|\}$$

is contained in a finite union of Lebesgue-null affine hyperplanes. Thus

$$\mathcal{L}^m(\Sigma_G^\circ) = 0.$$

If $\mu \ll \mathcal{L}^m$, then

$$\mu(\Sigma_G^\circ) = 0.$$

Step 8 therefore shows that the iteration converges to a point of $S(G)$ for μ-almost every initialization.

Finally, if every strict Voronoi basin has positive μ-mass, then for every $s \in S(G)$,

$$\Pr(y_\infty = s \mid G) = \mu(V_s(G)) > 0.$$

Since $S(G)$ is finite and $y_\infty \in S(G)$ almost surely,

$$\operatorname{supp} \mathcal{L}(y_\infty \mid G) = S(G).$$

This proves all the conclusions. □

B FAMILY-WIDE ASYMPTOTIC REALIZATION BY A RECURRENT MP-GNN

We now realize the ideal dynamics of Appendix A using one message-passing GNN block that is reused at every recurrent step:

$$y_{t+1} = \mathrm{GNN}(G, y_t). \tag{14}$$

The random initialization supplies both the initial node addresses and the choice of solution basin. At later steps, the block receives only the graph and the current state. Throughout this appendix, write $d_y := q \geq 2$ and $m = nd_y$, using the graph and output spaces of Appendix A.

B.1 ARCHITECTURE AND MAIN RESULT

The recurrent block. For one call, write $y^+ = \mathrm{GNN}(G, y)$. Choose a finite number of hidden layers L, finite hidden widths, and a continuous encoder

$$\mathrm{enc} : \mathbb{R}^{d_v} \times \mathbb{R}^{d_y} \longrightarrow \mathbb{R}^{d_0}.$$

The block has the form

$$\begin{aligned}
h_i^{(0)} &= \mathrm{enc}(X_i, y_i), \\
m_i^{(\ell)} &= \sum_{j=1}^{n} M_\ell\Big(h_i^{(\ell)}, h_j^{(\ell)}, A_{ji}\Big), \\
h_i^{(\ell+1)} &= U_\ell\Big(h_i^{(\ell)}, m_i^{(\ell)}\Big), \qquad 0 \leq \ell < L, \\
g^{(L)} &= \sum_{j=1}^{n} R_L\Big(h_j^{(L)}\Big), \\
y_i^+ &= O\Big(h_i^{(L)}, g^{(L)}\Big).
\end{aligned} \tag{15}$$

Here the aggregation runs over all n senders, including $j = i$, since the edge tensor A assigns a feature vector to every ordered pair. The component maps, $\{M_\ell, U_\ell\}_{\ell=0}^{L-1}, R_L, O$, are arbitrary continuous maps between finite-dimensional Euclidean spaces and are shared across nodes. The only global operation is the final invariant sum: no global aggregate is formed or broadcast in the hidden layers. The entire L-layer block, including its encoder and readout, is denoted as $\mathrm{GNN}(G, \cdot)$ and is reused with the same parameters at every recurrent time. Hidden representations are recomputed within each call; only y_t is carried to the next call.

Remark (Sparse graphs). For an edge set $E(G) \subseteq [n] \times [n]$, encode $A_{ji} = (e_{ji}, a_{ji})$, where $e_{ji} = \mathbf{1}_{\{(j,i)\in E(G)\}}$ and $a_{ji} = 0$ whenever $e_{ji} = 0$. The class of such tensors is closed under relabeling, so Appendix A applies to any permutation-invariant domain of these graphs. The block constructed in Lemma 5 has $M_0 = c_i \otimes c_j \otimes A_{ji}$, which vanishes on every absent pair, even outside the compact realization family. Hence its all-pairs message sum agrees exactly with the neighbor sum over $(j, i) \in E(G)$. On the compact realization family, $\mathcal{R}_{\mathrm{edge}}$ records $(1, a_{ji})$ in present-edge slots and zero in absent-edge slots. Consequently, the block in Theorem 5 can use neighbor-only hidden message passing followed by the readout in (15).

It's straightforward that reindexing the message and readout sums gives the equivariance:

$$\mathrm{GNN}(\pi G, \pi y) = \pi\, \mathrm{GNN}(G, y). \tag{16}$$

Facts about the ideal update. Fix the operator T constructed in Theorem 4. Let Ω and P be as in Definitions 4 and 5. The proof in Appendix A gives

$$T_G(y) = (1 - \beta(G, y))y + \beta(G, y)P(G, y), \qquad 0 < \beta(G, y) < 1, \qquad (G, y) \in \Omega, \tag{17}$$

where β is continuous. The selector is continuous on Ω and is constant along the segment from y to $P(G, y)$. We will also use the bound

$$\|T_G(y) - y\| \leq \mathrm{dist}(y, S(G)) \qquad (G \in \mathcal{G}_\star,\ y \in \mathcal{Y}_n). \tag{18}$$

Indeed, off the switching set this follows from (17); on the switching set $T_G(y) = y$.

Theorem 5 (Family-wide asymptotic realization from random initialization)**.** *Assume $d_y \geq 2$. Let (G, Y_0) have a tight Borel probability law $\mathbb{P}$ on $\mathcal{G} \times \mathcal{Y}_n$, and let μ_G be a regular conditional distribution of Y_0 given G. Such a Borel kernel exists by disintegrating the joint law on the ambient Euclidean product and restricting the graph variable to $\mathcal{G}$. Assume:*

1. *$G \in \mathcal{G}_\star$ almost surely;*
2. *conditionally on G, the law μ_G of Y_0 satisfies $\mu_G \ll \mathcal{L}^m$ almost surely;*
3. *for almost every G and every $s \in S(G)$,*

$$\mu_G(V_s(G)) > 0.$$

All solution-dependent probability events are understood on the full-measure set $G \in \mathcal{G}_\star$. Fix a limit tolerance $\varepsilon > 0$, an operator tolerance $\rho > 0$, and a probability loss $\delta \in (0, 1)$. Then there exist a compact permutation-invariant event $E_{\varepsilon,\rho,\delta} \subseteq \mathcal{G} \times \mathcal{Y}_n$ with

$$\mathbb{P}(E_{\varepsilon,\rho,\delta}) \geq 1 - \delta \tag{19}$$

and one deterministic, autonomous, weight-tied MP-GNN block such that, for every $(G, y_0) \in E_{\varepsilon,\rho,\delta}$, the trajectory

$$\widehat{y}_0 = y_0, \qquad \widehat{y}_{t+1} = \mathrm{GNN}(G, \widehat{y}_t), \qquad t \geq 0,$$

converges to a limit $\widehat{y}_\infty$ satisfying

$$\begin{aligned} \operatorname{dist}(\widehat{y}_\infty, S(G)) &< \varepsilon, \\ \sup_{t \geq 0} \|\mathrm{GNN}(G, \widehat{y}_t) - T_G(\widehat{y}_t)\| &< \rho. \end{aligned} \tag{20}$$

The block receives only $(G, \widehat{y}_t)$ at every recurrent time.

The graph projection

$$\mathcal{K}_{\varepsilon,\rho,\delta} := \mathrm{proj}_{\mathcal{G}} E_{\varepsilon,\rho,\delta}$$

is compact, and the same fixed collection of continuous component maps works for every graph in this projection on its corresponding retained initialization fiber. The block is permutation equivariant as in (16). In particular,

$$\mathbb{P}(\widehat{y}_t \text{ converges and } \operatorname{dist}(\widehat{y}_\infty, S(G)) < \varepsilon) \geq 1 - \delta.$$

Moreover, the good event may be chosen so that every labeled branch has positive joint capture probability:

$$\mathbb{P}\big(Y_0 \in V_{f_a(G)}(G), \quad \widehat{y}_t \text{ converges to } \widehat{y}_\infty, \quad \|\widehat{y}_\infty - f_a(G)\| < \varepsilon\big) > 0, \qquad a = 1, \ldots, M. \tag{21}$$

Coincident labels refer to the same distinct target. The learned limit is required to be ε-close to the solution set; it need not belong exactly to $S(G)$.

The probability law need not be permutation invariant. Equivariance is a property of the block, and the retained event is made invariant by its construction. The branch-capture guarantee is joint over graphs and initializations; it does not assert positive retained mass in every basin of every individual graph.

B.2 Exact realization using the current node states

In this subsection, we establish conditions under which a continuous equivariant operator can be exactly realized by a message-passing GNN in our setting. We begin with the relevant symmetry-breaking condition. Without suitable symmetry-breaking features, standard message-passing GNNs may fail to distinguish graph inputs that the target operator distinguishes, limiting their ability to represent arbitrary continuous equivariant operators (see Xu et al. (2018)). We therefore introduce a condition under which the current node states provide distinct node addresses, and prove that these addresses enable exact realization on a compact family. This result provides the basis for constructing a message-passing GNN that follows the ideal dynamics introduced in Appendix A.

Definition 6 (Robust state-based node symmetry breaking)**.** *Let $K \subseteq \mathcal{G} \times \mathcal{Y}_n$ be compact and permutation invariant. We say that K admits* robust state-based node symmetry breaking *if there exist finitely many compact address regions $\mathcal{A}_1, \dots, \mathcal{A}_J \subseteq \mathbb{R}^{d_y}$ and $\eta > 0$ such that*

$$\operatorname{dist}(\mathcal{A}_j, \mathcal{A}_k) \geq \eta \qquad (j \neq k),$$

and, for every $(G, y) \in K$, the n node arguments $y_1, \dots, y_n$ belong to n different address regions.

Because there are finitely many compact address regions, pairwise disjointness already guarantees a positive separation margin; η simply denotes a common margin. The regions $\mathcal{A}_1, \dots, \mathcal{A}_J$ and their separation margin η are fixed for the whole family K, while the assignment of nodes to regions may vary between inputs. Since the node states $y_1, \dots, y_n$ belong to different address regions, each y_i serves as a unique node identifier within the current input (G, y).

For example, take $d_y = 2$ and a fixed two-node graph G_0 that is unchanged by swapping its nodes. Write $u_\theta = (\cos\theta, \sin\theta)$ and define two opposite arcs:

$$\mathcal{A}_1 = \{u_\theta : \theta \in [-\pi/4, \pi/4]\}, \qquad \mathcal{A}_2 = -\mathcal{A}_1.$$

These compact regions lie on the right and left sides of the unit circle, respectively, with $\operatorname{dist}(\mathcal{A}_1, \mathcal{A}_2) = \sqrt{2}$. The compact permutation-invariant family

$$K = \{(G_0, (u, -u)),\ (G_0, (-u, u)) : u \in \mathcal{A}_1\}$$

therefore satisfies Definition 6 (i.e., admits robust state-based node symmetry breaking) with $J = 2$ and $\eta = \sqrt{2}$: one node lies in each region, although their assignments can be swapped.

In contrast, the following full-circle family does not satisfy Definition 6

$$K_{\text{circle}} = \left\{ \big(G_0, (u_\theta, -u_\theta)\big) : \theta \in [0, 2\pi] \right\}.$$

The two node states are still always distance 2 apart, but each now ranges over the entire circle. There is no finite collection of compact regions that simultaneously (i) are positively separated, (ii) cover the entire unit circle, and (iii) place u_θ and $-u_\theta$ in different regions for every θ. Indeed, because the circle is connected, conditions (i) and (ii) force it to lie entirely in one region, contradicting (iii).

In the remainder of this subsection, we show how continuous equivariant operators can be exactly realized by message-passing GNNs under robust state-based node symmetry breaking. Subsections B.3 and B.4 then establish this condition for the compact families used in our construction.

Lemma 5 (Exact MP-GNN realization on a compact family)**.** *Let $K \subseteq \mathcal{G} \times \mathcal{Y}_n$ be compact, permutation invariant, and admit robust state-based node symmetry breaking. Fix integers $u, v \geq 0$ with $u + v \geq 1$ and, when $v > 0$, positive integers $p_1, \dots, p_v$. For $\ell = 1, \dots, u$, let $\varphi_\ell : K \to \mathbb{R}$ be continuous and permutation invariant. For $b = 1, \dots, v$, let $F_b : K \to \mathbb{R}^{n \times p_b}$ be continuous and permutation equivariant. Then there exists a MP-GNN, with output width $u + \sum_{b=1}^{v} p_b$ and the structural form (15) realizes all these maps simultaneously and exactly on K: its output row at node i is*

$$\tau(G, y, i) := \Big(\varphi_1(G, y), \dots, \varphi_u(G, y), [F_1(G, y)]_i, \dots, [F_v(G, y)]_i\Big).$$

Empty output blocks are omitted. The component maps and all widths are chosen once for the whole compact family.

Proof. We construct one message-passing layer that records the graph and state in node-code order, and then define a continuous readout on that record.

Step 1: Construct the message-passing representation. Fix one collection of compact address regions $\mathcal{A}_1, \dots, \mathcal{A}_J$ satisfying Definition 6 simultaneously for all $(G, y) \in K$. These regions remain fixed throughout the construction; only the region containing each node state may vary with the input. The map that equals the jth standard basis vector $e_j \in \mathbb{R}^J$ on $\mathcal{A}_j$ is continuous on their closed union. The Tietze extension theorem (Munkres, 2014, Theorem 35.1(b)), applied coordinatewise, gives a continuous map

$$c : \mathbb{R}^{d_y} \longrightarrow \mathbb{R}^J, \qquad c(w) = e_j \quad (w \in \mathcal{A}_j).$$

For $(G, y) \in K$, the codes $c_i := c(y_i)$ are distinct standard basis vectors. Set $d_0 = d_v + d_y + J$ and define

$$\operatorname{enc}(X_i, y_i) := (X_i, y_i, c(y_i)).$$

For the single local layer, take

$$M_0\Big(h_i^{(0)}, h_j^{(0)}, A_{ji}\Big) := c_i \otimes c_j \otimes A_{ji},$$
$$U_0\Big(h_i^{(0)}, m_i^{(0)}\Big) := \big(c_i,\ c_i \otimes (1, X_i, y_i),\ m_i^{(0)}\big).$$

Thus $h_i^{(1)} = (c_i, N_i, E_i)$, where

$$N_i := c_i \otimes (1, X_i, y_i), \qquad E_i := \sum_{j=1}^{n} c_i \otimes c_j \otimes A_{ji}.$$

Set $R_1(c_i, N_i, E_i) := (N_i, E_i)$. The final global sum is

$$g^{(1)} = (\mathcal{R}_{\text{node}}, \mathcal{R}_{\text{edge}}), \qquad \mathcal{R}_{\text{node}} := \sum_i c_i \otimes (1, X_i, y_i), \qquad \mathcal{R}_{\text{edge}} := \sum_{i,j} c_i \otimes c_j \otimes A_{ji}.$$

All tensors are vectorized before concatenation. The displayed component maps are continuous and shared across nodes.

Step 2: Recover the graph up to relabeling. For a standard basis vector e_k, the tensor $e_k \otimes z$ places z in its kth block and zero in the remaining blocks. Hence $\mathcal{R}_{\text{node}}$ records $(1, X_i, y_i)$ in the slot indexed by c_i, and $\mathcal{R}_{\text{edge}}$ records A_{ji} in the slot indexed by the ordered pair (c_i, c_j).

Note that, the roles of the constant-one coordinates are therefore as follows: The 1 in $(1, X_i, y_i)$ is a node-occupancy indicator. Without it, a node satisfying $X_i = 0$ and $y_i = 0$ would produce a zero block, indistinguishable from an unused address slot. This ensures that the global tensors reconstruct nodes and edges even when their features are zero; they do not carry solution information or participate in symmetry breaking.

Suppose (G, y) and (G', y') produce the same two global tensors. Both inputs have the same number n of nodes, each occupying a distinct address slot marked by the constant-one coordinate. Matching these occupied slots therefore defines a unique permutation $\pi \in \mathfrak{S}_n$ such that

$$c'_{\pi(i)} = c_i, \qquad i \in [n].$$

Equality of the corresponding blocks then implies

$$X'_{\pi(i)} = X_i, \qquad y'_{\pi(i)} = y_i, \qquad A'_{\pi(j),\pi(i)} = A_{ji}.$$

Thus $(G', y') = (\pi G, \pi y)$. If the receiver codes also agree, $c'_{i'} = c_i$, their distinctness gives $i' = \pi(i)$. Therefore

$$\Gamma(G, y, i) := (c_i, \mathcal{R}_{\text{node}}, \mathcal{R}_{\text{edge}})$$

determines the input and its receiving node up to simultaneous relabeling.

Step 3: The target is determined by this representation. If $\Gamma(G, y, i) = \Gamma(G', y', i')$, Step 2 supplies a permutation matching both inputs and receivers. Invariance of each φ_ℓ and equivariance of each F_b imply

$$\tau(G, y, i) = \tau(G', y', i').$$

Consequently, the prescription

$$D_0\big(\Gamma(G, y, i)\big) := \tau(G, y, i)$$

defines a single-valued map on the reconstruction image $\mathcal{I} := \Gamma(K \times [n])$.

Step 4: Construct a continuous output map. Both Γ and τ are continuous on the compact space $K \times [n]$, where $[n]$ has the discrete topology. The induced map D_0 is continuous. For completeness, if $w_k \to w$ in $\mathcal{I}$, choose preimages $z_k \in K \times [n]$. Every subsequence of these preimages has a further convergent subsequence, say $z_{k_j} \to z$. Continuity gives $\Gamma(z) = w$ and

$$D_0(w_{k_j}) = \tau(z_{k_j}) \longrightarrow \tau(z) = D_0(w).$$

If $D_0(w_k)$ did not converge to $D_0(w)$, a subsequence staying a positive distance away would contradict this argument. Thus D_0 is continuous. This is the compact-domain quotient argument in explicit form.

The image $\mathcal{I}$ is compact and hence closed in its Euclidean ambient space. Coordinatewise Tietze extension gives a continuous ambient map $\widetilde{D}$ agreeing with D_0 on $\mathcal{I}$. Define

$$O\big((c_i, N_i, E_i), g^{(1)}\big) := \widetilde{D}(c_i, \mathcal{R}_{\mathrm{node}}, \mathcal{R}_{\mathrm{edge}}).$$

On K, this output equals $\tau(G, y, i)$, as required. □

Lemma 5 shows that, once robust node-state symmetry breaking is available, any prescribed finite collection of continuous permutation-invariant scalar maps and continuous permutation-equivariant node maps can be realized simultaneously and exactly on K. This proof differs from the Stone–Weierstrass density arguments commonly used in GNN universality theory, e.g., Keriven & Peyré (2019); Azizian & marc lelarge (2021); Geerts & Reutter (2022). It constructs a finite-dimensional representation $(c_i, \mathcal{R}_{\mathrm{node}}, \mathcal{R}_{\mathrm{edge}})$ that reconstructs (G, y, i) up to simultaneous node relabeling. The target therefore factors continuously through this representation, and the Tietze extension theorem provides an ambient continuous readout. Because the present MP-GNN class allows arbitrary continuous component maps, this construction yields exact realization. By contrast, Stone–Weierstrass uses a function algebra that contains the constants and separates the relevant points or equivalence classes to establish density in in the relevant continuous-function space, yielding uniform approximation rather than exact equality.

B.3 Node separation and a common collection of boxes

To apply Lemma 5, we need the node states to provide robust addresses (symmetry breaking). The next two lemmas establish this condition: the first derives node separation from random initialization, and the second places the separated states in common address boxes, allowing arbitrarily small probability losses.

Lemma 6 (Finite-time node separation for the ideal dynamics)**.** *Assume $d_y \geq 2$, $G \in \mathcal{G}_\star$ almost surely, and $\mu_G \ll \mathcal{L}^m$ almost surely. Put $Y_t := T_G^t(Y_0)$. Then*

$$\mathbb{P}((Y_t)_i \neq (Y_t)_j \text{ for all } t \in \mathbb{N}_0 \text{ and } i \neq j \,|\, G) = 1 \quad \text{for almost every } G. \tag{22}$$

In particular, this event has joint probability one.

Let $E \subseteq \mathcal{G}_\star \times \mathcal{Y}_n$ be compact and permutation invariant, let $H \in \mathbb{N}_0$, and let $\gamma > 0$. There exist $\eta > 0$ and a compact invariant $E' \subseteq E$ such that

$$\mathbb{P}(E \setminus E') < \gamma,$$
$$\|[T_G^t(y_0)]_i - [T_G^t(y_0)]_j\| \geq \eta \quad ((G, y_0) \in E',\ 0 \leq t \leq H,\ i \neq j).$$

If finitely many measurable subsets $C_a \subseteq E$ have positive probability, E' may also be chosen with $\mathbb{P}(C_a \cap E') > 0$ for every a.

Proof. For $n = 1$ there are no node pairs; take $E' = E$ and any $\eta > 0$. Assume $n \geq 2$, and fix a graph $G \in \mathcal{G}_\star$ for which the conditional initialization law is absolutely continuous. The key observation is that a collision forces an initial node state difference onto a fixed line determined by a solution. In two or more dimensions, such an event has zero measure.

For $i < j$, the node-state-difference map

$$D_{ij} : \mathcal{Y}_n \to \mathbb{R}^{d_y}, \qquad D_{ij}(y) = y_i - y_j,$$

is linear and surjective. For any $s \in S(G)$, define

$$N_{s,ij} := D_{ij}^{-1}\big(\mathrm{span}\{s_i - s_j\}\big),$$

which consists of states whose node difference $y_i - y_j$ is a scalar multiple of the target difference $s_i - s_j$. The span on the right has dimension at most one.

Write $W = \mathrm{span}\{s_i - s_j\}$. Since D_{ij} is surjective, the rank–nullity theorem gives $\dim\ker D_{ij} = m - d_y$. Its restriction to $N_{s,ij} = D_{ij}^{-1}(W)$ is a surjective linear map onto W with the same kernel. Applying rank–nullity to this restriction yields

$$\dim N_{s,ij} = \dim\ker D_{ij} + \dim W = m - d_y + \dim\mathrm{span}\{s_i - s_j\}.$$

The span has dimension zero if $s_i = s_j$ and one otherwise. Thus, since $d_y \geq 2$,

$$\dim N_{s,ij} \leq m - d_y + 1 \leq m - 1 < m.$$

Thus $N_{s,ij}$ is a proper linear subspace and has zero Lebesgue measure. Define set

$$N_G := \Sigma_G^\circ \bigcup \left(\bigcup_{s \in S(G)} \bigcup_{i<j} N_{s,ij} \right)$$

where $\Sigma_G^\circ \subseteq \mathcal{Y}_n$ is the switching set for the fixed graph G, as defined in Definition 3. The set N_G collects exceptional initializations excluded to ensure a unique selected branch from $\{f_1(G), \cdots, f_M(G)\}$ and rule out node-state collisions at every finite time. The set N_G is Lebesgue null: the switching set is Lebesgue null by Theorem 4, and the remaining union is finite. Absolute continuity gives $\mu_G(N_G) = 0$.

For $y_0 \notin N_G$, let $s = P(G, y_0)$. Basin preservation and (17) give

$$y_t = s + a_t(y_0 - s), \qquad a_t := \prod_{k=0}^{t-1} (1 - \beta(G, y_k)) > 0, \qquad a_0 := 1. \tag{23}$$

If $(y_t)_i = (y_t)_j$ at a finite time, then

$$0 = a_t\big((y_0)_i - (y_0)_j\big) + (1 - a_t)(s_i - s_j).$$

Since $a_t > 0$, the initial row difference belongs to $\operatorname{span}\{s_i - s_j\}$, contradicting $y_0 \notin N_G$. The same null set works for all finite times. This proves the conditional claim (22), and integration over G proves the joint claim.

For the compact restriction, define

$$\Delta_H(G, y_0) := \min_{\substack{0 \leq t \leq H \\ i<j}} \|[T_G^t(y_0)]_i - [T_G^t(y_0)]_j\|.$$

Finite iterates of T are jointly continuous and equivariant, so Δ_H is continuous and invariant. The probability-one statement implies $\mathbb{P}(E \cap \{\Delta_H = 0\}) = 0$, and hence

$$\mathbb{P}(E \cap \{\Delta_H < \eta\}) \longrightarrow 0 \qquad (\eta \downarrow 0).$$

Choose $\eta > 0$ so that this loss is below γ and, when subsets C_a are prescribed, below $\frac{1}{2} \min_a \mathbb{P}(C_a)$. Then $E' := E \cap \{\Delta_H \geq \eta\}$ is compact, invariant, and has all the required properties. □

Lemma 7 (Separated boxes for finitely many state maps)**.** *Let E be a compact metric space with a finite Borel measure μ. Let $\mathcal{T}$ be a nonempty finite index set, and let*

$$V_t = (v_{t,1}, \ldots, v_{t,n}) : E \longrightarrow \mathbb{R}^{n \times d_y}, \qquad t \in \mathcal{T},$$

be continuous. (Here i denotes the node index.) Suppose, for some $\eta > 0$,

$$\|v_{t,i}(x) - v_{t,j}(x)\| \geq \eta \qquad (x \in E,\ t \in \mathcal{T},\ i \neq j).$$

For any $\omega, \gamma > 0$, there exist a compact $E' \subseteq E$ and finitely many pairwise positively separated compact axis-aligned boxes $Q_1, \ldots, Q_J \subset \mathbb{R}^{d_y}$, with centers $b_1, \ldots, b_J$, such that:

1. *$\mu(E \setminus E') < \gamma$;*
2. *every $v_{t,i}(x)$ lies in exactly one box, and the n items of each $V_t(x)$ lie in different boxes;*
3. *replacing every item in $V_t(x)$ by its box center changes the full state by less than ω:*

$$\|V_t(x) - b(V_t(x))\| < \omega \qquad (x \in E',\ t \in \mathcal{T}).$$

If finitely many measurable subsets $C_a \subseteq E$ have positive measure, E' can preserve positive measure in every C_a. If E carries a continuous node-permutation action and $V_t(\pi x) = \pi V_t(x)$, then E' can also be chosen permutation invariant.

Proof. We construct one common grid in the node-state space $\mathbb{R}^{d_y}$. First, we choose its mesh small enough to separate different nodes and control the error from replacing states by cell centers. Second, we shift the grid so that the set of inputs placing any node state on a grid boundary has measure zero. Third, we remove inputs whose node states lie in a sufficiently thin neighborhood of these boundaries, losing arbitrarily little measure. The remaining node states then lie in finitely many separated compact boxes.

Step 1: Choose a sufficiently small mesh. Since E is compact and there are only finitely many continuous maps $v_{t,i}$, the union of all their ranges is compact and hence bounded. Choose a grid mesh ℓ satisfying

$$0 < \ell < \min\left\{\frac{2\omega}{\sqrt{nd_y}}, \frac{\eta}{\sqrt{d_y}}\right\}.$$

For a shift vector $a \in [0,\ell)^{d_y}$, the grid cells are

$$a + \ell z + [0,\ell]^{d_y}, \qquad z \in \mathbb{Z}^{d_y}.$$

Each cell has diameter $\sqrt{d_y}\ell < \eta$, so it cannot contain two node states from the same $V_t(x)$, whose distance is at least η. The other bound on ℓ will control the center-replacement error in Step 4.

Step 2: Shift the grid to give its boundaries zero measure. Fix a coordinate $s \in [d_y]$. We first identify the coordinate values that occur on a set of inputs of positive measure:

$$B_s := \{b \in \mathbb{R} : \mu\{x \in E : (v_{t,i}(x))_s = b\} > 0 \text{ for some } t \in \mathcal{T},\ i \in [n]\}\,.$$

This set is at most countable. Indeed, for any fixed (t,i) and positive integer p, only finitely many values b can satisfy

$$\mu\{x \in E : (v_{t,i}(x))_s = b\} \geq \frac{1}{p}.$$

The corresponding level sets are disjoint, and infinitely many such sets would contradict $\mu(E) < \infty$. Every positive-measure level has measure at least $1/p$ for some p, so there are at most countably many such levels. Taking the finite union over (t,i) shows that B_s is countable.

The grid boundaries in coordinate s occur at the values $a_s + k\ell$, $k \in \mathbb{Z}$. We want none of these values to belong to B_s. A shift that places some $b \in B_s$ on a boundary must satisfy $a_s = b - k\ell$ for some integer k. Thus all such shifts belong to the countable set

$$\{b - k\ell : b \in B_s,\ k \in \mathbb{Z}\}.$$

Since the interval $[0,\ell)$ is uncountable, it contains a shift a_s outside this set. For this choice, every boundary value $a_s + k\ell$ lies outside B_s. By the definition of B_s,

$$\mu\{x \in E : (v_{t,i}(x))_s = a_s + k\ell\} = 0 \qquad (t \in \mathcal{T},\ i \in [n],\ k \in \mathbb{Z}).$$

Choose each coordinate shift in this way and fix $a = (a_1, \ldots, a_{d_y})$ for the entire family. The union of all grid-boundary hyperplanes is

$$\mathcal{H} := \bigcup_{s=1}^{d_y} \bigcup_{k\in\mathbb{Z}} \{v \in \mathbb{R}^{d_y} : v_s = a_s + k\ell\}.$$

By the countable union bound, the set of inputs for which any $v_{t,i}(x)$ lies on $\mathcal{H}$ has measure zero.

Step 3: Remove a thin neighborhood of the boundaries. For $0 < \tau < \ell/2$, remove the inputs

$$N_\tau := \left\{x \in E : \min_{t\in\mathcal{T},\, i\in[n]} \operatorname{dist}(v_{t,i}(x), \mathcal{H}) < \tau\right\}.$$

The set $\mathcal{H}$ is closed: for each coordinate s, the lattice $a_s + \ell\mathbb{Z}$ is closed, and there are only finitely many coordinates. The displayed minimum is continuous, so N_τ is relatively open in E. As $\tau \downarrow 0$, these sets decrease to the zero-measure event that some node state lies on $\mathcal{H}$. Since μ is finite, continuity of measure from above gives $\mu(N_\tau) \to 0$.

Choose τ sufficiently small that

$$\mu(N_\tau) < \gamma$$

and, if the subsets C_a are prescribed, also

$$\mu(N_\tau) < \tfrac{1}{2} \min_a \mu(C_a).$$

Then $E' := E \setminus N_\tau$ is compact, loses less than γ measure, and retains positive measure in every prescribed C_a. Node permutations reorder the node states without changing their internal coordinates. Since every node is tested against the same fixed grid, equivariance gives

$$\min_{t,i} \operatorname{dist}(v_{t,i}(\pi x), \mathcal{H}) = \min_{t,i} \operatorname{dist}(v_{t,\pi^{-1}(i)}(x), \mathcal{H}) = \min_{t,j} \operatorname{dist}(v_{t,j}(x), \mathcal{H}).$$

Thus N_τ and $E' = E \setminus N_\tau$ are permutation invariant, regardless of whether μ is invariant.

Step 4: Obtain separated boxes and bound the center error. Every retained node state lies at distance at least τ from every grid boundary, and therefore belongs to one of the shrunken boxes

$$Q_z := a + \ell z + [\tau, \ell - \tau]^{d_y}.$$

Keep only boxes containing a node state $v_{t,i}(x)$ for some $x \in E'$, and enumerate them as $Q_1, \ldots, Q_J$. Only finitely many are needed because all node-state ranges are bounded.

These boxes are compact, and distinct boxes have distance at least $2\tau > 0$. Their diameters satisfy

$$\operatorname{diam}(Q_j) = \sqrt{d_y}(\ell - 2\tau) < \eta.$$

Consequently, the n node states of each retained $V_t(x)$ belong to different boxes.

Finally, each node state is within $\sqrt{d_y}(\ell - 2\tau)/2$ of its box center. Summing the squared errors over the n nodes gives

$$\|V_t(x) - b(V_t(x))\| \le \frac{\sqrt{n d_y}}{2}(\ell - 2\tau) < \omega.$$

This proves all the claims. □

In the application below, E' consists of retained graph–initialization pairs, and $V_t(G, y_0) = T_G^t(y_0)$ for finitely many times. The boxes therefore provide one fixed collection of address regions for all retained graphs, initializations, and these times. Subsection B.4 then constructs terminal dynamics that keep each node in its box and contract toward its center, preserving the addressing condition while ensuring convergence near the selected solution.

B.4 Proof of the main result

Given the preparations, we are now ready to prove Theorem 5.

Proof of Theorem 5. Choose positive probability budgets satisfying

$$\delta_{\mathrm{loc}} + \delta_{\mathrm{stop}} + \delta_{\mathrm{sep}} + \delta_{\mathrm{grid}} < \delta.$$

These budgets allow us to exclude graph–initialization pairs with total probability less than δ, while preserving positive probability for each labeled solution branch. The remaining pairs form a compact family whose ideal trajectories all approach their selected solutions within a common finite number of steps. During these steps, different nodes have uniformly separated states, which can be placed in small, separated boxes from one fixed collection shared across all retained inputs.

We then construct an update rule that follows T_G exactly until the state is sufficiently close to its selected solution. From that point onward, each node moves halfway toward the center of its current box at every step. This ensures convergence while keeping different nodes distinguishable. Switching close to the solution and using small boxes controls both the final solution error and the deviation from the ideal update. Lemma 5 then realizes this rule with a single fixed GNN block, whose repeated application generates all retained trajectories.

Step 1: Localize the initializations and the ideal trajectories. For $G \in \mathcal{G}^\star$, the proof of Theorem 4 identifies the G-section of the joint bad set with Σ_G°, even after taking the joint closure. Its switching-set nullity and the first two probability assumptions therefore give $\mathbb{P}(\Omega) = 1$. For each label a, define

$$B_a := \{(G, y_0) \in \Omega : P(G, y_0) = f_a(G)\}.$$

These sets are Borel, since P and the regular branches are continuous on their domains. For $G \in \mathcal{G}_\star$, the G-section of B_a is $V_{f_a(G)}(G)$, and the section is empty otherwise. Integration of the indicator of B_a against the conditional kernel is measurable in G, so disintegration and the positive-basin assumption give, with $\mathbb{P}_G$ denoting the graph marginal,

$$\mathbb{P}(B_a) = \int_{\mathcal{G}_\star} \mu_G(V_{f_a(G)}(G))\, d\mathbb{P}_G(G) > 0.$$

A tight finite Borel measure on a metric space is inner regular by compact sets: restrict first to a compact set of arbitrarily high probability and use Borel regularity there. Hence choose compact sets $C_0 \subseteq \Omega$ and $C_a \subseteq B_a$ such that

$$\mathbb{P}(C_0) > 1 - \delta_{\mathrm{loc}}, \qquad \mathbb{P}(C_a) > 0 \quad (a = 1, \dots, M).$$

Finite symmetrization gives the compact invariant event

$$E^0 := \bigcup_{\pi \in \mathfrak{S}_n} \pi \left(C_0 \cup \bigcup_{a=1}^{M} C_a \right) \subseteq \Omega, \qquad \mathbb{P}(E^0) > 1 - \delta_{\mathrm{loc}}.$$

The compact sets C_a are the branch subsets whose mass we preserve.

Write $s(G, y_0) := P(G, y_0)$. The union of the full ideal segments,

$$\mathcal{C} := \left\{ \big(G, (1-a)y_0 + a\, s(G, y_0)\big) : (G, y_0) \in E^0,\ a \in [0,1] \right\},$$

is a compact subset of Ω: it is the continuous image of $E^0 \times [0,1]$, and Appendix A shows that each segment stays in its initial strict basin. Set

$$b_* := \min_{(G,y) \in \mathcal{C}} \beta(G, y) \in (0,1), \qquad R_0 := \max_{(G,y_0) \in E^0} \|y_0 - s(G, y_0)\| < \infty.$$

Equation (17) then gives the uniform bound

$$\|T_G^t(y_0) - s(G, y_0)\| \le (1 - b_*)^t R_0 \qquad ((G, y_0) \in E^0,\ t \ge 0). \tag{24}$$

Step 2: Choose an entrance threshold with a positive margin. We use distance to the solution set to decide when to switch from the ideal update $T_G(\cdot)$ to the final convergent update. By excluding a small-probability set of graph–initialization pairs, we ensure that this distance stays uniformly away from the chosen threshold at every time up to a common finite horizon. The resulting gap separates the states where the two updates are used, allowing us to combine them into a continuous rule.

For $x = (G, y_0) \in E^0$, define

$$y_t(x) := T_G^t(y_0), \qquad d_t(x) := \mathrm{dist}(y_t(x), S(G)).$$

Recall that E^0 is a compact subset of Ω, so its graph projection is a compact subset of $\mathcal{G}_\star$. Each branch f_a is locally Lipschitz on $\mathcal{G}_\star$ and hence continuous on this projection. Moreover, $y_0(x)$ is a coordinate projection, and the recurrence $y_{t+1}(x) = T_G(y_t(x))$ uses the jointly continuous map $(G, y) \mapsto T_G(y)$. Induction therefore shows that $y_t(x)$ is continuous in $x = (G, y_0)$ for every finite t. Consequently,

$$d_t(x) = \min_{a=1,\dots,M} \|y_t(x) - f_a(G)\|$$

is continuous as the minimum of finitely many continuous functions.

The distance d_t is also permutation invariant. Indeed, equivariance of T gives $y_t(\pi x) = \pi y_t(x)$, while $S(\pi G) = \pi S(G)$. Since node permutations preserve Euclidean distances,

$$d_t(\pi x) = \mathrm{dist}(\pi y_t(x), \pi S(G)) = \mathrm{dist}(y_t(x), S(G)) = d_t(x).$$

Finally, Appendix A shows that the ideal trajectory remains in the strict basin of its selected solution $s(x)$. Thus $s(x)$ remains the unique nearest solution, and

$$d_t(x) = \|y_t(x) - s(x)\|.$$

Choose the radius before choosing a time horizon. For each fixed $t \in \mathbb{N}_0$, there are at most countably many distance values $b \ge 0$ for which

$$\mathbb{P}\{x \in E^0 : d_t(x) = b\} > 0,$$

by the same countability argument used in Step 2 of the proof of Lemma 7. Since there are only countably many times t, the collection of all such values across all finite times is still countable. The interval $(0, \min\{\varepsilon, \rho/2\})$ is uncountable, so we can choose a radius r in this interval outside that collection. This ensures

$$\mathbb{P}\{x \in E^0 : d_t(x) = r\} = 0 \qquad \text{for every } t \in \mathbb{N}_0.$$

Now choose $H \in \mathbb{N}_0$ with

$$(1 - b_*)^H R_0 < r/4.$$

In particular, every trajectory from E^0 is within $r/4$ of its selected solution by time H.

For $\zeta > 0$, the events

$$N_{\text{stop}}(\zeta) := \{x \in E^0 : |d_t(x) - r| < \zeta \text{ for some } 0 \le t \le H\}$$

decrease to a null event as $\zeta \downarrow 0$. Choose $0 < \zeta < r/2$ such that

$$\mathbb{P}(N_{\text{stop}}(\zeta)) < \min\left\{\delta_{\text{stop}}, \tfrac{1}{2} \min_a \mathbb{P}(C_a)\right\}.$$

The event

$$E_{\text{stop}} := E^0 \setminus N_{\text{stop}}(\zeta)$$

is compact and permutation invariant, with

$$\mathbb{P}(E_{\text{stop}}) > 1 - \delta_{\text{loc}} - \delta_{\text{stop}}, \qquad \mathbb{P}(C_a \cap E_{\text{stop}}) > \tfrac{1}{2}\mathbb{P}(C_a) > 0 \quad (a = 1, \dots, M).$$

On E_{stop}, every ideal trajectory stays at least ζ away from the distance threshold r through time H:

$$|d_t(x) - r| \ge \zeta \qquad (x \in E_{\text{stop}},\ 0 \le t \le H).$$

Thus, at each of these times, the distance to the solution set is either at most $r - \zeta$ or at least $r + \zeta$.

On E_{stop}, define the entrance time

$$h(x) := \min\{0 \le t \le H : d_t(x) \le r - \zeta\}.$$

It exists because $d_H < r/4 < r - \zeta$. Every retained trajectory satisfies

$$d_t(x) \ge r + \zeta \quad (0 \le t < h(x)), \qquad d_{h(x)}(x) \le r - \zeta. \tag{25}$$

Thus it first visits the inner neighborhood of radius $r - \zeta$ after remaining outside the larger neighborhood of radius $r + \zeta$. For each $k \in \{0, \dots, H\}$, its entrance-time level set is

$$L_k := E_{\text{stop}} \cap \{d_k \le r - \zeta\} \cap \bigcap_{t<k} \{d_t \ge r + \zeta\}.$$

Thus L_k consists of graph–initialization pairs, (G, y_0), whose ideal trajectories first enter the $(r-\zeta)$-neighborhood of the solution set at time k. This is compact and invariant. When $k = 0$, there are no pre-entrance inequalities. The sets L_k form a finite partition of E_{stop}. The horizon H is a uniform upper bound on this retained family, whereas $h(x)$ may be smaller and depends on the trajectory.

Step 3: Obtain one collection of distinct node-address boxes. Apply Lemma 6 to E_{stop}, horizon H, loss δ_{sep}, and the subsets $C_a \cap E_{\text{stop}}$. It gives a compact invariant $E_{\text{sep}} \subseteq E_{\text{stop}}$ and $\eta > 0$ such that

$$\mathbb{P}(E_{\text{stop}} \setminus E_{\text{sep}}) < \delta_{\text{sep}}, \qquad \mathbb{P}(C_a \cap E_{\text{sep}}) > 0,$$

and, uniformly over all retained inputs and times up to H, distinct nodes have states at least η apart:

$$\|[T_G^t(y_0)]_i - [T_G^t(y_0)]_j\| \ge \eta \qquad \big((G, y_0) \in E_{\text{sep}},\ 0 \le t \le H,\ i \ne j\big).$$

Then choose any $0 < \omega < \zeta$, apply Lemma 7 on E_{sep} with the restricted measure $\mathbb{P}$, tolerance ω, loss δ_{grid}, subsets $C_a \cap E_{\text{sep}}$, and state maps

$$V_t(G, y_0) := T_G^t(y_0), \qquad t = 0, \dots, H.$$

The maps are continuous and equivariant, and the row-separation hypothesis was just established. The lemma gives boxes $Q_1, \dots, Q_J$ and a compact invariant final event

$$E := E_{\varepsilon,\rho,\delta} \subseteq E_{\text{sep}}, \qquad \mathbb{P}(E) > 1 - \delta, \qquad \mathbb{P}(C_a \cap E) > 0 \quad (a = 1, \dots, M). \tag{26}$$

The probability bound follows by adding the four losses. For every $x \in E$ and $t \in \{0, \dots, H\}$, there exist pairwise distinct indices $j_1, \dots, j_n \in [J]$ such that

$$[y_t(x)]_i \in Q_{j_i} \qquad (i = 1, \dots, n). \tag{27}$$

Replacing each node state by the center of its box changes the full state by less than ω:

$$\|y_t(x) - (b_{j_1}, \dots, b_{j_n})\| < \omega,$$

where b_j is the center of Q_j. The indices j_i may depend on (x, t), while the boxes and their centers are fixed for the whole family.

Using the final retained event E, define two families of graph–state pairs:

$$K_{\mathrm{tr}} := \bigcup_{k=0}^{H} \bigcup_{t=0}^{k-1} \{(G, y_t(x)) : x = (G, y_0) \in E \cap L_k\},$$
$$K_{\mathrm{ent}} := \bigcup_{k=0}^{H} \{(G, y_k(x)) : x = (G, y_0) \in E \cap L_k\}.$$

Here K_{tr} collects the states of each retained trajectory *before* its first visit to the $(r-\zeta)$-neighborhood of $S(G)$; these are the transient states. The family K_{ent} collects the state *at* that first visit for each trajectory; these are the entrance states. By the definition of L_k,

$$\mathrm{dist}(y, S(G)) \geq r + \zeta \quad \text{on } K_{\mathrm{tr}}, \qquad \mathrm{dist}(y, S(G)) \leq r - \zeta \quad \text{on } K_{\mathrm{ent}}.$$

Both families are compact, being finite unions of continuous images of the compact sets $E \cap L_k$. They are permutation invariant because these sets are invariant and the maps y_t are equivariant. In every state from either family, different nodes belong to different boxes from the same collection $Q_1, \dots, Q_J$—i.e., see (27). Hence, K_{tr} and K_{ent} admit robust node-state symmetry breaking (Definition 6).

Step 4: Construct the terminal dynamics and realize both regimes. Let $Q := \bigcup_{j=1}^{J} Q_j$, and define $b(v) := b_j$ $(v \in Q_j)$, where b_j is the center of Q_j. This map is continuous on Q because it is constant on each box and the boxes are positively separated. For a state y with every node state $y_i \in Q$, define

$$b(y) := (b(y_1), \dots, b(y_n)), \qquad \Psi(y) := b(y) + \tfrac{1}{2}(y - b(y)).$$

The update Ψ moves each node state halfway toward the center of its current box. Since the boxes are convex, each node remains in that same box, so

$$b(\Psi(y)) = b(y).$$

Repeated updates therefore converge to the fixed center pattern $b(y)$, halving each node's distance to its box center at every step.

For each entrance state, include the entire line segment joining it to its box-center pattern. These segments form the terminal family:

$$K_{\mathrm{term}} := \left\{ \big(G, b(y) + a(y - b(y))\big) : (G, y) \in K_{\mathrm{ent}},\ a \in [0, 1] \right\}. \tag{28}$$

Here $a = 1$ gives the entrance state y, while $a = 0$ gives the center pattern $b(y)$ toward which the terminal updates converge. Fix an entrance state y. Its node states belong to distinct boxes: there are pairwise distinct indices $j_1, \dots, j_n$ such that $y_i \in Q_{j_i}$. For any state $z = b(y) + a(y - b(y))$ on the segment,

$$z_i = b_{j_i} + a(y_i - b_{j_i}) \in Q_{j_i}, \qquad a \in [0, 1],$$

because Q_{j_i} is convex and contains both y_i and its center b_{j_i}. Thus each node remains in the box it occupied at entrance, and different nodes continue to occupy different boxes—i.e., K_{term} admits robust node-state symmetry breaking (Definition 6).

The family K_{term} is compact as a continuous image of $K_{\mathrm{ent}} \times [0, 1]$, and it is permutation invariant. Moreover,

$$\Psi(z) = b(y) + \frac{a}{2}(y - b(y)) \in K_{\mathrm{term}}(G),$$

where $K_{\mathrm{term}}(G) := \{w : (G, w) \in K_{\mathrm{term}}\}$. Thus every terminal update stays in this family, which contains all terminal iterates and their limits.

For a terminal state $z = b(y) + a(y - b(y))$ arising from an entrance state $y = y_{h(x)}(x)$, let $s = s(x)$ be that initialization's selected solution. The grid bound and (25) imply

$$\|z - y\| < \omega, \qquad \|z - b(y)\| < \omega, \qquad \|z - s\| \le r - \zeta + \omega < r. \tag{29}$$

These estimates hold also at both endpoints of the segment. In particular, the solution-set distance on K_{term} is below r, whereas it is at least $r + \zeta$ on K_{tr}. The two compact families, K_{term} and K_{tr}, are disjoint and, when both are nonempty, positively separated.

Set $K := K_{\mathrm{tr}} \cup K_{\mathrm{term}}$ and prescribe

$$F^*(G, y) := \begin{cases} T_G(y), & (G, y) \in K_{\mathrm{tr}}, \\ \Psi(y), & (G, y) \in K_{\mathrm{term}}. \end{cases} \tag{30}$$

The two pieces are continuous and equivariant, and their domains are disjoint compact subsets of K. Thus F^* is a well-defined continuous equivariant map on K.

Since K_{tr} and K_{term} admit robust node-state symmetry breaking, K admits robust state-based symmetry breaking, with address regions $\mathcal{A}_j = Q_j$. Apply Lemma 5 with $u = 0$, $v = 1$, and $F_1 = F^*$. It gives one MP-GNN satisfying

$$\mathrm{GNN}(G, y) = F^*(G, y) \qquad ((G, y) \in K). \tag{31}$$

The block is defined on the entire input domain by the extension steps in Lemma 5. Its proof extends the address encoder to all of $\mathbb{R}^{d_y}$ and uses the Tietze extension theorem to extend the decoder from the compact reconstruction image to its entire Euclidean ambient space. The resulting GNN is therefore defined for every $(G, y) \in \mathcal{G} \times \mathcal{Y}_n$, while its agreement with $F^\star$ is guaranteed on K. In particular, it already realizes Ψ on K_{term}, so no separate extension of Ψ is needed. Both regimes are implemented by the same fixed block using only the current input (G, y), without an additional time input or stopping flag.

Step 5: Verify the full trajectory and branch capture. Fix $x = (G, y_0) \in E$ and write $h = h(x)$. For $t < h$, each ideal update from the transient family leads either to its next transient state or to the entrance state. From the entrance state onward, Ψ stays in K_{term}. Thus exact realization and induction give

$$\widehat{y}_t = T_G^t(y_0) \quad (0 \le t \le h), \qquad \widehat{y}_{h+k} = b(\widehat{y}_h) + 2^{-k}\big(\widehat{y}_h - b(\widehat{y}_h)\big) \quad (k \ge 0). \tag{32}$$

This includes the case $h = 0$. The trajectory converges to $\widehat{y}_\infty = b(\widehat{y}_h)$, and (29) gives

$$\|\widehat{y}_\infty - s(G, y_0)\| \le r - \zeta + \omega < r < \varepsilon.$$

The learned update equals T_G on the transient. On the terminal family, $b(z) = b(y)$ for its generating entrance state, so

$$\|\Psi(z) - z\| = \tfrac{1}{2}\|z - b(z)\| < \omega/2.$$

Using (18) and (29), uniformly on K_{term},

$$\begin{aligned} \|\,\mathrm{GNN}(G, z) - T_G(z)\| &\le \|\Psi(z) - z\| + \|z - T_G(z)\| \\ &\le \tfrac{1}{2}\|z - b(z)\| + \|z - s\| \\ &\le \omega/2 + r - \zeta + \omega < r + \omega/2 < 2r < \rho. \end{aligned}$$

The right-hand bounds are constants independent of the trajectory and time, proving the supremum bound in (20).

The preceding argument establishes convergence and the uniform operator-error bound for every $(G, y_0) \in E$. In particular, every retained trajectory converges to a limit within ε of its selected solution, and hence of $S(G)$. Together with (26), this gives

$$\mathbb{P}(\widehat{y}_t \to \widehat{y}_\infty,\ \mathrm{dist}(\widehat{y}_\infty, S(G)) < \varepsilon) \ge \mathbb{P}(E) \ge 1 - \delta.$$

To verify branch capture, fix a label a. By construction, $C_a \subseteq B_a$, so every $(G, y_0) \in C_a \cap E$ satisfies

$$y_0 \in V_{f_a(G)}(G), \qquad s(G, y_0) = f_a(G).$$

The limit estimate therefore gives

$$\|\widehat{y}_\infty - f_a(G)\| = \|\widehat{y}_\infty - s(G, y_0)\| < \varepsilon \qquad ((G, y_0) \in C_a \cap E).$$

Since each restriction preserved positive probability in C_a,

$$\mathbb{P}\big(Y_0 \in V_{f_a(G)}(G),\ \widehat{y}_t \to \widehat{y}_\infty,\ \|\widehat{y}_\infty - f_a(G)\| < \varepsilon\big) \geq \mathbb{P}(C_a \cap E) > 0.$$

Thus every labeled branch has positive capture probability under the joint law of graphs and initializations.

Finally, the graph projection $K_{\varepsilon,\rho,\delta} := \mathrm{proj}_{\mathcal{G}} E$ is compact because it is the continuous image of the compact set E. Lemma 5 gives a single GNN that agrees with $F^\star$ on all of K. For every retained pair $(G, y_0) \in E$, the entire learned trajectory stays in K. Thus the same GNN, with all component maps fixed, generates every retained trajectory by repeated application. Moreover, the shared nodewise maps and sum aggregations commute with node relabeling, giving the permutation equivariance: $\mathrm{GNN}(\pi G, \pi y) = \pi\,\mathrm{GNN}(G, y)$. This concludes the proof. □

C An obstruction to almost-sure validity for contractive equivariant dynamics

We use the graph space $\mathcal{E}_n$, output space $\mathcal{Y}_n = \mathbb{R}^{n\times q}$, Euclidean norms, and permutation actions of Appendix A. In particular, $\pi y = P_\pi y$, and the graph domain $\mathcal{G} \subseteq \mathcal{E}_n$ is closed under node relabeling. The Euclidean norm on a matrix is its Frobenius norm. All continuity and neighborhood statements on $\mathcal{G}$ refer to its relative topology.

For a graph $G \in \mathcal{G}$, define its automorphism group and the corresponding fixed subspace by

$$\Gamma_G := \{\pi \in \mathfrak{S}_n : \pi G = G\}, \qquad \mathrm{Fix}(\Gamma_G) := \{y \in \mathcal{Y}_n : \pi y = y \text{ for every } \pi \in \Gamma_G\}.$$

The latter is a closed linear subspace, since

$$\mathrm{Fix}(\Gamma_G) = \bigcap_{\pi\in\Gamma_G} \ker\big(y \mapsto P_\pi y - y\big).$$

Let ν be a Borel probability measure on $\mathcal{G}$. We write $G_0 \in \mathrm{supp}\,\nu$ when every relatively open neighborhood of G_0 has positive ν-measure. Thus a full-support graph law satisfies this condition at every $G_0 \in \mathcal{G}$.

C.1 The operator-level obstruction

Theorem 6 (Positive-probability failure of continuous contractive equivariant dynamics)**.** *Let $S(G) = \{f_1(G), \dots, f_M(G)\}$ be a permutation-equivariant finite-branch solution map in the sense of Definition 1. For this theorem, assume additionally that every branch $f_a : \mathcal{G} \to \mathcal{Y}_n$ is continuous. Suppose there is a graph $G_0 \in \mathrm{supp}\,\nu$ such that*

$$S(G_0) \cap \mathrm{Fix}(\Gamma_{G_0}) = \varnothing. \tag{33}$$

Let

$$F : \mathcal{G} \times \mathcal{Y}_n \to \mathcal{Y}_n, \qquad (G, y) \mapsto F_G(y),$$

be jointly continuous and permutation equivariant. Assume that, for some $\kappa \in [0, 1)$,

$$\|F_G(y) - F_G(z)\| \leq \kappa\|y - z\|, \qquad F_{\pi G}(\pi y) = \pi F_G(y), \tag{34}$$

for all $G \in \mathcal{G}$, $y, z \in \mathcal{Y}_n$, and node permutations π. Then the following statements hold.

1. ***A unique, initialization-independent limit.*** *For every $G \in \mathcal{G}$, there is a unique fixed point $p_F(G) \in \mathcal{Y}_n$. Every trajectory $y_{t+1} = F_G(y_t)$ satisfies*

$$\|y_t - p_F(G)\| \leq \kappa^t\|y_0 - p_F(G)\|, \qquad t \geq 1. \tag{35}$$

The map $p_F : \mathcal{G} \to \mathcal{Y}_n$ is continuous and permutation equivariant. In particular,

$$p_F(G) \in \mathrm{Fix}(\Gamma_G) \qquad (G \in \mathcal{G}).$$

2. ***Failure on a neighborhood of the obstruction graph.*** *The number*

$$\gamma_0 := \min_{1\le a\le M} \operatorname{dist}\big(f_a(G_0), \operatorname{Fix}(\Gamma_{G_0})\big) \tag{36}$$

is strictly positive and does not depend on F. For every $\varepsilon \in (0,\gamma_0)$, there is a relatively open neighborhood $V_{F,\varepsilon}$ of G_0 such that

$$\operatorname{dist}\big(p_F(G), S(G)\big) > \varepsilon \qquad (G \in V_{F,\varepsilon}). \tag{37}$$

3. ***Positive-probability failure under random initialization.*** *Let (G, Y_0) have any Borel probability law $\mathbb{P}$ on $\mathcal{G}\times\mathcal{Y}_n$ with graph marginal ν, and define*

$$Y_{t+1} = F_G(Y_t), \qquad t \ge 0.$$

For every $\varepsilon \in (0,\gamma_0)$, set $\delta_{F,\varepsilon} := \nu(V_{F,\varepsilon}) > 0$. Then Y_t converges for every (G, Y_0), and its limit $Y_\infty = p_F(G)$ satisfies

$$\mathbb{P}(\operatorname{dist}(Y_\infty, S(G)) > \varepsilon) \ge \delta_{F,\varepsilon} > 0. \tag{38}$$

Consequently,

$$\mathbb{P}\big(Y_\infty \in S(G)\big) < 1.$$

No independence, density, or full-support assumption is needed for the initialization law.

Proof. We verify the three conclusions.

Step 1: Existence, uniqueness, and convergence for each graph. Fix $G \in \mathcal{G}$ and $y_0 \in \mathcal{Y}_n$. Repeated use of (34) gives

$$\|y_{t+1} - y_t\| \le \kappa^t \|y_1 - y_0\|.$$

For integers $r > s \ge 0$, it follows that

$$\|y_r - y_s\| \le \sum_{t=s}^{r-1} \kappa^t \|y_1 - y_0\|.$$

The right-hand side tends to zero as $s \to \infty$, uniformly in $r > s$, because the geometric series is summable. Hence (y_t) is Cauchy. The space $\mathcal{Y}_n$ is complete, so y_t has a limit p. Continuity of F_G yields

$$F_G(p) = \lim_{t\to\infty} F_G(y_t) = \lim_{t\to\infty} y_{t+1} = p.$$

If p and $\widetilde{p}$ are fixed points, then

$$\|p - \widetilde{p}\| \le \kappa \|p - \widetilde{p}\|,$$

which forces $p = \widetilde{p}$ because $\kappa < 1$. Write this unique fixed point as $p_F(G)$. Its uniqueness makes the limit independent of y_0, and iteration of

$$\|y_{t+1} - p_F(G)\| \le \kappa \|y_t - p_F(G)\|$$

proves (35). If $\kappa = 0$, the trajectory reaches the fixed point after one step.

Step 2: Continuity of the fixed-point map. For any $G, H \in \mathcal{G}$, the fixed-point equations and (34) imply

$$\begin{aligned}\|p_F(G) - p_F(H)\| &= \|F_G(p_F(G)) - F_H(p_F(H))\| \\ &\le \kappa \|p_F(G) - p_F(H)\| + \|F_G(p_F(H)) - F_H(p_F(H))\|.\end{aligned}$$

Therefore

$$\|p_F(G) - p_F(H)\| \le \frac{\|F_G(p_F(H)) - F_H(p_F(H))\|}{1-\kappa}. \tag{39}$$

Fixing H and letting $G \to H$, joint continuity of F makes the numerator tend to zero. This proves continuity of p_F on $\mathcal{G}$.

Step 3: Equivariance and preservation of graph automorphisms. By definition, $p_F(\pi G)$ is a fixed point of πG. For any node permutation π,

$$F_{\pi G}(\pi p_F(G)) = \pi F_G(p_F(G)) = \pi p_F(G)$$

which implies $\pi p_F(G)$ is also a fixed point of πG. By uniqueness of the fixed point at πG,

$$p_F(\pi G) = \pi p_F(G).$$

If $\pi \in \Gamma_G$, then $\pi G = G$, and consequently $\pi p_F(G) = p_F(G)$. Thus $p_F(G) \in \mathrm{Fix}(\Gamma_G)$.

Step 4: A positive error persists on a graph neighborhood. The fixed subspace $\mathrm{Fix}(\Gamma_{G_0})$ is closed. By (33), each of the finitely many points $f_a(G_0)$ lies outside this subspace. Its distance to the subspace is therefore strictly positive, and the finite minimum in (36) satisfies $\gamma_0 > 0$. Step 3 gives

$$\|p_F(G_0) - f_a(G_0)\| \geq \gamma_0 \qquad (1 \leq a \leq M).$$

Fix $\varepsilon \in (0, \gamma_0)$ and put $\eta := (\gamma_0 - \varepsilon)/2 > 0$. By continuity of p_F and of the finitely many branches, there is a relatively open neighborhood $V_{F,\varepsilon}$ of G_0 such that

$$\|p_F(G) - p_F(G_0)\| < \eta, \qquad \max_{1 \leq a \leq M} \|f_a(G) - f_a(G_0)\| < \eta \qquad (G \in V_{F,\varepsilon}).$$

For every such G and every a, the reverse triangle inequality gives

$$\begin{aligned}\|p_F(G) - f_a(G)\| &\geq \|p_F(G_0) - f_a(G_0)\| - \|p_F(G) - p_F(G_0)\| - \|f_a(G) - f_a(G_0)\| \\ &> \gamma_0 - 2\eta = \varepsilon.\end{aligned}$$

Taking the minimum over the finite branch set proves (37).

Step 5: The probability statement. Because $G_0 \in \operatorname{supp} \nu$, $\delta_{F,\varepsilon} = \nu(V_{F,\varepsilon}) > 0$. The function

$$G \longmapsto \mathrm{dist}(p_F(G), S(G)) = \min_{1 \leq a \leq M} \|p_F(G) - f_a(G)\|$$

is continuous, so the events in the statement are measurable. By Step 1, $Y_\infty = p_F(G)$ for every initial state. Consequently,

$$\{G \in V_{F,\varepsilon}\} \subseteq \{\mathrm{dist}(Y_\infty, S(G)) > \varepsilon\}.$$

Taking $\mathbb{P}$-probabilities and using its graph marginal ν proves (38). The last assertion follows because the event on the right contains no valid limit. □

Remark (Scope of the assumptions). The graph domain need not be compact or connected, and the graph law need not be permutation invariant. Full support of ν is sufficient but not necessary: the proof only uses $G_0 \in \operatorname{supp} \nu$. The neighborhood argument only needs continuity of the branches at G_0; continuity on $\mathcal{G}$ is imposed above to ensure that all global error events are measurable without additional assumptions. In particular, the branch assumption holds on a domain where the branches are locally Lipschitz, as in the regular stable setting of Appendix A. The failure is uniform over initial states on $V_{F,\varepsilon}$ only at the level of their limits; no common finite convergence time over the unbounded space $\mathcal{Y}_n$ is asserted.

C.2 Application to independently weighted triangles

Fix $0 < w_- < w_+ < \infty$ and let $W = [w_-, w_+]^3$. For $w = (w_{12}, w_{13}, w_{23}) \in W$, define

$$A(w) = \begin{pmatrix} 0 & w_{12} & w_{13} \\ w_{12} & 0 & w_{23} \\ w_{13} & w_{23} & 0 \end{pmatrix}, \qquad G(w) = (A(w), 0).$$

Here $n = 3$, $d_e = d_v = 1$, and $X = 0 \in \mathbb{R}^{3\times 1}$. Set

$$\mathcal{G}_\triangle := \{G(w) : w \in W\}, \qquad L_G := \mathrm{diag}(A\mathbf{1}) - A, \qquad H_G := I_3 + L_G.$$

We take $q = 2$ and write $y = [u\ b] \in \mathcal{Y}_3 = \mathbb{R}^{3\times 2}$, where $u, b \in \mathbb{R}^3$ are its two columns. The single-source diffusion solution set is

$$S(G) = \{f_1(G), f_2(G), f_3(G)\}, \qquad f_k(G) := [H_G^{-1} e_k \ \ e_k]. \tag{40}$$

Thus a valid output satisfies $H_G u = b$, with b a one-hot source indicator.

Corollary 1 (Impossibility under independent edge weights)**.** *Let the three edge weights be independent and uniform on $[w_-, w_+]$, and let ν be the induced law on $\mathcal{G}_\triangle$. Suppose $F : \mathcal{G}_\triangle \times \mathcal{Y}_3 \to \mathcal{Y}_3$ satisfies the continuity, equivariance, and global contraction assumptions of Theorem 6. For every*

$$0 < \varepsilon < \sqrt{\frac{2}{3}},$$

there is a number $\delta_{F,\varepsilon} > 0$ such that, under any joint law of (G, Y_0) with graph marginal ν, the recurrence converges and

$$\mathbb{P}(\mathrm{dist}(Y_\infty, S(G)) > \varepsilon) \geq \delta_{F,\varepsilon}.$$

In particular, no such operator converges to a valid solution almost surely over independently weighted graphs. The same conclusion holds for any full-support Borel law on $\mathcal{G}_\triangle$.

Proof. We verify the solution-map, support, and symmetry hypotheses.

Step 1: The graph family and solution branches. Relabeling the nodes permutes the three edge coordinates, so $\mathcal{G}_\triangle$ is permutation invariant. For every $v \in \mathbb{R}^3$,

$$v^\top H_G v = \|v\|_2^2 + \sum_{1 \leq i < j \leq 3} w_{ij}(v_i - v_j)^2 \geq \|v\|_2^2.$$

Hence H_G is invertible and $\|H_G^{-1}\|_{\mathrm{op}} \leq 1$. The map $G \mapsto H_G$ is linear in the edge matrix up to the constant identity. For any $G, \widetilde{G} \in \mathcal{G}_\triangle$, the resolvent identity gives

$$H_G^{-1} - H_{\widetilde{G}}^{-1} = H_G^{-1}(H_{\widetilde{G}} - H_G)H_{\widetilde{G}}^{-1}.$$

Consequently,

$$\|H_G^{-1} - H_{\widetilde{G}}^{-1}\|_{\mathrm{op}} \leq \|H_{\widetilde{G}} - H_G\|_{\mathrm{op}}.$$

Since linear maps between finite-dimensional Euclidean spaces are globally Lipschitz, every branch in (40) is globally Lipschitz.

Furthermore,

$$H_{\pi G} = P_\pi H_G P_\pi^\top, \qquad H_{\pi G}^{-1} = P_\pi H_G^{-1} P_\pi^\top.$$

Using $P_\pi e_k = e_{\pi(k)}$ yields

$$f_{\pi(k)}(\pi G) = [P_\pi H_G^{-1} e_k \;\; P_\pi e_k] = \pi f_k(G).$$

Thus Definition 1 holds with $\tau_\pi(k) = \pi(k)$. The branches are distinct for every graph because their second columns are distinct; indeed,

$$\|f_k(G) - f_\ell(G)\| \geq \|e_k - e_\ell\|_2 = \sqrt{2} \qquad (k \neq \ell).$$

Their collapse partition consists of three singleton sets throughout the domain. Hence $D = U = \varnothing$ and $(\mathcal{G}_\triangle)_\star = \mathcal{G}_\triangle$ in the notation of Definition 2.

Step 2: The graph law has full support. The map $w \mapsto G(w)$ is a homeomorphism from W onto $\mathcal{G}_\triangle$. In fact,

$$\|G(w) - G(\widetilde{w})\| = \sqrt{2}\,\|w - \widetilde{w}\|_2.$$

Every nonempty relatively open subset of the cube W has positive three-dimensional Lebesgue measure: it contains the intersection of W with a sufficiently small Euclidean ball, and this intersection has positive volume. The uniform product law has constant positive density on W. Its pushforward ν therefore has full support on $\mathcal{G}_\triangle$.

Step 3: A symmetric graph provides a fixed positive margin. Choose any $c \in (w_-, w_+)$ and set $G_0 = G(c, c, c)$. Every node permutation fixes G_0, so $\Gamma_{G_0} = \mathfrak{S}_3$ and

$$\mathrm{Fix}(\Gamma_{G_0}) = \{[\alpha\mathbf{1} \;\; \beta\mathbf{1}] : \alpha, \beta \in \mathbb{R}\}.$$

For any $k \in \{1, 2, 3\}$ and $\beta \in \mathbb{R}$,

$$\|e_k - \beta\mathbf{1}\|_2^2 = (1 - \beta)^2 + 2\beta^2 = 3\left(\beta - \frac{1}{3}\right)^2 + \frac{2}{3} \geq \frac{2}{3}.$$

It follows that

$$S(G_0) \cap \mathrm{Fix}(\Gamma_{G_0}) = \varnothing, \qquad \gamma_0 \geq \sqrt{\frac{2}{3}}.$$

Step 2 gives $G_0 \in \operatorname{supp} \nu$. Applying Theorem 6 proves the claimed probability bound for every $\varepsilon \in (0, \sqrt{2/3})$ and for any initialization law. The proof uses only full support of the graph law, so the last assertion follows as well. □

C.3 Consequence for recurrent message-passing GNNs

Corollary 2 (Contractive recurrent MP-GNNs)**.** *Consider a deterministic, autonomous, weight-tied MP-GNN block of the form (15), with continuous component maps and only y_t carried between recurrent calls. Suppose its complete update satisfies*

$$\| \mathrm{GNN}(G, y) - \mathrm{GNN}(G, z)\| \leq \kappa \|y - z\| \qquad (G \in \mathcal{G},\ y, z \in \mathcal{Y}_n)$$

for some $\kappa < 1$. Then the conclusions of Theorem 6 hold whenever its solution-map and graph-law hypotheses hold. In particular, Corollary 1 applies on $\mathcal{G}_\triangle$, for every finite number of hidden layers and every choice of finite hidden widths.

Proof. Set $F_G(y) := \mathrm{GNN}(G, y)$. The encoder is continuous in (G, y). Inductively, if all $h_i^{(\ell)}$ are continuous, then each message is continuous by continuity of M_ℓ, its finite sum is continuous, and $h_i^{(\ell+1)}$ is continuous by continuity of U_ℓ. The final finite sum and continuous output map preserve continuity. Thus F is jointly continuous.

Under the simultaneous relabeling $(G, y) \mapsto (\pi G, \pi y)$, the encoder satisfies $h^{(0)}_{\pi(i)}(\pi G, \pi y) = h_i^{(0)}(G, y)$. If this identity holds at layer ℓ, reindexing the sender sum by $j \mapsto \pi(j)$ gives the same identity for messages and then for layer $\ell + 1$. The final global sum is invariant, and the node outputs are permuted. Hence $F_{\pi G}(\pi y) = \pi F_G(y)$, as also stated in (16). The assumed bound is exactly the remaining contraction hypothesis. The two cited results now apply. □

Relation to the multi-attractor results. For the triangle family, the preceding proof established $(\mathcal{G}_\triangle)_\star = \mathcal{G}_\triangle$. Consequently, Theorem 4 applies. If the initialization law has a density and full support on $\mathcal{Y}_3$, every strict Voronoi basin has positive probability. Indeed, for each solution s, a sufficiently small open ball centered at s lies in $V_s(G)$ because the other solutions are finitely many and distinct. The ideal operator therefore has almost-sure validity and positive capture probability for every solution at every fixed graph in $\mathcal{G}_\triangle$.

By contrast, every contractive operator above has one limit $p_F(G)$ for each fixed G. Its limiting law under any initialization distribution is a point mass. It cannot reach two distinct solutions with positive probability at the same graph. Moreover, Corollary 1 rules out exact almost-sure validity under the independent-weight graph law, even if coverage of multiple solutions is not required.

D Single-source diffusion: protocol and additional results

Datasets. Each instance is a complete undirected graph on three nodes. Its fixed input consists only of the three positive edge weights; static node features are identically zero and omitted from the implemented encoder. Draw w_{12}, w_{23}, w_{31} independently from $\mathrm{Unif}[0.5, 1.5]$. Randomly permute node labels consistently with the adjacency matrix. No node indices, drawing coordinates, prescribed source, or exact diffusion fields are supplied as input features. We generate 500/100/100 train/validation/test instances.

Model architecture. Across the three models, we use the same architecture:

$$\begin{aligned} h_i^{(0)} &= \mathrm{enc}(y_i), \\ m_i^{(\ell)} &= \sum_{j \neq i} M_\ell\big(h_i^{(\ell)}, h_j^{(\ell)}, w_{ij}\big), \\ h_i^{(\ell+1)} &= U_\ell\big(h_i^{(\ell)}, m_i^{(\ell)}\big), \qquad \ell = 0, 1, \end{aligned}$$

$$[N_\theta(G,y)]_i = O(h_i^{(2)}), \qquad F_\theta(G,y) = P(ay + N_\theta(G,y)). \tag{41}$$

Each component is a two-affine-layer MLP with a width-32 hidden layer and a SiLU activation between the affine layers. Messages use sum aggregation over the two neighbors. Layers within a block have separate parameters, but the entire block is shared across recurrent time t. The scalar skip a is also shared over nodes, coordinates, and time. The output projection is $P([u\ b]) = [u\ \mathrm{clip}_{[0,1]}(b)]$; it does not impose the mass constraint or choose a source. Both columns of y_0 are i.i.d. $\mathcal{N}(0,1)$ samples, independently across nodes and trajectories.

Training. For supervised training, sample k_G uniformly from $\{1,2,3\}$ once per training graph, independently of the neural initialization, and retain $s_G = [H_G^{-1} e_{k_G}\ \ e_{k_G}]$ for every subsequent visit. The target is *not* resampled between updates or epochs. We train the model by minimizing the loss function $\|y_T - s_G\|_F^2/6$ with $T = 10$. For energy based training, the training loss is $E_G(u_T, b_T)$, where energy function E_G is defined in (3) and $T = 10$. Our multi-attractor model is directly trained with this energy function without constraints on its weights, while contractive models are trained over the same energy function but applying the Lipschitz control technique described below.

During training, a batch contains 32 different training graphs and eight starts per graph (256 trajectories); the final batch has 20 graphs (160 trajectories). With 500 graphs, there are 16 optimizer updates per epoch. We train 1,200 epochs (19,200 updates) with Adam, its default momentum parameters $(0.9, 0.999)$, no weight decay, and no gradient clipping. The initial learning rate is 10^{-3}, multiplied by 0.2 after epoch 600 and by 0.05 relative to the initial rate after epoch 960.

Lipschitz control. We enforce global contraction with respect to the recurrent state in the infinity norm. For a node-state array h, this means $\|h\|_\infty = \max_i \|h_i\|_\infty$. For a weight matrix, we use the corresponding induced norm $\|W\|_\infty = \max_r \sum_s |W_{rs}|$.

Let $\sigma_{\mathrm{S}}(r) = r/(1+e^{-r})$ denote SiLU. We use the fixed, conservative Lipschitz bound $c_{\mathrm{S}} := 1 + 1/e$, since $|\sigma_{\mathrm{S}}'(r)| \le 1 + |r|e^{-|r|} \le c_{\mathrm{S}}$ where $r \in \mathbb{R}$. Consequently, a two-affine-layer MLP $g(z) = W_2\sigma_{\mathrm{S}}(W_1 z + \beta_1) + \beta_2$ satisfies

$$\mathrm{Lip}_\infty(g) \le c_{\mathrm{S}} \|W_2\|_\infty \|W_1\|_\infty.$$

For a message MLP $M_\ell(h_i, h_j, w_{ij})$, the graph is held fixed when measuring state sensitivity. Thus its first-layer norm is taken only over the *combined* columns multiplying h_i and h_j; the edge-weight columns and all biases are excluded from this norm bound. In particular, L_{M_ℓ} bounds the joint Lipschitz constant in $[h_i; h_j]$, rather than a separate constant for each argument.

Concatenation uses a maximum in the infinity norm:

$$\|[h_i; h_j] - [\widetilde{h}_i; \widetilde{h}_j]\|_\infty = \max\{\|h_i - \widetilde{h}_i\|_\infty, \|h_j - \widetilde{h}_j\|_\infty\}.$$

Therefore, summing the two incoming messages gives a bound $2L_{M_\ell}$, with no additional factor for the two message arguments. The update MLP receives $[h_i; m_i]$, giving the layer bound $L_{U_\ell}\max\{1, 2L_{M_\ell}\}$. Writing L_{enc} and L_O for the encoder and output MLP bounds, the complete residual block satisfies

$$\mathrm{Lip}_y(N_\theta) \le L_O L_{\mathrm{enc}} \prod_{\ell=0}^{1} L_{U_\ell} \max\{1, 2L_{M_\ell}\}.$$

We impose the following constraints once after parameter initialization and immediately after every Adam update, before the next forward pass. Set $\rho = 0.9$, $c = c_{\mathrm{S}}^{-1/2}$, and $\eta = 1 - 10^{-5}$. First clip a to $[-\rho + 0.05, \rho - 0.05]$, then use this clipped value to set the output-layer budget. The absolute row-sum caps and resulting MLP bounds are

MLP	First-layer cap	Second-layer cap	Lipschitz bound
enc	ηc	ηc	$L_{\mathrm{enc}} \le \eta^2$
M_ℓ	$\eta c/2$	ηc	$L_{M_\ell} \le \eta^2/2$
U_ℓ	ηc	ηc	$L_{U_\ell} \le \eta^2$
O	ηc	$\eta(\rho - \lvert a\rvert)c$	$L_O \le \eta^2(\rho - \lvert a\rvert)$

For M_ℓ, the first-layer cap applies to the sum of absolute entries across both state-input blocks together, not to each block separately. All other caps apply to the full weight-matrix rows. For a constrained row or row segment w with cap $B > 0$, we rescale

$$w \leftarrow \frac{w}{\max\{1, \|w\|_1/B\}}.$$

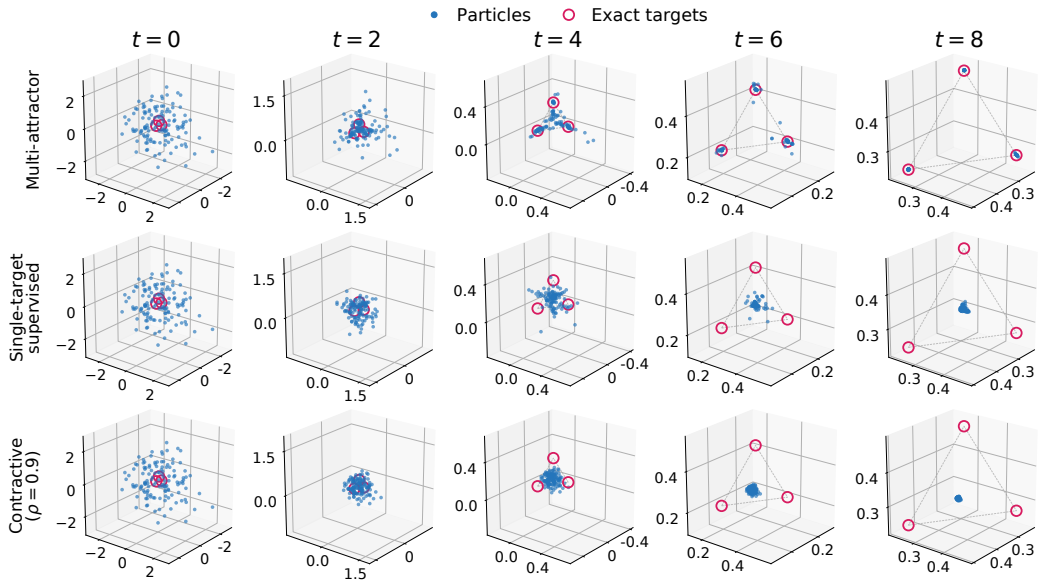


Figure 4: u-counterpart of Figure 3: the same test graph, seed, 128 trajectories, methods, and time steps. Blue points show $[(u_t)_1, (u_t)_2, (u_t)_3]$; magenta circles mark the three true solutions.

Thus rows already satisfying the cap are unchanged; oversized rows are scaled radially to the boundary. This projection is an optimizer-side operation, performed without differentiating through it.

Since $2L_{M_\ell} \leq \eta^2 < 1$, the two message-passing layers yield $\mathrm{Lip}_y(N_\theta) \leq \eta^8(\rho - |a|)$. The coordinatewise clipping map P is nonexpansive in the infinity norm, so the full update $F_\theta(G, y) = P(ay + N_\theta(G, y))$ satisfies

$$\mathrm{Lip}_y(F_\theta) \leq |a| + \eta^8(\rho - |a|) \leq \rho.$$

The constraints therefore certify contraction throughout training, rather than only after training or at sampled states. The multi-attractor and supervised models have no such projection. Single-target supervision itself does not guarantee a unique equilibrium.

Additional visualizations. Figure 3 in the main text shows the evolution of the source coordinates $b_t \in \mathbb{R}^3$. Since the full recurrent state is $y_t = [u_t\ b_t]$, Figure 4 complements this visualization by showing the diffusion coordinates $u_t \in \mathbb{R}^3$. The same pattern emerges: the multi-attractor trajectories form three distinct clusters, whereas the supervised and contractive controls each concentrate near a single point that remains visibly separated from all three exact diffusion solutions.

E Ising Experiments: Details and Additional Results

This section provides details and additional results regarding the Ising model experiments.

Datasets. We generate lattice graphs by `networkx.triangular_lattice_graph` with open boundaries, following Jore & Liu (2026). Each undirected edge receives an independent uniform coupling from $\{-0.5, -1, -1.5\}$; both directed orientations are stored with equal weight. We generate graphs with controlled sizes to assess scalability. We consider four levels of sizes (see Table 4). Every size has 2,500 train-source and 500 test graphs; the final 250 train-source graphs are held out for validation.

For each graph used in supervised training, Gurobi (Gurobi Optimization, LLC, 2026) provides one optimal spin configuration with a reported MIP gap of zero. Enumerating the full solution set is computationally infeasible. Labeling extra-large graphs can take more than 5,000 seconds per instance, so we omit supervised methods at that scale. Certified test references are still used to evaluate the extra-large graphs.

Table 4: Graph sizes.

Level	Nodes	Mean nodes	Edges	Mean edges
Compact	6–55	26.972	9–135	61.068
Medium	136–253	189.174	360–693	510.918
Large	561–990	770.580	1584–2838	2195.904
Extra-large	2016–3916	2965.198	5859–11484	8667.174

Model Architecture. For the recurrent dynamics $y_{t+1} = \mathrm{GNN}_\theta(G, y_t)$, we use a graph convolutional backbone augmented with physical node features.

At each step, we compute for each node i (here $N(i)$ represents the neighbors of i):

$$f_i = \sum_{j \in N(i)} J_{ij} y_j, \qquad a_i = \sum_{j \in N(i)} |J_{ij}|, \qquad q_i = f_i / \max(a_i, 10^{-8}), \tag{42}$$

$$c_i^{(\beta)} = \tanh\left[\frac{\sum_{j \in N(i)} \operatorname{atanh}(\mathrm{clip}(\tanh(\beta J_{ij}) y_j, -1 + 10^{-6}, 1 - 10^{-6}))}{\max(a_i, 10^{-8})} \right], \tag{43}$$

with $\beta \in \{0.5, 1, 2, 4\}$, giving the seven features $\phi_i = [f_i, q_i, a_i, c_i^{(0.5)}, c_i^{(1)}, c_i^{(2)}, c_i^{(4)}]$. Here, f_i is the local interaction field, a_i measures the total incident coupling strength, and q_i provides a normalized field. The four $c_i^{(\beta)}$ features provide nonlinear summaries of neighboring states at different coupling scales. Their functional form is inspired by the cavity equations for Ising belief propagation (Ricci-Tersenghi, 2012), with node states replacing directed cavity messages and an additional normalization by the incident coupling strength. Clipping keeps the argument of atanh within $(-1, 1)$, including for Gaussian initial states. These quantities are supplied as input features; they do not define a separate belief-propagation solver or prescribe the update direction.

Given these features, the recurrent cell $y^+ = F_\theta(G, y, \phi)$ is defined as follows. The encoder is

$$h_i^{(0)} = \mathrm{LayerNorm}(W_{e,2}\sigma(W_{e,1}[y_i, \phi_i] + b_{e,1}) + b_{e,2}),$$

with $\sigma = \mathrm{SiLU}$, dimensions $8 \to 16 \to 16$, and affine LayerNorm with epsilon 10^{-5}. No zero-padding coordinates or random identifiers are supplied. The three weighted GraphConv layers are

$$h_i^{(\ell+1)} = \sigma\left(W_{r,\ell} h_i^{(\ell)} + W_{n,\ell} \sum_{j \in N(i)} J_{ij} h_j^{(\ell)} + b_{n,\ell} \right), \qquad \ell = 0, 1, 2.$$

They use sum aggregation, a biased neighbor transform, and a bias-free root transform, without intermediate normalization or hidden residual connections. Independent $16 \to 16 \to 16$ SiLU MLPs produce $u_i = U(h_i^{(3)})$ and $v_i = R(h_i^{(3)})$. The readout combines local and graph-level information:

$$\bar{v} = \frac{1}{n} \sum_j v_j, \qquad \Delta_i = w_o^\top [u_i, \bar{v}] + b_o, \qquad y_i^+ = \tanh(y_i + \Delta_i). \tag{44}$$

Training. For minibatch $\mathcal{B}$ with $N_\mathcal{B} = \sum_{G \in \mathcal{B}} |V_G|$, the proposed model uses the following loss in training: $(1/N_\mathcal{B}) \sum_\mathcal{B} E_G(y_T)$ with $T = 10$. Training uses such node-weighted energy within each minibatch, whereas test energies are averaged equally across graphs. In particular, we use Adam for 200 epochs, LR 10^{-3}, default betas $(0.9, 0.999)$, epsilon 10^{-8}, and zero weight decay. There is no LR schedule, clipping, dropout, rollback, or early stopping. Graph batches are 32/32/4/1 for compact/medium/large/extra-large; each update uses one fresh trajectory per graph. The loader shuffles each epoch. The best validation checkpoint is saved after every completed epoch.

Baselines. We compare single-target supervision, unique-equilibrium models, untied feedforward networks, and learned solution distributions. All methods are evaluated on the same test graphs. Table 5 summarizes model capacities and requested training budgets; resource limits and completed budgets are reported below.

- *Single-target supervision.* We use the same architecture as our multi-attractor GNN, replacing the energy objective with MSE against one Gurobi-certified optimal spin configuration per training graph. Obtaining even one certified solution can exceed 5,000 seconds per graph at the extra-large scale, so we omit this supervised baseline at that scale. We allocate 2,000 training epochs and select the checkpoint with the lowest validation MSE. This extended budget, compared with 200 epochs for our energy-trained model, accommodates the slower optimization observed under single-target supervision: validation MSE continued to improve beyond 200 epochs.
- *Unique-equilibrium controls.* We adopt the implicit graph neural network (IGNN) of Gu et al. (2020), using its released implementation for the implicit layers and enforcing a contraction bound of $\rho = 0.9$ to ensure a unique equilibrium. To ensure fairness, we use the same physical-feature family and scalar spin-output interface as our model and is trained with the same energy objective. We evaluate hidden widths $H \in \{16, 32, 64\}$, including a capacity-comparable model and two larger variants, to assess whether increased capacity improves performance under the unique-equilibrium constraint.
- *Untied feedforward depth controls.* We stack independently parameterized copies of our recurrent block to obtain $y_{t+1} = \mathrm{GNN}_{\theta_t}(G, y_t)$, with distinct parameters θ_t at each stage. We consider 1, 3, and 10 blocks, corresponding to 3, 9, and 30 message-passing layers. These models retain the block architecture, physical features, Gaussian node initialization, and terminal energy objective of our model. They test whether increasing feedforward depth and parameter count can recover the benefits of recurrent weight sharing.
- *Distributional and diffusion-based models.* We include a mean-field GNN, an annealed Bernoulli GNN, DIFUSCO, and Variational Annealing on Graphs (VAG-CO). The *mean-field GNN* uses our width-16, three-layer backbone to map a fresh Gaussian initialization $y_0 \sim \mathcal{N}(0, I_n)$ to predicted spin means $y_1 \in (-1, 1)^n$ in one forward pass. It defines conditionally independent spins with $\Pr(s_i = +1 \mid G, y_0) = (1 + y_{1,i})/2$ and is trained by differentiating their analytic expected energy, $\mathbb{E}[E_G(s) \mid G, y_0] = E_G(y_1)$, following the training of Karalias & Loukas (2020). The otherwise identical *annealed Bernoulli GNN* additionally rewards conditional Bernoulli entropy, with its coefficient decreasing linearly from 0.5 to zero over 200 training epochs, following the annealed variational training of Sun et al. (2022). *DIFUSCO* (Sun & Yang, 2023) is adapted to Ising using binary categorical diffusion and a denoiser conditioned on noisy spins, diffusion time, and graph features. It is trained with clean-label cross entropy against one Gurobi-certified spin configuration and generates candidates by reverse diffusion from uniform random bits. *VAG-CO* (Sanokowski et al., 2023) generates spins autoregressively in a randomized breadth-first order, jointly sampling five spins at each decoding step. It is trained with an energy-minus-entropy objective using score-function gradients and a leave-one-out baseline, with temperature annealed during training. For both DIFUSCO and VAG-CO, the *matched* setting replaces their GNN backbone with our width-16, three-layer GraphConv backbone and local/global-mean readout, while retaining method-specific inputs and output heads. DIFUSCO additionally receives diffusion-time and structural features and predicts binary denoising probabilities. VAG-CO additionally receives partial spin assignments, assignment masks, and decoding priorities, and predicts a distribution over the 2^5 assignments of each five-spin block. These additions yield 3,250 and 5,904 parameters, respectively, compared with 3,153 for our model. The *large* settings adopt the model-size configurations from their original GitHub repositories: width 256 with 12 layers for DIFUSCO and width 120 with three layers for VAG-CO; see Table 5.

Resource limits. Structural models, matched DIFUSCO, and energy diffusion use training batches 32/32/4/1 by size; large DIFUSCO uses 8/8/1/1, and VAG uses 32/1/1/1. Training samples for each graph are processed in parallel, but VAG tokens are sequential. Large VAG is OOM on large and extra-large graphs even alone on a 24-GB GPU at graph batch one. Larger-graph training is capped at 30 accumulated wall-clock hours, including validation and retries; capped rows use the best validated checkpoint. Matched VAG completes 68/17/3 of 200 epochs at medium/large/extra-large sizes.

Metrics. We evaluate $K = 500$ test graphs, generating $M = 20$ candidate spin configurations per graph. Here, $g \in \{1, \ldots, K\}$ indexes test graphs and $m \in \{1, \ldots, M\}$ indexes candidates. Let $s_{g,m} \in \{-1, +1\}^{n_g}$ denote the mth candidate for graph G_g, where n_g is its number of nodes. For models with continuous outputs, we set $s_{g,m} = \mathrm{sign}_+(y_{g,m,T})$, where T is the evaluation horizon and $\mathrm{sign}_+(0) = +1$; discrete generators use their native spin outputs.

Table 5: Learned configurations. H: hidden width; L: total learned message passing depth.

Model	H	L	Parameters	LR	Epochs
Ours	16	3	3,153	10^{-3}	200
Recurrent MSE	16	3	3,153	10^{-3}	2000
IGNN adapter (H16)	16	3	2,851	10^{-3}	200
IGNN adapter (H32)	32	3	10,819	10^{-3}	200
IGNN adapter (H64)	64	3	42,115	10^{-3}	200
One-shot GNN	16	3	3,153	10^{-3}	200
Feedforward (3 blocks)	16	9	9,459	10^{-3}	200
Feedforward (10 blocks)	16	30	31,530	10^{-3}	200
Mean-field GNN	16	3	3,153	10^{-3}	200
Annealed Bernoulli GNN	16	3	3,153	10^{-3}	200
DIFUSCO (matched)	16	3	3,250	10^{-3}	200
DIFUSCO (large)	256	12	1,909,762	$2 \cdot 10^{-4}$	100
VAG (matched)	16	3	5,904	10^{-3}	200
VAG (large)	120	3	281,192	$5 \cdot 10^{-4}$	30

We measure energy per node, $e_{g,m} = E_{G_g}(s_{g,m})/n_g$. Let e_g^* be the Gurobi-certified optimal energy per node for G_g. The relative optimality gap is

$$\gamma_{g,m} = \max\left(0, \frac{e_{g,m} - e_g^*}{\max(|e_g^*|, \epsilon)}\right),$$

where $\epsilon > 0$ safeguards the denominator. We report three quality metrics:

$$\bar{e} = \frac{1}{KM}\sum_{g=1}^{K}\sum_{m=1}^{M} e_{g,m},\ \mathrm{Best}_{20} = \frac{1}{K}\sum_{g=1}^{K}\min_{1\le m\le M} e_{g,m},\ \mathrm{Hit10} = \frac{100}{KM}\sum_{g=1}^{K}\sum_{m=1}^{M}\mathbf{1}\{\gamma_{g,m} \le 0.10\}.$$

Thus, $\bar{e}$ averages energy over all candidates, Best_{20} averages the best candidate energy within each graph, and Hit10 is the percentage of candidates within 10% of the certified optimum. All graphs receive equal weight. For iterative models, candidates are evaluated at the terminal step; we do not select intermediate states along a trajectory.

For the experiment in Table 1, we measure diversity among optimal spin configurations, allowing a numerical tolerance of 10^{-6}:

$$D_M(g) = \left|\{s_{g,m} : 1 \le m \le M,\ \gamma_{g,m} \le 10^{-6}\}\right|, \qquad G_M = \frac{100}{K}\sum_{g=1}^{K}\mathbf{1}\{D_M(g) \ge 2\}.$$

Here, $D_M(g)$ counts the distinct optimal configurations found among M candidates for graph G_g. We report its graph-average $D_M = K^{-1}\sum_{g=1}^{K} D_M(g)$. The percentage G_M measures how often the model finds at least two different optimal configurations for the same graph, indicating its ability to recover multiple solutions.

For larger graphs, we relax the quality threshold and count distinct configurations within 10% of the certified optimum among 20 candidates:

$$V_{20}^{10\%} = \frac{1}{K}\sum_{g=1}^{K}\left|\{s_{g,m} : 1 \le m \le 20,\ \gamma_{g,m} \le 0.10 + 10^{-6}\}\right|.$$

For example, $V_{20}^{10\%} = 20$ means that every graph has 20 distinct candidates meeting this threshold. Repeated configurations count only once, while global spin flips are counted as distinct configurations. These metrics measure the diversity of sufficiently accurate outputs; they do not require the underlying trajectories to have converged.

Recurrence is numerically converged at T when $\sqrt{n^{-1}\|y_T - y_{T-1}\|_2^2} < 10^{-4}$; C_T is the percentage meeting this criterion. Among nonconverged trajectories, periods $p = 2, \ldots, 10$ are tested using

$$\left[\frac{1}{pn}\sum_{k=0}^{p-1}\|y_{T-k} - y_{T-k-p}\|_2^2\right]^{1/2} < 10^{-4},$$

assigning the smallest qualifying period. Remaining trajectories are unclassified at the tested horizon, not necessarily divergent or chaotic. Convergence is not applicable to feedforward or time-inhomogeneous generative models.

Longer test-time trajectories. Table 1 reports a convergence rate slightly below 100% at $t = 200$. To determine whether the remaining trajectories exhibit persistent oscillations or simply require more iterations, we extend the same 30,000 ($500 \times 20 \times 3$) trajectories across three training seeds to $t = 2000$, without changing the learned weights. Only 11 trajectories fail the numerical fixed-point criterion at $t = 200$; all satisfy it by the measured horizon $t = 500$ and remain numerically stationary at subsequent reported horizons through $t = 2000$ (Table 6). The extended checks also confirm sustained numerical stationarity at the final horizon, supporting the interpretation that these exceptions reflect longer transients.

Although training uses only $T = 10$ iterations, the learned update remains stable and settles to numerical fixed points when applied for substantially more iterations than seen during training. These are numerical observations rather than asymptotic guarantees. The longer runs serve only to examine convergence: all reported energy, Hit10, D_{20}, and G_{20} metrics use the terminal outputs at $t = 200$. Thus, the reported solution quality and diversity are already obtained by stopping at $t = 200$, without waiting for every trajectory to satisfy the convergence criterion.

Table 6: Ising test-time dynamics. Percentages are means ± population standard deviations across three training seeds.

t	C_t (%)	Period 2–10 (%)	Unclassified (%)
5	3.3733 ± 0.6370	—	—
10	32.5833 ± 2.0956	—	—
20	77.2733 ± 2.0129	0.0000 ± 0.0000	22.7267 ± 2.0129
50	97.7400 ± 0.2617	0.0000 ± 0.0000	2.2600 ± 0.2617
100	99.7100 ± 0.0638	0.0000 ± 0.0000	0.2900 ± 0.0638
200	99.9633 ± 0.0170	0.0000 ± 0.0000	0.0367 ± 0.0170
500	100.0000 ± 0.0000	0.0000 ± 0.0000	0.0000 ± 0.0000
1000	100.0000 ± 0.0000	0.0000 ± 0.0000	0.0000 ± 0.0000
2000	100.0000 ± 0.0000	0.0000 ± 0.0000	0.0000 ± 0.0000

Larger Graphs. Table 7 evaluates scalability on larger graphs while retaining the same width-16 architecture and 3,153 trainable parameters. Our model achieves the lowest mean and best-of-20 energies, the highest Hit10, and the largest number of distinct valid configurations among the evaluated methods at every scale. Its Hit10 reaches 96.88%, 99.96%, and 100% on medium, large, and extra-large graphs, respectively, with an average of 19.376, 19.992, and 20.000 distinct configurations within 10% of optimum among 20 candidates. Thus, a small shared recurrent model continues to produce both accurate and diverse solutions as graph size increases. This is consistent with the set-valued representation established by our theory: a single shared update can accommodate multiple solutions for the same input graph. Here, the empirical evidence concerns finite-horizon outputs, rather than convergence to distinct optimal attractors.

Increasing baseline capacity also helps in some cases, but does not consistently recover this combination of quality and diversity. The 3-block feedforward model substantially improves over one-shot prediction, and large DIFUSCO is competitive on medium and large graphs, yet both yield worse mean and best-of-20 energies than our model. Further increasing feedforward depth to 10 untied blocks reduces Hit10 and produces the same rounded configuration across all 20 initializations for each tested graph. These results support recurrent weight sharing as an effective way to scale solution generation without increasing parameter count, while the time-capped and OOM entries identify practical resource limits of the evaluated baseline configurations.

Table 7: Scaling to larger Ising graphs with fixed model capacity. The proposed model retains its width-16 architecture and 3,153 parameters across medium, large, and extra-large datasets, each containing 500 test graphs. Metrics use 20 candidates per graph: $\bar{e}$ and Best_{20} are mean and best-of-20 energies per node; Hit10 is the percentage of candidates within 10% of the certified optimum; and $V_{20}^{10\%}$ is the mean number of distinct configurations meeting this threshold. † marks training capped at 30 hours; OOM denotes a memory failure. ‡ marks supervised methods omitted because of training-label generation costs; certified test references remain available for evaluation.

Method	$\bar{e}\downarrow$	$\text{Best}_{20}\downarrow$	Hit10 (%) $\uparrow$	$V_{20}^{10\%}\uparrow$
Medium				
Ours	−1.2431	−1.2862	96.88	19.376
Recurrent MSE	1.5248	1.2913	0.00	0.000
IGNN adapter (H16)	0.5713	0.5713	0.00	0.000
IGNN adapter (H32)	−0.7917	−0.7917	0.00	0.000
IGNN adapter (H64)	0.0052	0.0052	0.00	0.000
One-shot GNN	−1.0970	−1.1792	1.60	0.320
Feedforward (3 blocks)	−1.1990	−1.2590	58.78	11.756
Feedforward (10 blocks)	−1.1966	−1.1966	56.60	0.566
Mean-field GNN	−1.0963	−1.1781	1.61	0.322
Annealed Bernoulli GNN	−1.1002	−1.1811	1.73	0.346
DIFUSCO (matched)	−1.0864	−1.2224	10.49	2.098
DIFUSCO (large)	−1.1934	−1.2653	55.25	11.050
VAG (matched)†	−0.9949	−1.0899	0.01	0.002
VAG (large)	0.0306	−0.2003	0.00	0.000
Large				
Ours	−1.2789	−1.3023	99.96	19.992
Recurrent MSE†	0.4071	−0.2762	0.00	0.000
IGNN adapter (H16)	0.0864	0.0864	0.00	0.000
IGNN adapter (H32)	−0.9817	−0.9817	0.00	0.000
IGNN adapter (H64)	0.1216	0.1216	0.00	0.000
One-shot GNN	−1.1296	−1.1722	0.00	0.000
Feedforward (3 blocks)	−1.2444	−1.2734	75.85	15.170
Feedforward (10 blocks)	−1.2289	−1.2289	41.00	0.410
Mean-field GNN	−1.1287	−1.1720	0.00	0.000
Annealed Bernoulli GNN	−1.1352	−1.1778	0.00	0.000
DIFUSCO (matched)	−1.1285	−1.2057	0.59	0.118
DIFUSCO (large)	−1.2514	−1.2876	80.26	16.052
VAG (matched)†	−0.9778	−1.0318	0.00	0.000
VAG (large)	OOM	OOM	OOM	OOM
Extra-large				
Ours	−1.3111	−1.3227	100.00	20.000
Recurrent MSE‡	–	–	–	–
IGNN adapter (H16)	0.2937	0.2937	0.00	0.000
IGNN adapter (H32)	−0.8143	−0.8143	0.00	0.000
IGNN adapter (H64)	2.9178	2.9178	0.00	0.000
One-shot GNN	−1.1443	−1.1665	0.00	0.000
Feedforward (3 blocks)	−1.2753	−1.2912	98.85	19.770
Feedforward (10 blocks)	−1.2390	−1.2390	5.00	0.050
Mean-field GNN	−1.1435	−1.1658	0.00	0.000
Annealed Bernoulli GNN	−1.1549	−1.1767	0.00	0.000
DIFUSCO (matched)‡	–	–	–	–
DIFUSCO (large)‡	–	–	–	–
VAG (matched)†	0.0980	0.0355	0.00	0.000
VAG (large)	OOM	OOM	OOM	OOM

F STRUCTURAL MODULE DETECTION IN PROTEIN GRAPHS: DETAILS AND ADDITIONAL RESULTS

This appendix gives the experiment details for Section 3.2.

Dataset. We download the PROTEINS dataset through PyTorch Geometric's `TUDataset`. For each graph we remove self-loops, collapse duplicate directed entries to one undirected edge, and use both directions only for message passing. The original graph classification label is retained for audit purposes but is never supplied to the models. The task graph contains 1,113 graphs, 43,471 nodes, and 81,044 undirected objective edges.

We compute exact graph-isomorphism groups, randomly order the groups with split seed 2027, and allocate complete groups to 80/10/10 splits. This gives 891 training, 111 validation, and 111 test graphs. No exact topology is shared across validation and test or with another split; duplicate topologies inside training remain grouped. Feature-normalization statistics are fit on the 891 training graphs only; standardized added topology features are clipped to $[-8, 8]$.

Table 8: Graph statistics. Entries for nodes and edges are min/mean/median/max.

Split	Graphs	Nodes	Undirected edges
Train	891	4 / 36.53 / 24 / 620	5 / 67.98 / 46 / 1049
Validation	111	5 / 52.56 / 31 / 504	6 / 98.33 / 61 / 894
Test	111	8 / 45.84 / 32 / 285	16 / 86.15 / 57 / 424

For exact references, binary x_{ik} assign node i to group k, $\sum_k x_{ik} = 1$. Auxiliary w_{ijk} linearize $x_{ik}x_{jk}$ on each observed edge, and $v_k = \sum_i d_i x_{ik}$. Gurobi maximizes

$$4m \sum_{\{i,j\}\in E} \sum_k w_{ijk} - \sum_k v_k^2, \tag{45}$$

whose value divided by $4m^2$ is Q_G. All 1,113 instances terminate with Gurobi's optimal status. One certified partition per graph is retained as the supervised target.

Node features. All neural models in this experiment receive the same node and edge features. We augment the original node attributes with standard structural descriptors drawn from local degree profiles (Cai & Wang, 2018), random-walk encodings (Dwivedi et al., 2022), and graph statistics documented in NetworkX (Hagberg et al., 2007). The input $s_i \in \mathbb{R}^{19}$ comprises five original-attribute/degree coordinates and fourteen additional structural coordinates:

$$s_i = [\,\widetilde{a}_i, \mathrm{onehot}_3(\ell_i), \widehat{d}_i, \mathcal{Z}_{\mathrm{node}}(\psi_i)\,].$$

Here a_i is the original continuous node attribute, ℓ_i is its original three-category node label, and $\widehat{d}_i = d_i / \max(1, \max_j d_j)$. The standardized attribute $\widetilde{a}_i$ uses the training-node mean and sample standard deviation (denominator floored at 10^{-8}). The original node labels are attributes, not target module assignments.

Let $N(i)$ be the neighbor set, $n = |V|$, $W_{ij} = A_{ij}/\max(d_i, 1)$, τ_i the triangle count at i, κ_i its core number, and π_i its PageRank. The fourteen raw structural coordinates are

$$\psi_i = \big[\,\log(1+d_i), (W^2)_{ii}, (W^3)_{ii}, (W^4)_{ii}, (W^8)_{ii}, c_i, \log(1+\tau_i), \frac{\kappa_i}{\max(1, \max_j \kappa_j)}, n\pi_i,$$
$$\log(1+\mu_i), \log(1+\sigma_i), \frac{|N_2(i)|}{\max(n-1,1)}, \frac{|\mathcal{C}(i)|}{n}, \frac{b_i}{\max(d_i,1)}\big]. \tag{46}$$

The four diagonal powers are random-walk return probabilities at lengths $2, 3, 4, 8$. $c_i = 2\tau_i/[d_i(d_i-1)]$ is the local clustering coefficient, set to zero when $d_i < 2$. The neighbor-degree mean and population standard deviation are μ_i and σ_i (both zero for an empty neighborhood). $N_2(i)$ contains nodes reachable in two hops after excluding i and its one-hop neighbors; $\mathcal{C}(i)$ is the connected component containing i; b_i counts incident bridges. Core numbers use the undirected simple graph. PageRank is approximated by 80 iterations from a uniform initialization, with damping .85, uniform teleportation, and uniform redistribution of dangling-node mass. The operator $\mathcal{Z}_{\mathrm{node}}$ standardizes each added coordinate using all training nodes, floors population standard deviations at 10^{-6}, and clips the result to $[-8, 8]$.

Edge features. We combine endpoint attributes with standard neighborhood-similarity and connectivity descriptors. For a directed message edge $j \to i$, set $C_{ji} = N(j) \cap N(i)$ and $U_{ji} = N(j) \cup N(i)$. The thirteen raw coordinates are

$$\chi_{ji} = \big[\log(1+d_j), \log(1+d_i), \widehat{d}_j, \widehat{d}_i, \mathbf{1}[\ell_j = \ell_i], |\widetilde{a}_j - \widetilde{a}_i|, \log(1+|C_{ji}|), \frac{|C_{ji}|}{\max(|U_{ji}|, 1)}, \\ \sum_{w \in C_{ji}} \frac{1}{\log(\max(d_w, 2) + 10^{-12})}, \sum_{w \in C_{ji}} \frac{1}{\max(d_w, 1)}, \mathbf{1}[\{j,i\} \text{ is a bridge}], \\ \frac{1}{\sqrt{\max(d_j d_i, 1)}}, \frac{|C_{ji}|}{\max(\min(d_j, d_i), 1)}\big].$$

The neighborhood-similarity terms include common-neighbor counts and Jaccard similarity (Liben-Nowell & Kleinberg, 2003), the Adamic–Adar index (Adamic & Adar, 2003), and the resource-allocation and hub-promoted indices (Zhou et al., 2009). The last coordinate is the hub-promoted index. Standard implementations of Jaccard, Adamic–Adar, and resource-allocation scores are also available in NetworkX (Hagberg et al., 2007). We use $e_{ji} = \mathcal{Z}_{\text{edge}}(\chi_{ji})$, with training-edge population moments, standard-deviation floor 10^{-6}, and clipping to $[-8, 8]$. Both edge orientations are included in message passing. All added features are deterministic functions of graph topology and original attributes. They contain no node identifiers, target partitions, or reference optimizer outputs, and do not depend on the current assignment state.

State and initialization. The recurrent variable is the soft partition $P_t \in \mathbb{R}^{n \times 4}$. For each trajectory we independently draw node-wise and graph-wise Gaussian logits and initialize

$$P_{0,i} = \text{softmax}\left(\frac{\epsilon_i + b_G}{\sqrt{2}}\right), \qquad \epsilon_i, b_G \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_4). \tag{47}$$

The same b_G is shared by the nodes of a graph within a trajectory, but is redrawn for each trajectory.

Message-passing cell. We use a width-32, three-layer PNA-style cell (Corso et al., 2020). In particular, let $\sigma = \text{ReLU}$, at each time step the hidden embeddings are recomputed as

$$h_i^{(0)} = \text{LayerNorm}\left(W_{e,2}\sigma(W_{e,1}[s_i, P_{t,i}] + b_{e,1}) + b_{e,2}\right),$$

with encoder dimensions $23 \to 32 \to 32$. The state-pair edge vector is

$$r_{ji}(P_t) = [P_{t,j}, P_{t,i}, P_{t,j} \odot P_{t,i}, |P_{t,j} - P_{t,i}|] \in \mathbb{R}^{16}. \tag{48}$$

For $\ell = 0, 1, 2$, compute

$$\begin{aligned} m_{ji}^{(\ell)} &= M_\ell([h_j^{(\ell)}, h_i^{(\ell)}, e_{ji}, r_{ji}(P_t)]), \\ a_i^{(\ell)} &= [\,\text{mean}_{j \in N(i)}\, m_{ji}^{(\ell)}, \text{std}_{j \in N(i)}\, m_{ji}^{(\ell)}, \max_{j \in N(i)}\, m_{ji}^{(\ell)}, \text{sum}_{j \in N(i)}\, m_{ji}^{(\ell)}\,], \\ h_i^{(\ell+1)} &= \text{LayerNorm}\left(h_i^{(\ell)} + U_\ell([h_i^{(\ell)}, a_i^{(\ell)}, P_{t,i}])\right). \end{aligned} \tag{49}$$

M_ℓ is a $93 \to 32 \to 32$ MLP with ReLU after both affine maps; U_ℓ is a $164 \to 32 \to 32$ MLP with ReLU only between affine maps. Aggregations are elementwise.

Global readout and update. The graph bottleneck and correction head are

$$g_G = \frac{1}{n}\sum_i R(h_i^{(3)}) \in \mathbb{R}^4, \qquad \Delta_{\theta,i} = O([h_i^{(3)}, g_G]), \tag{50}$$

where R is a $32 \to 32 \to 4$ MLP and O is a $36 \to 32 \to 4$ MLP, each with ReLU between its two affine maps. We update

$$P_{t+1} = \text{softmax}\big[\log(P_t \vee 10^{-6}) + \Delta_\theta(G, X, P_t)\big], \tag{51}$$

with row-wise softmax and coordinate-wise clipping before the logarithm. The three MP layers have distinct parameters within a cell; the entire cell is tied across time. There are 35,784 trainable parameters: 1,888 in the encoder, $3 \times 10,464$ in message passing, 1,188 in the global map, and 1,316 in the output head. The full model sizes are provided in Table 9.

Training and objectives. For an undirected graph $G = (V, E)$ with $m = |E|$ edges and node degrees d_i, let $z_i \in \{1, \dots, K\}$ denote the module of node i. At resolution $\gamma = 1$, we minimize the negative modularity:

$$E_G(z) = -Q_G(z) = \sum_{k=1}^{K} \left(\frac{\sum_i d_i \mathbf{1}[z_i = k]}{2m} \right)^2 - \frac{1}{m} \sum_{\{i,j\} \in E} \mathbf{1}[z_i = z_j].$$

The edge term rewards placing connected nodes in the same module; alone, it is minimized by assigning all nodes to one module. The degree-volume term accounts for the within-module connectivity expected from node degrees under the modularity null model. Combining the two terms therefore favors modules with more internal connectivity than this baseline predicts.

For a soft partition P, where each P_i is a probability vector over the K modules, we define the differentiable energy

$$e_G(P) = -\frac{1}{m} \sum_{\{i,j\} \in E} \langle P_i, P_j \rangle + \frac{1}{4m^2} \left[\left\| \sum_i d_i P_i \right\|_2^2 - \sum_i d_i^2 \|P_i\|_2^2 \right]. \tag{52}$$

Specifically, for independent assignments $Z_i \sim \mathrm{Categorical}(P_i)$,

$$e_G(P) = \mathbb{E}[E_G(Z)] - \frac{1}{4m^2} \sum_i d_i^2.$$

Thus, the soft objective equals the expected discrete energy up to a graph-dependent constant.

During training, each update samples ten training graphs with replacement and four independent initial soft partitions P_0 per graph. We unroll $T = 5$ tied cells and minimize $\mathbb{E}[e_G(P_T)]$. We use Adam with learning rate 3×10^{-4}, $\beta_1 = .9$, $\beta_2 = .999$, $\epsilon = 10^{-8}$, zero weight decay, global gradient-norm clipping at 1, 3,500 optimizer updates, batch size ten base graphs, and no learning-rate decay. The final update-3,500 checkpoint is used; there is no best-test or best-validation checkpoint substitution. The canonical models and all baselines in Table 2 use training seeds 0, 1, and 2. The unsupervised model never receives the Gurobi labels.

Baselines. Table 9 summarizes the learned configurations used in the reported comparisons. In this PROTEINS experiment, we use the same baseline groups with the Ising experiment. In particular,

- *Single-target supervision.* We use the same architecture as our multi-attractor GNN, replacing the energy objective $e_G(P)$ with cross entropy against Gurobi optimal solution per graph.
- *Unique-equilibrium models.* Similarly to the Ising experiments, We adopt the IGNN with hidden widths $H \in \{16, 32, 64\}$ and enforce a contraction bound of $\rho = 0.9$. To train IGNN, we use the same unsupervised loss function $\mathbb{E}[e_G(P_T)]$. Again, the input features are the same with the multi-attractor setting.
- *Untied feedforward depth controls.* The feedforward controls replace the same three-layer rich-PNA cell by $B \in \{1, 2, 3, 4, 5\}$ independent copies. They receive the same random P_0, losses, four training trajectories, optimizer, and 3,500-update budget.
- *Distributional and diffusion-based models.* We include the same distributional models with Ising experiments: a mean-field GNN, an annealed Bernoulli GNN, DIFUSCO, and Variational Annealing on Graphs (VAG-CO). They are still applicable here. "Matched" setting uses the same model backbone, learning rate, and training updates. "Large" setting uses the upstream original repository's configurations. Details are in Table 9.

Metrics. The caption of Table 2 defines every displayed metric.

Longer test-time trajectories. Table 2 reports a one-step convergence rate of $C_{200} = 88.05 \pm 2.23\%$ at $t = 200$. To determine whether the remaining trajectories exhibit persistent oscillations or simply require more iterations, we extend the same 6,660 ($111 \times 20 \times 3$) trajectories across three training seeds to $t = 10000$, without changing the learned weights. Here t denotes test-time recurrence steps, and results are reported in Table 10.

The convergence rate increases to $C_{800} = 98.75 \pm 0.62\%$ and $C_{3000} = 99.49 \pm 0.29\%$. By $t = 5000$, every trajectory is either convergent or classified as periodic at the reported horizons, with 6,632

Table 9: Learned configurations on PROTEINS. H: hidden width; L: total untied depth.

Model	H	L	Parameters	LR	Updates
Ours	32	3	35,784	$3 \cdot 10^{-4}$	3,500
Single-target Supervision	32	3	35,784	$3 \cdot 10^{-4}$	3,500
IGNN (H16)	16	3	11,340	$3 \cdot 10^{-4}$	3,500
IGNN (H32)	32	3	39,564	$3 \cdot 10^{-4}$	3,500
IGNN (H64)	64	3	146,700	$3 \cdot 10^{-4}$	3,500
Feedforward ($\times 1$)	32	3	35,784	$3 \cdot 10^{-4}$	3,500
Feedforward ($\times 2$)	32	6	71,568	$3 \cdot 10^{-4}$	3,500
Feedforward ($\times 3$)	32	9	107,352	$3 \cdot 10^{-4}$	3,500
Feedforward ($\times 4$)	32	12	143,136	$3 \cdot 10^{-4}$	3,500
Feedforward ($\times 5$)	32	15	178,920	$3 \cdot 10^{-4}$	3,500
Mean-field	32	3	35,784	$3 \cdot 10^{-4}$	3,500
Annealed Bernoulli	32	3	35,784	$3 \cdot 10^{-4}$	3,500
DIFUSCO (matched)	32	3	35,880	$3 \cdot 10^{-4}$	3,500
DIFUSCO (large)	256	12	5,555,466	$2 \cdot 10^{-4}$	2,800
VAG (matched)	32	3	40,270	$3 \cdot 10^{-4}$	3,500
VAG (large)	40	3	101,790	$5 \cdot 10^{-4}$	10,000

trajectories satisfying the numerical convergence criterion. The extended checks confirm that the vast majority of non-converged trajectories at $t = 200$ simply reflect longer transients, while a small remainder settles into stable periodicity rather than divergent behavior.

Although training uses only $T = 5$ unrolled iterations, the learned update remains stable and largely settles to numerical fixed points when applied for substantially more iterations than seen during training. The longer runs serve only to examine convergence: all reported modularity, Hit10, and discovery metrics use the terminal outputs at $t = 200$. Thus, the reported solution quality and diversity are already obtained by stopping at $t = 200$, without waiting for every trajectory to strictly satisfy the convergence criterion.

Table 10: PROTEINS test-time dynamics. Metrics are the same with Appendix E, Table 6.

t	C_t (%)	Period 2-10 (%)	Unclassified (%)
10	3.83 ± 1.78	–	–
20	16.65 ± 3.45	–	–
50	47.18 ± 1.92	–	–
100	68.72 ± 0.82	–	–
200	88.05 ± 2.23	0.02 ± 0.02	11.94 ± 2.25
500	97.51 ± 0.45	0.39 ± 0.28	2.10 ± 0.34
800	98.75 ± 0.62	0.42 ± 0.28	0.83 ± 0.48
1000	99.07 ± 0.30	0.42 ± 0.28	0.51 ± 0.24
2000	99.49 ± 0.31	0.42 ± 0.28	0.09 ± 0.07
3000	99.49 ± 0.29	0.42 ± 0.28	0.09 ± 0.04
5000	99.58 ± 0.28	0.42 ± 0.28	0.00 ± 0.00
10000	99.58 ± 0.28	0.42 ± 0.28	0.00 ± 0.00

G CHEMICAL REACTION NETWORKS: DETAILS AND ADDITIONAL RESULTS

This appendix provides the experiment details and additional results for Section 3.3.

Dataset. We construct kinetic instances on the published reaction-network structures from (Yao et al., 2025). We select networks labeled `Multi` from the Atom and Joshi construction families, preserving their original train/validation/test assignments. These labels indicate structural capacity for multistationarity but do not provide kinetic rates or stationary concentrations. For each selected structure, we add one inflow $\varnothing \to X_i$ and one outflow $X_i \to \varnothing$ per species. We then randomly propose two positive concentration vectors $c^{(1)}, c^{(2)}$ in log space and solve a linear feasibility prob-

lem for shared positive rate constants κ satisfying $f_G(c^{(1)}) = f_G(c^{(2)}) = 0$; the equations are linear in κ when the concentrations are fixed. We retain instances whose two witnesses have normalized residuals at most 10^{-7}, concentration-Jacobian condition numbers at most 10^9, and log-RMS separation at least 0.25. After feasibility filtering, we balance the two construction families within each split, obtaining 1,242 training, 266 validation, and 236 test instances, with 2–5 species and 6–19 reactions. Each instance contains the network structure, constructed rate constants, and two numerically verified stationary witnesses, which need not exhaust its solution set. Physics-only training uses only (α, β, κ); single-target supervision and supervised diffusion use one witness selected once and held fixed as the label for each training instance.

Table 11: Dataset construction. Balance is enforced within each split after feasibility filtering. Every retained instance has two constructed witnesses.

Split	Atom attempts	Joshi attempts	Atom feasible	Joshi feasible	Retained	Reactions
Train	790	2560	771	621	1242	6–19
Validation	180	570	172	133	266	6–18
Test	300	600	293	118	236	6–18

Nodes and edges. There is one node per species and one per reaction, including inflow/outflow reactions. For every pair (i, r) with $\alpha_{ri} > 0$ or $\beta_{ri} > 0$, we include both directed incidence edges. There is no extra node for $\varnothing$, no stoichiometric-copy expansion, and no direct species–species edge. The six static node coordinates are

$$\begin{aligned} x_i &= (1, 0, 0, 0, 0, 0), && \text{species,} \\ x_r &= \left(0, 1, \log \kappa_r, \sum_i \alpha_{ri}, \sum_i \beta_{ri}, \sum_i |\beta_{ri} - \alpha_{ri}|\right), && \text{reaction.} \end{aligned} \tag{53}$$

The first two coordinates distinguish node types; the remaining coordinates encode the log rate, total reactant/product orders, and net stoichiometric magnitude. Apart from the log-rate transform, these graph inputs are used directly, without dataset-fitted standardization. The four edge coordinates are

$$e_{i\to r} = (\alpha_{ri}, \beta_{ri}, 1, 0), \qquad e_{r\to i} = (\alpha_{ri}, \beta_{ri}, 0, 1).$$

The last two entries encode message direction, not reactant versus product. A species can be both a reactant and a product, with its two coefficients retained on the same incidence pair. Reaction roles are thus edge attributes, not a single global species label. All compared neural models receive these same base graph features.

State and initialization. The recurrent state is only the species vector $z_t = \log c_t \in [-20, 20]^s$. With width $w = 32$, compute the static encoding

$$b_v = \tanh\big(\operatorname{LayerNorm}(W_x x_v + b_x)\big).$$

Each candidate draws independent node vectors $\xi_v \sim \mathcal{N}(0, I_8)$. A bias-free $8 \to 32$ map and a scalar affine head initialize

$$z_{0,i} = \operatorname{clip}_{[-20,20]}\left(w_{\text{init}}^{\mathsf{T}}(b_i + W_\xi \xi_i) + b_{\text{init}}\right).$$

Subsequent calls receive no identifier forcing. At each call, reconstruct a temporary node matrix $H^{(0)}$ from b, replace coordinate zero of each species row with its current z_i, and leave reaction rows equal to b_r. All temporary hidden coordinates are discarded after the call except the new species log concentrations. There is no auxiliary carried hidden state.

Message-passing backbone. One update contains three distinct learned GATv2 (Brody et al., 2022) blocks; the entire stack is tied across recurrent iterations. For $\ell = 1, 2, 3$, compute

$$U^{(\ell)} = \operatorname{LayerNorm}_\ell\left(\operatorname{GATv2}_\ell(H^{(\ell-1)}, E, e)\right), \qquad H^{(\ell)} = \begin{cases} \tanh U^{(\ell)}, & \ell < 3, \\ U^{(\ell)}, & \ell = 3. \end{cases}$$

Each block is PyG `GATv2Conv(32,32,heads=4,concat=False,edge_dim=4)`. Attention heads are averaged, not concatenated. Self-loops are enabled, with mean incident edge features for their attributes; these internal loops are not chemical reactions. Other settings are zero attention

dropout, LeakyReLU slope 0.2, separate source/target transforms, and biases enabled. LayerNorm is affine with epsilon 10^{-5}. There is no spectral constraint on these attention blocks.

Dynamic physical features. For $c = \exp z$, compute $\ell_r = \log \kappa_r + \sum_i \alpha_{ri} z_i$ and protected fluxes $\widetilde{v}_r = \exp(\mathrm{clip}_{[-30,30]}(\ell_r))$. Set $\widetilde{a} = |N|\widetilde{v}$ and $\widetilde{q}_i = (N\widetilde{v})_i/(\widetilde{a}_i + 10^{-12})$. The node-wise dynamic features are

$$\psi_i = (z_i, \widetilde{q}_i, \log(\widetilde{a}_i + 10^{-12})), \qquad \psi_r = (\ell_r, 0, 0).$$

A shared affine $3 \to 32$ map followed by $\tanh$ encodes ψ_v. The reaction log-flux feature itself is not clipped; clipping protects the exponentials used to construct the internal residual. The independent evaluation residual uses original coefficients and unclipped mass-action fluxes.

Gated readout and update. A bias-free conditioning map W_b and the physical-feature encoder give

$$\widehat{H} = \mathrm{GRUCell}\Big(\tanh\Big[H^{(3)} + W_b b + \tanh(W_\psi \psi + b_\psi)\Big], H^{(0)}\Big).$$

This width-32 GRU (Cho et al., 2014) is a within-call gating computation, not an additional inter-iteration memory. Three scalar affine heads applied to each species row of $\widehat{H}$ produce

$$d_i = \tfrac{1}{2}\sigma(w_d^\mathsf{T}\widehat{H}_i + b_d), \qquad g_i = 2\tanh(w_g^\mathsf{T}\widehat{H}_i + b_g), \qquad a_i^{\mathrm{NN}} = \tanh(w_a^\mathsf{T}\widehat{H}_i + b_a).$$

The update is

$$z_i^+ = \mathrm{clip}_{[-20,20]}\left[z_i + d_i\left(g_i\widetilde{q}_i + \tanh(\widetilde{R})a_i^{\mathrm{NN}}\right)\right], \qquad \widetilde{R} = \sqrt{s^{-1}\sum_i \widetilde{q}_i^2 + 10^{-24}}. \tag{54}$$

The signed-gain head starts with zero weights and bias $\mathrm{atanh}(1/2)$, so its initial gain is one. The model has 35,716 trainable parameters. It is a physics-assisted learned solver, not a generic GNN with only a physics loss, and is not guaranteed contractive. All aggregation/readout operations are permutation-equivariant. Initialization is equivariant when ξ is permuted with the nodes, and equivariant in distribution under independent identifier sampling. After initialization, the shared map depends only on G and z.

Training and residual objective. Using the original mass-action fluxes, define

$$q_i(c) = \frac{(Nv(c))_i}{(|N|v(c))_i + 10^{-12}}, \;\; R(c) = \sqrt{\frac{1}{s}\sum_i q_i(c)^2}, \;\; E_{\mathrm{res}}(c) = \frac{1}{s}\sum_i \phi_{0.1}(q_i(c)), \tag{55}$$

where $\phi_{0.1}(u) = u^2/0.2$ when $|u| < 0.1$, and $|u| - 0.05$ otherwise. We train the model by minimizing $\mathbb{E}[E_{\mathrm{res}}(c_T)]$. In particular, we use AdamW with LR 0.002, weight decay 10^{-5}, default $(\beta_1, \beta_2) = (0.9, 0.999)$, epsilon 10^{-8}, gradient-norm clipping at 5, at most 50 epochs, and early-stopping patience 12. Accumulate gradients over four graphs, with 20 candidate initializations per graph. Graph's recurrent unroll length T is sampled uniformly from $\{4, 8, 16, 24, 32\}$.

Baselines. We retain the four baseline groups used in the Ising and PROTEINS experiments: single-target supervision, unique-equilibrium models, untied feedforward depth controls, and distributional models. Table 12 summarizes their configurations. All models receive the same base graph inputs $x_i, x_r, e_{i\to r}, e_{r\to i}$ and are evaluated on the same 236 test graphs.

- *Single-target supervision.* We use the same architecture as our multi-attractor model, with the same random initialization, randomized unroll lengths, optimizer, and stopping rule, but replace the residual objective by the concentration-space MSE $s^{-1}\|c_T - c^{\mathrm{target}}\|_2^2$. For each training instance, c^{target} is drawn uniformly from the two witnesses once and then held fixed throughout training. The supervised diffusion models below use the same label.
- *Unique-equilibrium controls.* As in the other two experiments, we adopt IGNN (Gu et al., 2020) and enforce a contraction bound $\rho = 0.9$, which guarantees a unique equilibrium. It is trained with the same residual objective $\mathbb{E}[E_{\mathrm{res}}]$ as our model. We evaluate hidden widths $H \in \{16, 32, 64\}$, which include a capacity-comparable model ($H = 32$) and a smaller and a larger variant. Because its equilibrium is uniquely determined by G, IGNN produces a single candidate per graph; replicated copies are not counted toward diversity.

- *Untied feedforward depth controls.* We replace the tied recurrence by $B \in \{1, 2, 3, 4, 5\}$ independently parameterized copies of our update, $z_{t+1} = F_{\theta_t}(G, z_t)$, with distinct parameters θ_t at each stage. Each stage has its own input encoder, three GATv2 blocks, GRU, physical-feature encoder, and scalar heads. Only the identifier projection and initialization head, which produce z_0, are shared. The model thus has $3B$ message-passing blocks and $289 + 35{,}427B$ parameters, and it executes exactly B stages at both training and test time. It uses the same random initialization, 20 candidates per graph, residual objective, and optimizer as our model. These controls test whether increasing untied depth and parameter count can recover the benefits of recurrent weight sharing. None is iterated beyond its trained depth, so no settling rate is reported.
- *Distributional models.* Since concentrations are continuous, we replace the discrete generators used for Ising and PROTEINS by DDPM (Ho et al., 2020), DDIM (Song et al., 2021), and the time-reversed diffusion sampler (DIS) (Berner et al., 2024). DDPM is trained with the standard noise-prediction loss against the same fixed witness used for single-target supervision, after mapping log concentrations to the unbounded latent $u = 20\,\mathrm{atanh}(\log c/20)$. It uses 1,000 linear-beta noise levels and 100 reverse steps. DDIM shares the trained DDPM checkpoint and samples deterministically ($\eta = 0$) with 100 steps. DIS uses no witness. Instead, it learns to sample from the residual-based density $\propto \exp[-E_{\mathrm{res}}(c)/\tau]$ with $\tau = 10^{-4}$, using 100 steps. The *matched* setting keeps our width-32, three-block GATv2 backbone, GRU, physical-feature encoder, and heads. The only change is that the input encoder also receives the noisy species state and the normalized diffusion time, giving 35,780 parameters compared with 35,716 for our model. It uses the same learning rate, 50-epoch budget, and early-stopping patience as our model. The *large* setting follows the upstream repositories. DDPM/DDIM use DIFUSCO's gated residual graph backbone with width 256 and 12 layers (5,205,249 parameters). DIS uses the original FourierMLP controller, conditioned on a width-32, three-block GAT encoder (59,330 parameters), and is trained with its native log-variance loss. Every generator returns 20 candidates per graph after 100 sampling steps.

Table 12: Learned configurations for chemistry. H: hidden width; L: total untied depth.

Model	H	L	Parameters	LR	Epochs
Ours	32	3	35,716	$2 \cdot 10^{-3}$	50
Single-target Supervision	32	3	35,716	$2 \cdot 10^{-3}$	50
IGNN / H=16	16	3	10,273	$2 \cdot 10^{-3}$	50
IGNN / H=32	32	3	37,441	$2 \cdot 10^{-3}$	50
IGNN / H=64	64	3	142,465	$2 \cdot 10^{-3}$	50
Feedforward ($\times 1$)	32	3	35,716	$2 \cdot 10^{-3}$	50
Feedforward ($\times 2$)	32	6	71,143	$2 \cdot 10^{-3}$	50
Feedforward ($\times 3$)	32	9	106,570	$2 \cdot 10^{-3}$	50
Feedforward ($\times 4$)	32	12	141,997	$2 \cdot 10^{-3}$	50
Feedforward ($\times 5$)	32	15	177,424	$2 \cdot 10^{-3}$	50
DDPM / matched	32	3	35,780	$2 \cdot 10^{-3}$	50
DDIM / matched	32	3	35,780	$2 \cdot 10^{-3}$	50
DIS / matched	32	3	35,780	$2 \cdot 10^{-3}$	50
DDPM / large	256	12	5,205,249	$2 \cdot 10^{-4}$	200
DDIM / large	256	12	5,205,249	$2 \cdot 10^{-4}$	200
DIS / large	32/64	3/4	59,330	$5 \cdot 10^{-3}$	100

Metrics. All metrics use the $n = 236$ test graphs with equal weight per graph. Graph g has M_g candidates c_{gm}, with $M_g = 20$. A candidate is accepted, $I_{gm} = 1$, if it is finite, all its coordinates exceed 10^{-12}, and $R(c_{gm}) \le 10^{-3}$. Here R and E_{res} are defined as in the training objective, but are recomputed in float64 from the original α, β, κ rather than from the clipped internal features of the cell. We report

$$\overline{E}_{\mathrm{res}} = \frac{1}{n} \sum_g \frac{1}{M_g} \sum_m E_{\mathrm{res}}(c_{gm}), \qquad \mathrm{Hit}_{10^{-3}} = \frac{100}{n} \sum_g \frac{1}{M_g} \sum_m I_{gm}.$$

Nonfinite residuals are kept in these means, and acceptance involves no polishing or matching to reference roots. To count distinct predictions, we process the accepted candidates of each graph in

saved order. Each one joins the first existing representative within distance $\|\log c - \log c'\|_2/\sqrt{s_g} \leq 10^{-3}$, or otherwise becomes a new representative. With K_g the resulting number of representatives ($K_g = 0$ if no candidate is accepted), we report

$$D_{20}^{\text{raw}} = \frac{1}{n}\sum_g K_g, \qquad G_{20}^{\text{raw}} = \frac{100}{n}\sum_g \mathbf{1}\{K_g \geq 2\}. \tag{56}$$

For recurrent models, we measure numerical settling at horizon T by

$$C_T = \frac{100}{n}\sum_g \frac{1}{M_g}\sum_m \mathbf{1}\left\{\frac{\|z_{T+1}^{(g,m)} - z_T^{(g,m)}\|_2}{\sqrt{s_g}} < 10^{-3}\right\}, \tag{57}$$

and the column C in Table 3 reports C_{200}. This one-step test on the log-state is independent of the physical residual and does not certify asymptotic convergence. For IGNN, C is instead the percentage of graphs for which all three implicit layers satisfy $\|F_{\ell,g}(H_{\ell,200}) - H_{\ell,200}\|_{\infty,p} < 10^{-3}$ after 200 updates per layer. This measures hidden-state settling, not chemical validity. Feedforward and generative models have no autonomous state update, so C is not reported for them.

Longer test-time dynamics. To investigate the trajectories that fail the $T = 200$ one-step criterion, we extend the same three frozen Init-D models and initialization seeds to $t = 20{,}000$. No model weights, initialization distributions, or physical acceptance thresholds change. The main table's quality and diversity remain evaluated at $T = 200$, not at a retrospectively chosen longer horizon.

Table 13: Chemistry test-time dynamics, mean ± population SD across three training seeds. C_t is the fraction with next-step species log-state RMS change below 10^{-3}. Oscillatory denotes detected approximate periods 2–64 among trajectories failing that criterion; unresolved is the remainder. All percentages use the full trajectory population, and the three categories partition that population for $t \geq 200$, up to rounding. Dashes denote unperformed early-horizon oscillation tests.

t	C_t (%)	Oscillatory (%)	Unresolved (%)
10	2.26 ± 0.65	—	—
20	13.77 ± 3.72	—	—
50	63.11 ± 2.05	—	—
100	83.99 ± 0.79	—	—
200	91.99 ± 0.35	1.75 ± 1.50	6.26 ± 1.27
500	95.20 ± 1.72	2.23 ± 1.70	2.56 ± 0.54
1000	95.64 ± 1.67	2.49 ± 2.04	1.88 ± 0.46
2000	95.95 ± 1.83	2.77 ± 2.09	1.28 ± 0.26
5000	96.15 ± 2.01	2.77 ± 2.09	1.08 ± 0.27
10000	96.13 ± 2.01	2.77 ± 2.09	1.10 ± 0.27
20000	96.12 ± 2.01	2.77 ± 2.09	1.11 ± 0.31

Oscillation test: For $t \geq 200$, use the 257 saved log states $w_j = z_{t+j}$, $j = 0, \ldots, 256$. Let $A = \max_j \|w_j - \bar{w}\|_2/\sqrt{s}$. Among trajectories failing the one-step settling criterion, select the smallest $p \in \{2, \ldots, 64\}$ satisfying

$$\max_{0 \leq j \leq 256-p} \frac{\|w_{j+p} - w_j\|_2}{\sqrt{s}} \leq \min(10^{-3}, 0.02A).$$

Additionally require $A > 0$, half-window amplitude ratio in $[0.8, 1.25]$, and RMS distance at most 10^{-3} between the first/last cycle centers. Half-window amplitudes are maximum RMS deviations from their respective means over $w_0, \ldots, w_{127}$ and $w_{128}, \ldots, w_{256}$; cycle centers average the first and last p states. The window contains at least four cycles for every searched period.

Findings: Settling increases from $91.99 \pm 0.35\%$ at $t = 200$ to $96.15 \pm 2.01\%$ at $t = 5{,}000$ and then plateaus. At $t = 20{,}000$, $96.12 \pm 2.01\%$ are step-settled, $2.77 \pm 2.09\%$ are oscillatory, and $1.11 \pm 0.31\%$ are unresolved. The unresolved group contains 157 trajectories across the three seeds. The oscillatory group includes 40 small-amplitude trajectories excluded by the earlier amplitude-gated diagnostic. No nonfinite states occurred. Settling need not increase monotonically because each horizon is tested independently. The log-state box precludes finite-state norm divergence but not bounded nonconvergence. Unresolved does not establish slow convergence, chaos, or divergence: the category can include drift, irregular fluctuations, periods beyond 64, or transients. These

observations concern the learned solver, not physical chemical kinetics, and do not establish asymptotic attraction.

H Related Work

GNNs and their theory. Message-passing GNNs are at most as expressive as the 1-WL test (Xu et al., 2018; Morris et al., 2019), and a line of work characterizes which invariant and equivariant maps GNNs can approximate (Maron et al., 2019; Chen et al., 2019; Keriven & Peyré, 2019; Azizian & marc lelarge, 2021; Geerts & Reutter, 2022). Unique node identifiers or i.i.d. random node features restore universality with high probability (Loukas, 2019; Sato et al., 2021; Abboud et al., 2020). Recent work extends this to measurable targets with explicit approximation rates (Gonon et al., 2026), and to recurrent GNNs, which with random initialization can uniformly express any polynomial-time computable function on connected graphs (Rosenbluth & Grohe, 2026). All of these results concern *single-valued* targets $G \mapsto f(G)$, with randomness serving only to tell nodes apart. In our setting, the same random initial state must do double duty: it identifies nodes and selects the solution branch. Only the output-dimensional state y_t is carried between steps, so node distinguishability must be preserved by dynamics that also converge to a solution (Lemma 6, Theorem 2).

Recurrent/equilibrium GNNs. The original GNN was defined as the fixed point of a contraction (Scarselli et al., 2008). Later weight-tied models unroll a shared update (Li et al., 2015; Dai et al., 2018), and implicit GNNs guarantee a unique equilibrium through contraction, well-posedness, or monotonicity conditions (Gu et al., 2020; Liu et al., 2021; Park et al., 2021; Baker et al., 2023; Yang et al., 2025), as do deep equilibrium models more broadly (Bai et al., 2019; Winston & Kolter, 2020); path independence has even been advocated as a desirable property (Anil et al., 2022). Weight-tied message passing is also used for algorithmic reasoning and size extrapolation (Veličković et al., 2019; Tang et al., 2020; Bansal et al., 2022) and for SAT and constraint satisfaction (Selsam et al., 2018; Toenshoff et al., 2021). RUN-CSP in particular starts from random states and keeps the best of several parallel runs (Toenshoff et al., 2021), but without a representational account of the solutions reached. On the theory side, Liu et al. (2026) show that regular implicit operators (contractive in the state, Lipschitz in the input) represent exactly the locally Lipschitz maps, so that iterations rather than parameters supply expressivity. Jore & Liu (2026) drop contraction to represent set-valued maps through attractor landscapes, without considering permutation symmetry. We bring this line to graphs, where symmetry changes the picture in three ways: (i) the solution set is equivariant although individual branches need not be (Section 2.1); (ii) without auxiliary inputs, message passing can realize the dynamics only through symmetry breaking carried by the state itself (Theorem 2, Lemma 6); and (iii) contraction, the standard route to well-posedness above, misses every valid solution with positive probability (Theorems 3 and 6). As in Liu et al. (2026), complexity comes from iteration: each step of the ideal update is globally Lipschitz, while its limit selector jumps across basin boundaries (Theorem 1). Multiple attractors have long served as a resource in Hopfield networks (Hopfield, 1982; Ramsauer et al., 2020), where they store fixed patterns rather than solutions of an input-dependent problem.

Distributional models. Another way to handle multiple solutions is to learn a distribution over them. Unsupervised neural combinatorial optimization trains GNNs from an energy, either as mean-field relaxations (Karalias & Loukas, 2020; Schuetz et al., 2022) or as annealed, autoregressive, diffusion, or GFlowNet samplers (Sun et al., 2022; Sanokowski et al., 2023; Zhang et al., 2023); variational autoregressive networks play the same role for statistical-mechanics models (Wu et al., 2019). Supervised diffusion solvers and diffusion samplers generate candidates through time-inhomogeneous reverse processes (Sun & Yang, 2023; Ho et al., 2020; Song et al., 2021; Berner et al., 2024). Other work keeps supervision but relaxes the single target: multiple output heads with a hindsight loss (Li et al., 2018), learned selection among one-of-many solutions (Nandwani et al., 2020), label realignment under formulation symmetry (Chen et al., 2024), and equivariant flow matching with symmetry-aware target matching for bifurcation problems (Hendriks et al., 2025). Our approach differs in what is learned: a single autonomous update whose *fixed points* are the solutions. Each candidate is then a stationary point whose validity can be checked by its residual, the number of iterations can be extended at test time without retraining, and training requires only an energy or residual rather than solution labels. Classical multi-solution methods such as deflation and

homotopy continuation (Farrell et al., 2015; Breiding & Timme, 2018) operate instance by instance, whereas our model is trained once and amortized over a graph family.

Symmetry breaking. Deterministic equivariant networks cannot map a symmetric input to a less symmetric output (Smidt et al., 2021; Kaba & Ravanbakhsh, 2023). Existing remedies break symmetry through *augmented inputs*: random or positional node features (Sato et al., 2021; Abboud et al., 2020; Dwivedi et al., 2022), symmetry-breaking sets and objects (Smidt et al., 2021; Xie & Smidt, 2024; Goel et al., 2026), and probabilistic symmetry breaking through equivariant conditional distributions and sampled canonicalizations (Lawrence et al., 2025), the notion closest to our set-valued equivariance. We instead break symmetry through the model's own evolving state, with no auxiliary input: randomness enters once through y_0, and Lemma 6 shows that the dynamics preserve it. The two mechanisms are compatible. Augmented features can be appended to X or used to seed y_0 (our PROTEINS model uses random-walk encodings, and our chemistry model seeds y_0 from random node vectors), and the construction in Lemma 5 only needs distinct node codes, whatever their source. Augmentation alone, however, does not settle *which* solution the network should output: deterministic encodings still define a single-valued map that must break ties by convention on symmetric graphs, and continuous canonicalization is impossible in general (Dym et al., 2024). Theorems 3 and 6 concern updates without auxiliary inputs; combining multi-attractor dynamics with augmented features is a natural direction for future work.