STORM: SIMPLICIAL TOPOLOGICAL RECURRENT MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Many dynamical systems, from traffic networks to water infrastructure, involve signals and interactions across nodes, edges, and higher-order groups. Existing temporal models often ignore this structure or use it only as preprocessing, while graph-based approaches remain limited to pairwise relations. We introduce Simplicial Topological Recurrent Models (STORM), recurrent architectures on simplicial complexes that embed simplicial operators into memory gates, allowing topology to shape hidden-state dynamics over time. We prove permutation equivariance, finite-horizon stability to bounded admissible topology perturbations, and a Rademacher-complexity generalization bound with explicit temporal dependence. On traffic and hydraulic datasets, STORM improves over the best baselines in our evaluation by up to 50% at long horizons. For node-only forecasting, STORM and LightSTORM are highly competitive with strong graph models, with LightSTORM using up to roughly two orders of magnitude fewer parameters.

1 INTRODUCTION

Learning dynamical systems over complex relational domains is a central challenge across science and engineering Brunton & Kutz (2022), with applications in traffic, power grids, water distribution, and neural systems Battiston et al. (2021). In these settings, topology shapes signal evolution over time Majhi et al. (2022): traffic congestion propagates along roads, power flows follow conservation laws over transmission lines, and neural activity synchronizes across cortical regions.

One approach to learn from relational time series is to treat all variables as a multivariate sequence and model temporal dependencies with recurrent Hochreiter & Schmidhuber (1997); Beck et al. (2024) or attention-based architectures Vaswani et al. (2017); Liu et al. (2024). While these models can learn cross-variable dependencies, they do not explicitly exploit the underlying structure, potentially increasing sample and computational demands as the system grows Jin et al. (2024). On the other hand, temporal graph neural networks represent this structure as a graph and incorporate graph convolutions or diffusion operators into recurrent or attentional architectures Jin et al. (2024); Sabbaqi & Isufi (2023), imposing a graph-based inductive bias. However, standard graph-based temporal models are typically limited to pairwise relations and node-valued signals. Nevertheless, many physical and engineered systems, including brain, hydraulic, and traffic networks, involve quantities and interactions across nodes, edges, and higher-order groups, whose structure can shape the resulting dynamics Battiston et al. (2021); Majhi et al. (2022); Coletta et al. (2016).

A natural way to overcome these limitations is to model the domain with higher-order combinatorial structures such as simplicial complexes, cell complexes or hypergraphs, and leverage their associated inductive biases Papamarkou et al. (2024). We focus on simplicial complexes, whose closure property and boundary operators provide a consistent hierarchy across simplex orders, while Hodge Laplacians define order-specific convolutional operators Isufi et al. (2025). This structure enables signals to be represented and processed jointly across multiple simplex orders. We consider forecasting over a fixed simplicial complex with signals available at multiple simplex orders.

Related works. Recently, a growing body of work has begun to address the problem of learning from time-evolving signals on simplicial complexes. Topological summaries built via simplicial persistent homology have been used to condition a GRU for forecasting Chen et al. (2022b). A related approach enriches node features with edge and triangle information via Hodge Laplacians before a node-level recurrence Chen et al. (2023). Simplicial convolutions have been applied to signals as

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.

preprocessing before a recurrent encoder Mitchell et al. (2024), separating spatial aggregation from temporal modeling. Hodge-Laplacian filtering has been shown to couple simplex orders effectively for learning PDE dynamics Choi et al. (2024). Finally, simplicial convolutional filters have been embedded into a vector autoregressive framework Krishnan et al. (2024), extending classical linear time-series models to simplicial complexes. For a detailed discussion, see Appendix B.

Motivation. Existing methods for topological time series either summarize topology before recurrence Chen et al. (2022b), apply simplicial networks as preprocessing Mitchell et al. (2024), or rely on linear autoregressive dynamics Krishnan et al. (2024). What is missing is a nonlinear gated recurrent model whose memory update is directly shaped by simplicial operators across simplex orders. By embedding simplicial filters inside the gates, STORM lets topology control what information is written, retained, and exposed, rather than only filtering the input to a generic recurrent cell.

Contributions. STORM is the first nonlinear gated recurrent architecture in which simplicial convolutional operators parameterize the recurrent gates, enabling topology-aware memory updates for time series on nodes, edges, and faces.

- **C1. Architecture.** We introduce STORM, a family of gated recurrent architectures for multi-order signals on simplicial complexes. STORM parameterizes the recurrent gates with simplicial convolutional filters, coupling within-order Hodge diffusion and cross-order boundary interactions at every timestep. We also introduce LightSTORM, a Dirac-operator variant that shares parameters across simplex orders. Both use a continuous-time forget gate supporting per-simplex memory decay. Details are in Section 3.
- **C2. Theory.** We study three properties of STORM. These results show that the proposed topology-aware recurrence preserves the symmetries of the domain through permutation equivariance, degrades gracefully under bounded topology perturbations through stability, and admits a finite-horizon generalization bound whose dependence on the sequence length is characterized directly from the recurrent dynamics.
- **C3. Experiments.** We evaluate STORM on three real-world datasets against sequence, graph-based, and simplicial baselines. In multi-order forecasting, STORM achieves the strongest medium- and long-horizon performance on the traffic datasets, with final-step NMSE gains of $\approx$50% on PeMS-D8 and $\approx$40% on PeMS-D4 at S=80. In node-only forecasting, STORM and LightSTORM remain highly competitive with strong spatio-temporal graph models, while LightSTORM uses substantially fewer parameters. Extensive ablations isolate the effect of the main architectural components and show that topology-aware recurrent dynamics are particularly beneficial at long horizons. Controlled topology perturbation experiments further support the stability behavior predicted by our analysis.

2 BACKGROUND

Let $\mathcal{V}$ be a finite vertex set. A k-simplex is a subset of $\mathcal{V}$ with cardinality $k+1$. We denote the set of k-simplices in the complex by $\mathcal{S}_k$, with $n_k = |\mathcal{S}_k|$. A $(k-1)$-face of a k-simplex is a subset of it with cardinality k; a $(k+1)$-coface is a superset with cardinality $k+2$ Barbarossa & Sardellitti (2020). Two simplices sharing a common face are *lower adjacent*; those sharing a common coface are *upper adjacent* (Figure 1). A simplicial complex (SC) $\mathcal{K}$ of order K is a collection of simplices up to order K closed under inclusion. We consider complexes of order two: nodes ($\mathcal{S}_0$, n_0), edges ($\mathcal{S}_1$, n_1), and filled triangles ($\mathcal{S}_2$, n_2), as shown in Figure 1. After fixing an orientation for every simplex, the relationships between adjacent simplex orders are encoded by the oriented boundary incidence matrices $\mathbf{B}_k \in \mathbb{R}^{n_{k-1} \times n_k}$, with entries in $\{-1, 0, +1\}$, satisfying $\mathbf{B}_{k-1}\mathbf{B}_k = \mathbf{0}$. Concretely, $\mathbf{B}_1 \in \mathbb{R}^{n_0 \times n_1}$ maps each edge to its two endpoint nodes, while $\mathbf{B}_2 \in \mathbb{R}^{n_1 \times n_2}$ maps each filled triangle to its three boundary edges, with signs encoding orientation. From these matrices, we construct the Hodge Laplacians:

$$\mathbf{L}_k^{\downarrow} = \mathbf{B}_k^{\top}\mathbf{B}_k, \qquad \mathbf{L}_k^{\uparrow} = \mathbf{B}_{k+1}\mathbf{B}_{k+1}^{\top}, \qquad \mathbf{L}_k = \mathbf{L}_k^{\downarrow} + \mathbf{L}_k^{\uparrow}, \tag{1}$$

where nonexistent boundary terms are omitted at $k = 0$ and $k = K$. Here, $\mathbf{L}_k^{\downarrow}$ encodes lower adjacencies, whereas $\mathbf{L}_k^{\uparrow}$ encodes upper adjacencies. At $k = 0$, $\mathbf{L}_0 = \mathbf{L}_0^{\uparrow}$ recovers the standard graph Laplacian because nodes have no lower-adjacency term. At $k = K$, $\mathbf{L}_K = \mathbf{L}_K^{\downarrow}$ because the highest-order simplices have no upper neighbors. At each time t, we observe signals $\mathbf{X}_t^{(k)} \in \mathbb{R}^{n_k \times d_k}$ on

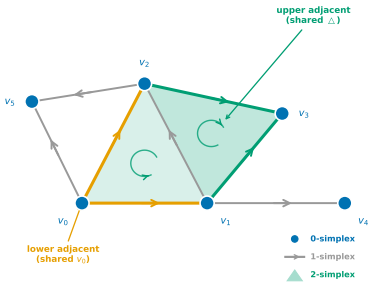


Figure 1: A SC of order two. Edges carry fixed orientations (arrows) and triangles a fixed winding (circular arrows). Lower adjacency (orange): shared node. Upper adjacency (green): shared triangle.

each order; the topology $\mathcal{K}$ is fixed. A simplicial convolutional filter (SCF) at order k is a polynomial in the two Laplacians with separate learnable matrix coefficients:

$$g(\mathbf{L}_k^{\downarrow}, \mathbf{L}_k^{\uparrow})\, \mathbf{X}_t^{(k)} = \mathbf{X}_t^{(k)} \mathbf{\Theta}_{\mathrm{id}} + \sum_{p=1}^{P} (\mathbf{L}_k^{\downarrow})^p \mathbf{X}_t^{(k)} \mathbf{\Theta}_p^{\downarrow} + \sum_{p=1}^{P} (\mathbf{L}_k^{\uparrow})^p \mathbf{X}_t^{(k)} \mathbf{\Theta}_p^{\uparrow} =: \mathbf{Y}_t^{(k)}. \tag{2}$$

The separate matrix coefficients allow the filter to learn different multi-channel transformations for the self, lower-adjacency, and upper-adjacency components, rather than tying them through a single polynomial in $\mathbf{L}_k$ Yang et al. (2022b); Yang & Isufi (2023). Signals can also be transferred between adjacent orders via the boundary operators: a signal from order $k{-}1$ is lifted to order k as $\widetilde{\mathbf{S}}_{\downarrow}^{(k)} = \mathbf{B}_k^{\top} \mathbf{X}_t^{(k-1)}$ and from order $k{+}1$ projected to order k as $\widetilde{\mathbf{S}}_{\uparrow}^{(k)} = \mathbf{B}_{k+1} \mathbf{X}_t^{(k+1)}$, yielding cross-order inputs processed alongside $\mathbf{X}_t^{(k)}$ in Section 3. For further background, see Appendix A.

3 STORM: Simplicial Topological Recurrent Models

The Simplicial Topological Recurrent Model (STORM) is a recurrent architecture for time series over a fixed simplicial complex $\mathcal{K}$. It jointly learns spatial and temporal dependencies across all simplex orders by embedding simplicial convolutional filters inside the recurrent gates. For each order k and timestep t, the model receives $\mathbf{X}_t^{(k)} \in \mathbb{R}^{n_k \times d_k}$, maintains hidden and cell states $\mathbf{h}_t^{(k)}, \mathbf{c}_t^{(k)} \in \mathbb{R}^{n_k \times H}$, and produces multi-step predictions through a linear decoder. For each gate g, the pre-activation $\mathbf{G}_{g,t}^{(k)} \in \mathbb{R}^{n_k \times H}$ aggregates intra- and cross-order topological information and drives the recurrent update (Figure 2).

STORM The model uses an LSTM-style gated recurrent cell with an exponential forget gate inspired by xLSTM Beck et al. (2024). At each timestep t and order k, an input gate $\mathbf{i}_t^{(k)}$ controls how much new information is written to the cell state $\mathbf{c}_t^{(k)}$, a forget gate $\mathbf{f}_t^{(k)}$ controls how much of the previous cell state is retained, and an output gate $\mathbf{o}_t^{(k)}$ controls what is read out as the hidden state $\mathbf{h}_t^{(k)}$. The input, candidate, output, and timescale pre-activations, indexed by $g \in \{i, z, o, \tau\}$, are computed from both the current input and the previous hidden state by topology-aware SCFs, so that the topology of $\mathcal{K}$ directly shapes what is remembered, forgotten, and output.

The gates and cell state are updated as:

$$\mathbf{i}_t^{(k)} = \sigma(\mathbf{G}_i^{(k)}), \qquad \widetilde{\mathbf{c}}_t^{(k)} = \tanh(\mathbf{G}_z^{(k)}), \qquad \mathbf{o}_t^{(k)} = \sigma(\mathbf{G}_o^{(k)}), \tag{3}$$

$$\mathbf{c}_t^{(k)} = \mathbf{f}_t^{(k)} \odot \mathbf{c}_{t-1}^{(k)} + \mathbf{i}_t^{(k)} \odot \widetilde{\mathbf{c}}_t^{(k)}, \qquad \mathbf{h}_t^{(k)} = \mathbf{o}_t^{(k)} \odot \tanh(\mathbf{c}_t^{(k)}), \tag{4}$$

with σ the sigmoid function, $\odot$ the elementwise multiplication, and all gates and states in $\mathbb{R}^{n_k \times H}$. Since signals may evolve at different rates across orders and across simplices within the same order,

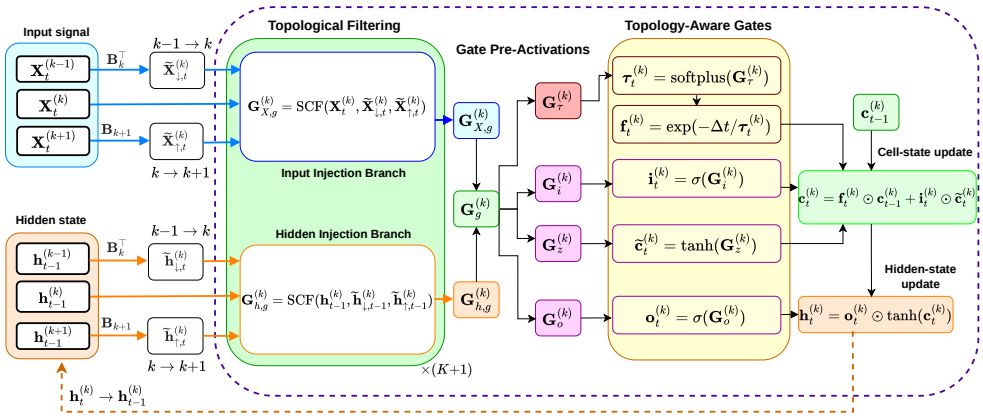


Figure 2: STORM architecture for order k at timestep t. Parallel SCF branches process the input and previous hidden state, with cross-order signals from $k\pm1$ injected through boundary operators. The same procedure is applied across all simplex orders.

we replace the standard forget gate with a parameterization that gives each simplex and hidden channel its own memory timescale. We define the forget gate as

$$[\mathbf{f}_t^{(k)}]_{ij} = \exp\Big(-\Delta t/[\boldsymbol{\tau}_t^{(k)}]_{ij}\Big), \qquad \boldsymbol{\tau}_t^{(k)} = \mathrm{clamp}\Big(\mathrm{softplus}(\mathbf{G}_{\tau,t}^{(k)}), \tau_{\min}, \tau_{\max}\Big), \tag{5}$$

where $\boldsymbol{\tau}_t^{(k)}$ is the learnable timescale, $\mathrm{softplus}(x) = \log(1 + e^x)$ ensures positivity, and $\Delta t > 0$ is the time interval between consecutive observations. This parameterization is motivated by exponential memory decay: a simplex with a small timescale forgets quickly, while one with a large timescale retains information over longer horizons. The gate pre-activations $\mathbf{G}_g^{(k)}$ are computed by an SCF on the fixed simplicial complex $\mathcal{K}$ Yang et al. (2022a); Yang & Isufi (2023), augmented with boundary-based cross-order injections. This introduces the topological inductive bias inside the recurrent gates: the Hodge Laplacians encode lower and upper adjacencies within each order, while the boundary operators inject information from neighboring orders. The filter weight matrices are independent of n_k, so the parameter count does not grow with the complex size. No pointwise nonlinearity is applied in the filter, as they are handled by the recurrent gates. Using the cross-order projections from Section 2, at time t, we obtain the input-branch contribution $\mathbf{G}_{X,g}^{(k)}$ as:

$$\mathbf{X}_t^{(k)}\boldsymbol{\Theta}_{g,\mathrm{id}}^{(k)} + \underbrace{\sum_{p=1}^{P}(\mathbf{L}_k^{\downarrow})^p\mathbf{X}_t^{(k)}\boldsymbol{\Theta}_{g,\downarrow,p}^{(k)} + \sum_{p=1}^{P}(\mathbf{L}_k^{\uparrow})^p\mathbf{X}_t^{(k)}\boldsymbol{\Theta}_{g,\uparrow,p}^{(k)}}_{\text{intra-order Hodge filtering}} + \underbrace{\sum_{p=0}^{P}(\mathbf{L}_k^{\downarrow})^p\widetilde{\mathbf{X}}_{t,\downarrow}^{(k)}\boldsymbol{\Theta}_{g,\mathrm{inj},\downarrow,p}^{(k)} + \sum_{p=0}^{P}(\mathbf{L}_k^{\uparrow})^p\widetilde{\mathbf{X}}_{t,\uparrow}^{(k)}\boldsymbol{\Theta}_{g,\mathrm{inj},\uparrow,p}^{(k)}}_{\text{cross-order injection}} \tag{6}$$

where P is the polynomial degree, $\widetilde{\mathbf{X}}_{t,\downarrow}^{(k)}$ and $\widetilde{\mathbf{X}}_{t,\uparrow}^{(k)}$ are the projected signals from neighboring orders, and the matrices $\boldsymbol{\Theta}^{(k)}$ are learnable coefficients specific to gate g, branch, and order k. The hidden-branch pre-activation $\mathbf{G}_{h,g}^{(k)}$ is obtained analogously by replacing $\mathbf{X}_t^k$ with $\mathbf{h}_{t-1}^k$ in equation 6 and using hidden-branch coefficients. The full pre-activation for gate g at order k and time t is:

$$\mathbf{G}_g^{(k)} = \mathbf{G}_{X,g}^{(k)} + \mathbf{G}_{h,g}^{(k)} + \mathbf{1}_{n_k}(\mathbf{b}_g^{(k)})^\top, \tag{7}$$

where $\mathbf{b}_g^{(k)} \in \mathbb{R}^H$ is a learnable bias. For the hidden branch, all coefficient matrices have size $H \times H$. For the input branch, the coefficient matrices map from the feature dimension of the corresponding source order to H: intra-order terms use $d_k \times H$, lower-injection terms use $d_{k-1} \times H$, and upper-injection terms use $d_{k+1} \times H$, with nonexistent neighboring orders omitted. As a design choice, STORM supports three configurations for cross-order coupling: *input*, where the injection terms in equation 6 are active only in the $\mathbf{X}_t$ branch; *hidden*, where they are active only in the $\mathbf{h}_{t-1}$ branch; and *both*, where they are active in both branches. In all configurations, both branches contribute to equation 7. This flexibility allows us to study the relative contribution of visible and latent cross-order interactions independently.

LightSTORM The SCF in equation 6 uses separate weight matrices for each order k, so the parameter count grows linearly with $K+1$. LightSTORM reduces this cost by replacing the per-order SCFs with a shared polynomial filter in the Dirac operator $\mathbf{D}$, which couples signals across simplex orders through boundary-based lower and upper components Battiloro et al. (2024b). At time t, after mapping each order to the hidden dimension when needed, we stack the order-wise input signals along the simplex dimension into $\mathbf{X} \in \mathbb{R}^{N\times H}$, where $N = \sum_k n_k$. The hidden states are stacked also into $\mathbf{h} \in \mathbb{R}^{N\times H}$. The LightSTORM input-branch contribution for gate g is

$$\mathbf{G}_{X,g} = \mathbf{X}\mathbf{\Theta}_{g,0} + \sum_{p=1}^{J} (\mathbf{D}^{(d)})^p \mathbf{X}\mathbf{\Theta}_{g,p}^{(d)} + \sum_{p=1}^{J} (\mathbf{D}^{(u)})^p \mathbf{X}\mathbf{\Theta}_{g,p}^{(u)}, \tag{8}$$

where $\mathbf{D}^{(d)}$ and $\mathbf{D}^{(u)}$ are the downward and upward Dirac components. The hidden-branch contribution $\mathbf{G}_{h,g}$ is obtained analogously by replacing $\mathbf{X}$ with $\mathbf{h}$ and using hidden-branch coefficients. For each branch, the coefficients $\mathbf{\Theta}_{g,0}$, $\mathbf{\Theta}_{g,p}^{(d)}$, and $\mathbf{\Theta}_{g,p}^{(u)} \in \mathbb{R}^{H\times H}$ are shared across simplex orders. The identity term uses $\mathbf{\Theta}_{g,0}$, while separate coefficients are used for the nonzero powers of the downward and upward Dirac components. The output is then partitioned into order-k blocks and used in equation 7. For the hidden-state filters, parameter sharing reduces the trainable filter count from $\mathcal{O}((K+1)G(4P+3)H^2)$ to $\mathcal{O}(G(2J+1)H^2)$. Further details are given in Appendix C.

A detailed complexity analysis of STORM and LightSTORM is provided in Appendix D.

4 Theoretical Analysis

We provide a theoretical characterization of STORM, focusing on permutation equivariance, stability to bounded topology perturbations, and finite-horizon Rademacher-complexity generalization bound with explicit dependence on the sequence length. Proofs are provided in Appendix E.

Equivariance. Since STORM learns from the topology of $\mathcal{K}$, its predictions should depend on the connectivity structure and not on the arbitrary way simplices are labeled. We formalize this property through permutation equivariance.

Proposition 1 (Permutation equivariance)**.** *For each order k, let $\mathbf{P}_k \in \{0,1\}^{n_k\times n_k}$ be a permutation matrix. Define the permuted complex by $\mathbf{B}'_k = \mathbf{P}_{k-1}\mathbf{B}_k\mathbf{P}_k^\top$, permuted inputs by $\mathbf{X}_t^{(k)\prime} = \mathbf{P}_k\mathbf{X}_t^{(k)}$, and permuted initial states analogously. Then for all $t \geq 0$ and $k \in \{0,\dots,K\}$:*

$$\mathbf{h}_t^{(k)\prime} = \mathbf{P}_k\,\mathbf{h}_t^{(k)}, \qquad \mathbf{c}_t^{(k)\prime} = \mathbf{P}_k\,\mathbf{c}_t^{(k)}.$$

Proposition 1 shows that STORM is equivariant to simplex relabeling, and therefore does not depend on the chosen labeling of the complex. Its orientation sensitivity is discussed in Appendix E.2.

Stability Under Topology Perturbations. A model is stable if small changes in its input domain produce proportionally small changes in its output. Since the topology $\mathcal{K}$ may be noisy or estimated from data, the corresponding Hodge Laplacians and boundary operators may be perturbed. We show that STORM remains stable under such admissible perturbations.

Given a fixed oriented simplicial complex $\mathcal{K}$, consider a small admissible perturbation of its weighted operators. The perturbed operators preserve the simplex sets, orientations, and boundary sparsity pattern, while the nonzero incidence weights are slightly perturbed:

$$\widehat{\mathbf{B}}_k = \mathbf{B}_k + \Delta\mathbf{B}_k, \qquad \operatorname{supp}(\Delta\mathbf{B}_k) \subseteq \operatorname{supp}(\mathbf{B}_k), \qquad \|\Delta\mathbf{B}_k\|_{\mathrm{op}} \leq \eta r_k,$$

and the induced Hodge Laplacians satisfy relative perturbation bounds of order η.

We use the following assumptions, stated in Appendix E.3: **E.1** bounds the perturbation size η; **E.2** bounds the boundary-operator norms; **E.3** requires finite-degree SCFs with finite coefficients; **E.4** requires Lipschitz nonlinearities; and **E.5** uniformly bounds the inputs, time constants, and time steps, with $0 < \tau_{\min} \leq \boldsymbol{\tau}_t^{(k)} \leq \tau_{\max} < \infty$ and $0 < \Delta_{\min} \leq \Delta t \leq \Delta_{\max} < \infty$. Hence the forget gate satisfies $\mathbf{f}_t^{(k)} \leq \gamma := \exp(-\Delta_{\min}/\tau_{\max}) < 1$ entrywise.

For $\lambda > 0$, let d_w be the weighted state metric

$$d_w(S,S') := \max_k \left\|\mathbf{h}^{(k)} - \mathbf{h}^{(k)\prime}\right\|_F + \lambda \max_k \left\|\mathbf{c}^{(k)} - \mathbf{c}^{(k)\prime}\right\|_F.$$

We write $\|\cdot\|_{\mathcal{P}}$ for the stacked output norm after a consistent relabeling of simplices across orders.

Theorem 1 (General stability of STORM)**.** *Under Assumptions **E.1–E.5**, suppose that the one-step STORM state map is q-Lipschitz in the state metric d_w, uniformly along the two trajectories. Then, with identical parameters run on $\mathcal{K}$ and $\widehat{\mathcal{K}}$, for every $t \geq 1$,*

$$\left\|\mathbf{Y}_t - \widehat{\mathbf{Y}}_t\right\|_{\mathcal{P}} \leq C_{\mathrm{out}}(\kappa\eta + \kappa'\eta^2) \sum_{s=0}^{t-1} q^s,$$

where $C_{\mathrm{out}}, \kappa, \kappa' \geq 0$ are finite and depend only on the fixed model parameters, reference operators, perturbation radius, chosen state metric, state dimensions, and input and time bounds. In particular, if $q \leq 1$, then

$$\left\|\mathbf{Y}_t - \widehat{\mathbf{Y}}_t\right\|_{\mathcal{P}} \leq \mathsf{C}\, t\, \eta + \mathsf{C}'\, t\, \eta^2,$$

with $\mathsf{C}, \mathsf{C}' < \infty$ independent of t.

Theorem 1 shows that the output discrepancy between two runs of STORM on a fixed topology and a small admissible perturbation of its weighted operators is $\mathcal{O}((\eta + \eta^2)\sum_{s=0}^{t-1} q^s)$. When $q \leq 1$, the bound grows at most linearly with t; when $q < 1$, the geometric sum is bounded by $(1-q)^{-1}$ and the discrepancy is $\mathcal{O}(\eta)$ uniformly in time.

The full proof, a sufficient contraction condition with the resulting BIBO theorem, and a comparison with related graph stability results Gama et al. (2020); Ruiz et al. (2020) are given in Appendix E.3.

Generalization Bound. We bound the generalization gap for the hypothesis class $\mathcal{F}_T = \{f_\theta : \theta \in \Theta\}$ using Rademacher complexity Mohri et al. (2018); Bartlett & Mendelson (2002); Shalev-Shwartz & Ben-David (2014). We derive the temporal dependence from the sensitivity of the STORM recurrence to its state and shared parameters. Consider M training trajectories $\{(\mathbf{X}^{(m)}_{1:T}, \mathbf{Y}^{*(m)})\}_{m=1}^{M}$ drawn from $\mathcal{D}$. Let $R(f_\theta) := \mathbb{E}_{(\mathbf{X},\mathbf{Y}^*)\sim\mathcal{D}}[\ell(f_\theta(\mathbf{X}), \mathbf{Y}^*)]$ and $\widehat{R}_M(f_\theta) := \frac{1}{M}\sum_{m=1}^{M} \ell(f_\theta(\mathbf{X}^{(m)}, \mathbf{Y}^{*(m)}))$ denote the expected and empirical risks. We use the following assumptions, detailed in Appendix E.4: **E.6** assumes the training trajectories are drawn i.i.d. from $\mathcal{D}$; **E.7** bounds the inputs uniformly; **E.8** requires the loss to be L_ℓ-Lipschitz in the prediction and bounded in $[0, B_\ell]$; **E.9** assumes a nonempty parameter class $\Theta \subset \mathbb{R}^d$, fixed independently of the observed sample and with finite Euclidean diameter D_Θ; **E.10** assumes $\mathcal{S}_0 = \mathbf{0}$, $\tau_t^{(k)} \in [\tau_{\min}, \tau_{\max}]$ with $0 < \tau_{\min} \leq \tau_{\max}$, and $\Delta t \in [\Delta_{\min}, \Delta_{\max}]$ with $\Delta_{\min} > 0$; and **E.11** assumes the simplicial complex is fixed and finite.

Let q_λ bound how much a difference between two recurrent states can change after one STORM update, and let L_Φ bound how much one update can change when the shared recurrent parameters are perturbed. Define

$$\Psi_T(q_\lambda) := \sum_{j=0}^{T-1} q_\lambda^j, \qquad L_T := \sqrt{A_{\mathrm{out}}^2 + \left[C_{\mathrm{out}} L_\Phi \Psi_T(q_\lambda)\right]^2},$$

where A_{out} and C_{out} account for changes in the output caused by changes in the readout parameters and recurrent state, respectively.

Theorem 2 (Finite-horizon generalization bound)**.** *Under assumptions **E.6– E.11**, there exists a numerical constant $C > 0$ such that, for any $\delta \in (0,1)$, with probability at least $1-\delta$, uniformly over $\theta \in \Theta$,*

$$R(f_\theta) \leq \widehat{R}_M(f_\theta) + C L_\ell D_\Theta L_T \sqrt{\frac{d}{M}} + B_\ell \sqrt{\frac{\log(1/\delta)}{2M}}.$$

If there exists $\lambda > 0$ such that $q_\lambda < 1$, then $\Psi_T(q_\lambda) \leq (1-q_\lambda)^{-1}$, and the temporal factor in the bound is independent of the sequence length T.

The bound therefore depends on the number of training trajectories M, the size of the parameter class through $D_\Theta\sqrt{d}$, and the recurrent sensitivity through L_T. The topology affects L_T through the simplicial operators used in the recurrent filters. The factor $\Psi_T(q_\lambda)$ describes how the effect of a parameter perturbation can accumulate over T recurrent steps. If $q_\lambda < 1$, this temporal factor remains bounded independently of T. Proof provided in E.4.

5 NUMERICAL RESULTS

We evaluate STORM and LightSTORM on multivariate time-series forecasting over fixed simplicial complexes. Our main setting jointly forecasts signals across multiple simplex orders. We use two traffic networks with triangles, PeMS-D4 and PeMS-D8, and one hydraulic network with nodes and edges, Water U.S. Environmental Protection Agency (2000). For PeMS-D4 and PeMS-D8, we construct fixed clique complexes from the sensor graphs and lift the observed node-level traffic signals to edges and triangles; Water provides native node-pressure and edge-flow measurements. Our experiments address three questions: **RQ1** whether topology-aware recurrence improves multi-order forecasting, particularly at longer horizons; **RQ2** whether STORM and LightSTORM remain competitive when forecasting nodes only against strong spatio-temporal graph models; and **RQ3** which architectural components drive the gains and whether the empirical response to topology perturbations is consistent with the stability analysis. For multi-order forecasting, we compare against topology-unaware sequence models (LSTM Hochreiter & Schmidhuber (1997), xLSTM Beck et al. (2024), iTransformer Liu et al. (2024)), a graph-based recurrent model (GCN-xLSTM), and simplicial baselines (SC-VAR Krishnan et al. (2024) and SCRNN Mitchell et al. (2024)). For node forecasting, we compare against DCRNN Li et al. (2018), Graph WaveNet Wu et al. (2019), STID Shao et al. (2022), and STAEformer Liu et al. (2023). We evaluate $S \in \{5, 10, 20, 40, 60, 80\}$ horizons using final-step NMSE. Full experimental details and additional results are provided in Appendix F.

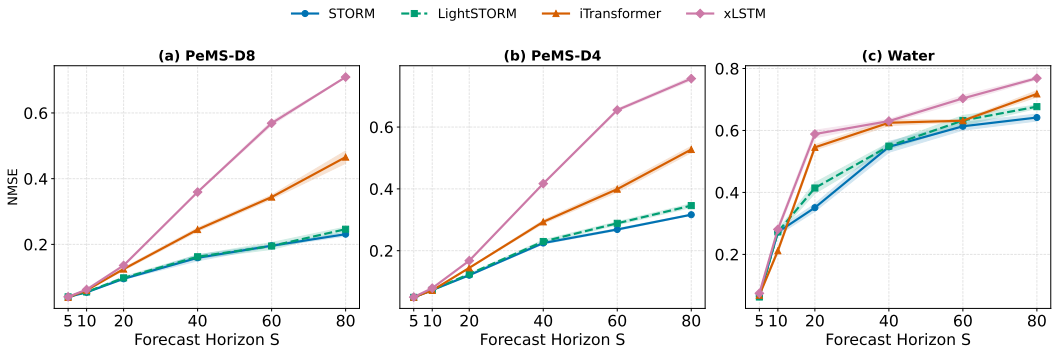


Figure 3: NMSE vs. forecasting step S (mean $\pm$ std, 5 seeds; lower is better). For STORM and LightSTORM, each point shows the best injection variant at the corresponding dataset and horizon.

Main results. Tables 1–3 and Figure 3 summarize the main comparison. On PeMS-D4, STORM and LightSTORM outperform all other baselines at every horizon except S=10, where they rank second. On PeMS-D8, STORM is the best-performing model from S=10 onward, with its gain over the strongest baseline increasing with the forecasting horizon and reaching 50.4% at S=80. The simplicial baselines SC-VAR and SCRNN remain less accurate, suggesting that the use of simplicial structure alone does not explain these gains and that incorporating topological operators directly into the recurrent memory updates plays an important role. The benefit of topology-aware recurrence is therefore strongly horizon-dependent: differences are modest at short horizons but become much larger as the forecasting horizon increases. On Water, the results are more mixed: iTransformer is stronger at short horizons, while STORM performs best from S=20 onward. LightSTORM retains much of STORM's performance on the traffic datasets with fewer parameters, while scaling iTransformer does not close the long-horizon gap (Appendix F.4).

Node forecasting. Although STORM is designed to process signals across multiple simplex orders, its formulation is flexible and can be used when predictions are required only at a subset of orders. We therefore also evaluate node-only forecasting to test whether processing higher-order information can benefit node predictions alone. In this setting, higher-order signals are processed internally, while supervision and evaluation are restricted to nodes. We evaluate this setting on PeMS-D4 and PeMS-D8. Table 4 shows that STORM and LightSTORM remain competitive in the node-only setting. LightSTORM performs particularly well on PeMS-D8, where it is best across

Table 1: NMSE (lower is better) at forecast step S on PeMS-D8 (mean $\pm$ std, 5 seeds). **Bold**: best overall. Underline: second best. Last row: blue/red = gain/loss of our best vs. best baseline (%).

Model	S=5	S=10	S=20	S=40	S=60	S=80
STORM (both)	0.0404 ± 0.0005	**0.0541** ± 0.0007	0.0963 ± 0.0004	0.1596 ± 0.0041	0.1959 ± 0.0033	**0.2310** ± 0.0059
STORM (hidden)	0.0402 ± 0.0005	0.0560 ± 0.0004	**0.0953** ± 0.0019	**0.1592** ± 0.0093	0.2008 ± 0.0065	0.2370 ± 0.0125
LightSTORM (both)	0.0403 ± 0.0003	0.0558 ± 0.0006	0.0982 ± 0.0037	0.1652 ± 0.0037	**0.1951** ± 0.0105	0.2462 ± 0.0067
LightSTORM (hidden)	0.0403 ± 0.0004	0.0552 ± 0.0008	0.0983 ± 0.0029	0.1624 ± 0.0055	0.2063 ± 0.0090	0.2511 ± 0.0177
iTransformer	**0.0388** ± 0.0001	0.0604 ± 0.0005	0.1248 ± 0.0038	0.2451 ± 0.0055	0.3439 ± 0.0062	0.4656 ± 0.0188
xLSTM	0.0403 ± 0.0003	0.0625 ± 0.0006	0.1359 ± 0.0022	0.3593 ± 0.0016	0.5687 ± 0.0065	0.7092 ± 0.0009
LSTM	0.0392 ± 0.0004	0.0622 ± 0.0009	0.1373 ± 0.0007	0.3616 ± 0.0085	0.5640 ± 0.0064	0.6958 ± 0.0066
GCN-xLSTM	0.0404 ± 0.0002	0.0622 ± 0.0006	0.1333 ± 0.0029	0.2909 ± 0.0064	0.4427 ± 0.0053	0.5623 ± 0.0092
SC-VAR	0.0508 ± 0.0002	0.0966 ± 0.0007	0.2407 ± 0.0003	0.5675 ± 0.0004	0.7904 ± 0.0001	0.8934 ± 0.0007
SCRNN	0.8706 ± 0.0004	0.8726 ± 0.0003	0.8753 ± 0.0006	0.8817 ± 0.0060	0.8820 ± 0.0033	0.8856 ± 0.0020
Gain vs. best baseline	−3.6%	+10.4%	+23.6%	+35.0%	+43.3%	+50.4%

Table 2: NMSE (lower is better) at forecast step S on PeMS-D4 (mean $\pm$ std, 5 seeds). **Bold**: best overall. Underline: second best. Last row: blue/red = gain/loss of our best vs. best baseline (%).

Model	S=5	S=10	S=20	S=40	S=60	S=80
STORM (both)	0.0495 ± 0.0002	0.0725 ± 0.0004	0.1214 ± 0.0013	**0.2247** ± 0.0033	**0.2686** ± 0.0031	**0.3164** ± 0.0017
STORM (hidden)	0.0494 ± 0.0003	0.0728 ± 0.0008	**0.1206** ± 0.0020	0.2259 ± 0.0006	0.2759 ± 0.0077	0.3287 ± 0.0179
LightSTORM (both)	0.0494 ± 0.0004	0.0735 ± 0.0005	0.1244 ± 0.0008	0.2297 ± 0.0051	0.2886 ± 0.0051	0.3458 ± 0.0061
LightSTORM (hidden)	**0.0491** ± 0.0002	0.0729 ± 0.0009	0.1253 ± 0.0023	0.2449 ± 0.0069	0.3093 ± 0.0133	0.3626 ± 0.0074
iTransformer	0.0494 ± 0.0001	**0.0715** ± 0.0005	0.1444 ± 0.0012	0.2933 ± 0.0054	0.3991 ± 0.0097	0.5274 ± 0.0100
xLSTM	0.0498 ± 0.0002	0.0788 ± 0.0005	0.1676 ± 0.0014	0.4170 ± 0.0015	0.6550 ± 0.0042	0.7567 ± 0.0060
LSTM	0.0499 ± 0.0002	0.0789 ± 0.0004	0.1690 ± 0.0005	0.4202 ± 0.0036	0.6500 ± 0.0052	0.7589 ± 0.0044
GCN-xLSTM	0.0495 ± 0.0004	0.0779 ± 0.0004	0.1573 ± 0.0012	0.3641 ± 0.0022	0.5420 ± 0.0043	0.6453 ± 0.0077
SC-VAR	0.0623 ± 0.0004	0.1173 ± 0.0009	0.2717 ± 0.0005	0.6046 ± 0.0004	0.8303 ± 0.0002	0.9339 ± 0.0004
SCRNN	0.8036 ± 0.0002	0.8057 ± 0.0012	0.8137 ± 0.0010	0.8187 ± 0.0023	0.8151 ± 0.0013	0.8257 ± 0.0011
Gain vs. best baseline	+0.6%	−1.4%	+16.5%	+23.4%	+32.7%	+40.0%

Table 3: NMSE (lower is better) at forecast step S on Water (mean $\pm$ std, 5 seeds). **Bold**: best overall. Underline: second best. Last row: blue/red = gain/loss of our best vs. best baseline (%).

Model	S=5	S=10	S=20	S=40	S=60	S=80
STORM (both)	0.0679 ± 0.0010	0.2772 ± 0.0035	0.3519 ± 0.0144	0.5517 ± 0.0143	**0.6135** ± 0.0125	**0.6417** ± 0.0113
STORM (hidden)	**0.0655** ± 0.0027	0.2720 ± 0.0098	**0.3509** ± 0.0122	**0.5471** ± 0.0176	0.6222 ± 0.0123	0.6548 ± 0.0105
LightSTORM (both)	0.0697 ± 0.0010	0.3016 ± 0.0061	0.4143 ± 0.0175	0.5495 ± 0.0103	0.6379 ± 0.0097	0.6884 ± 0.0144
LightSTORM (hidden)	0.0696 ± 0.0020	0.3054 ± 0.0053	0.4144 ± 0.0227	0.5604 ± 0.0077	0.6321 ± 0.0163	0.6768 ± 0.0055
iTransformer	0.0683 ± 0.0021	**0.2118** ± 0.0075	0.5455 ± 0.0085	0.6249 ± 0.0095	0.6315 ± 0.0033	0.7180 ± 0.0104
xLSTM	0.0745 ± 0.0018	0.2809 ± 0.0088	0.5884 ± 0.0123	0.6303 ± 0.0060	0.7035 ± 0.0098	0.7687 ± 0.0046
LSTM	0.0658 ± 0.0039	0.2380 ± 0.0083	0.5576 ± 0.0081	0.5984 ± 0.0061	0.6469 ± 0.0065	0.7090 ± 0.0088
GCN-xLSTM	0.0739 ± 0.0041	0.3252 ± 0.0207	0.5305 ± 0.0158	0.5990 ± 0.0101	0.6512 ± 0.0109	0.7182 ± 0.0070
SC-VAR	0.2353 ± 0.0109	0.5703 ± 0.0077	0.8588 ± 0.0056	0.8928 ± 0.0035	0.7244 ± 0.0004	0.9276 ± 0.0088
SCRNN	0.6467 ± 0.0035	0.6617 ± 0.0047	0.6577 ± 0.0074	0.7245 ± 0.0249	0.7266 ± 0.0308	0.7644 ± 0.0297
Gain vs. best baseline	+0.5%	−28.4%	+33.9%	+8.6%	+2.9%	+9.3%

most medium and long horizons, while on PeMS-D4 the two models are strongest at short and medium horizons and remain competitive at longer ones. LightSTORM also achieves this with a substantially smaller parameter budget than STAEformer. Overall, these results suggest that the multi-order formulation remains effective even when supervision is restricted to nodes.

Empirical stability. We evaluate Theorem 1 by perturbing the weighted boundary operators of trained STORM models. We consider perturbation noise $\eta \in \{10^{-4}, 10^{-3}, 10^{-2}\}$ while keeping the model parameters, inputs, and initial states fixed. The results in Table 5 show that the discrepancy between clean and perturbed predictions increases approximately linearly with the noise level, while predictive NMSE remains nearly unchanged, consistent with the stability result of Theorem 1. Full experimental details and analysis are reported in Appendix G.3.

Ablation studies. Here, we study which components of STORM drive the forecasting gains. Table 6 compares full STORM, LightSTORM, intra-order-only and cross-order-only variants, and a topology-free recurrent backbone; full results are reported in Appendix H. Results show that removing topology substantially degrades long-horizon performance, while cross-order coupling retains a

Table 4: Node forecasting on PeMS-D4 and PeMS-D8 at representative horizons: final-step node NMSE (lower is better). **Bold**: best. Underline: second best.

Model	PeMS-D4				PeMS-D8			
	Params.	S=20	S=40	S=80	Params.	S=20	S=40	S=80
STAEformer	1.59–2.07M	0.1292	**0.1709**	**0.2196**	1.21–1.42M	0.1209	0.1925	**0.1965**
Graph WaveNet	308–346K	0.1494	0.2959	0.5130	305–344K	0.1181	0.2332	0.4346
STID	37.0–41.1K	0.1427	0.2700	0.3996	32.6–36.8K	0.1058	0.2053	0.3673
DCRNN	371K	0.1999	0.4091	0.8598	371K	0.1395	0.3285	0.7199
STORM	18.4–199K	0.1258	0.2196	0.2859	25.3–194K	0.1001	0.1599	0.2283
LightSTORM	33.9–65.9K	**0.1223**	0.2143	0.2885	25.7–65.9K	**0.0998**	**0.1438**	0.2007

Table 5: Empirical stability at $S = 80$ under increasing topology perturbations with noise η. Disc. denotes the output discrepancy between clean and perturbed predictions.

Dataset	Clean NMSE	$\eta = 10^{-4}$		$\eta = 10^{-3}$		$\eta = 10^{-2}$	
		Disc.	Perturbed NMSE	Disc.	Perturbed NMSE	Disc.	Perturbed NMSE
PeMS-D4	0.310179	0.0080	0.310173	0.0808	0.310123	0.7922	0.310049
PeMS-D8	0.237332	0.0072	0.237327	0.0697	0.237294	0.6872	0.237544
Water	0.646768	0.0017	0.646766	0.0172	0.646748	0.1659	0.646865

large part of the gain and intra-order-only filtering is weaker. Further ablations show that injecting topology into the hidden state, through hidden or both injection, substantially outperforms input-only injection at long horizons on the traffic datasets, suggesting that cross-order interactions inside the latent recurrent dynamics may be important for long-range forecasting. Increasing the hidden dimension improves performance, but reduced-width STORM variants and LightSTORM remain competitive, indicating that the gains are not explained by recurrent capacity alone.

Table 6: Topology ablation: final-step NMSE (lower is better). **Bold**: best. Underline: second best.

Model	Topology	PeMS-D8			PeMS-D4			Water		
		S=20	S=40	S=80	S=20	S=40	S=80	S=20	S=40	S=80
STORM	Full Hodge + cross-order	**0.096**	**0.160**	**0.231**	**0.121**	**0.225**	**0.316**	**0.352**	0.552	**0.642**
LightSTORM	Shared Dirac	0.098	0.165	0.246	0.124	0.230	0.346	0.414	0.550	0.688
STORM	Cross-order only	0.103	0.185	0.283	0.131	0.265	0.416	0.394	0.559	0.646
STORM	Intra-order only	0.124	0.237	0.459	0.146	0.312	0.506	0.448	**0.533**	0.655
Recurrent backbone	None	0.141	0.366	0.697	0.169	0.420	0.755	0.558	0.598	0.709

6 Conclusion

We introduced Simplicial Topological Recurrent Models (STORM), a family of recurrent architectures for multivariate time series on fixed simplicial complexes that jointly model spatial and temporal dependencies by embedding simplicial filters inside recurrent updates. STORM combines within-order Hodge-Laplacian filtering with cross-order boundary coupling, while LightSTORM uses shared Dirac-polynomial filters for a parameter-efficient alternative. We establish permutation equivariance, stability to admissible topology perturbations, and a finite-horizon Rademacher-complexity generalization bound. An empirical analysis further supports the stability behavior predicted by the theory. Across the multi-order forecasting experiments, STORM achieves the best performance at nearly all evaluated horizons, with its advantage becoming particularly pronounced at long horizons. The models are also flexible enough to perform node-only forecasting, where STORM and LightSTORM remain competitive with strong spatio-temporal graph models, with LightSTORM using up to roughly two orders of magnitude fewer parameters than other graph baselines. Ablations further isolate the contribution of the architectural components, showing the observed gains are associated with topology-aware recurrence, with hidden-state coupling becoming particularly important at long horizons. Together, these methodological and theoretical contributions provide a foundation for learning representations of dynamical signals over higher-order networks and open the way to recurrent topological models for complex scientific and real-world systems.

AI USE STATEMENT

In this work, we have not used generative AI tools for generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, clean and reformat dataset, or interpret results. Translation and support qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to improve readability and checking the grammar. We have reviewed all AI-assisted work, as the text was checked and edited by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide the information needed to reproduce our experimental results in Appendix F, including dataset construction and preprocessing, experimental protocols, hyperparameters, model configurations, and hardware. We also provide an anonymous `.zip` file as supplemental materials containing the source code and the datasets used in our experiments, for the reviewers.

REFERENCES

Mathieu Alain, So Takao, Brooks Paige, and Marc Peter Deisenroth. Gaussian processes on cellular complexes. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 879–905. PMLR, 2024. URL https://proceedings.mlr.press/v235/alain24a.html.

Sergio Barbarossa and Stefania Sardellitti. Topological signal processing over simplicial complexes. *IEEE Transactions on Signal Processing*, 68:2992–3007, 2020.

Federico Barbero, Cristian Bodnar, Haitz Sáez de Ocáriz Borde, Michael Bronstein, Petar Veličković, and Pietro Liò. Sheaf neural networks with connection laplacians. In Alexander Cloninger, Timothy Doster, Tegan Emerson, Manohar Kaul, Ira Ktena, Henry Kvinge, Nina Miolane, Bastian Rieck, Sarah Tymochko, and Guy Wolf (eds.), *Proceedings of Topological, Algebraic, and Geometric Learning Workshops 2022*, volume 196 of *Proceedings of Machine Learning Research*, pp. 28–36. PMLR, 25 Feb–22 Jul 2022. URL https://proceedings.mlr.press/v196/barbero22a.html.

Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of machine learning research*, 3(Nov):463–482, 2002.

Claudio Battiloro, Indro Spinelli, Lev Telyatnikov, Michael Bronstein, Simone Scardapane, and Paolo Di Lorenzo. From latent graph to latent topology inference: Differentiable cell complex module. In *International Conference on Learning Representations*, 2024a.

Claudio Battiloro, Lucia Testa, Lorenzo Giusti, Stefania Sardellitti, Paolo Di Lorenzo, and Sergio Barbarossa. Generalized simplicial attention neural networks. *IEEE Transactions on Signal and Information Processing over Networks*, 10:833–850, 2024b.

Claudio Battiloro, Zhiyang Wang, Hans Riess, Paolo Di Lorenzo, and Alejandro Ribeiro. Tangent bundle convolutional learning: from manifolds to cellular sheaves and back. *IEEE Transactions on Signal Processing*, 72:1892–1909, 2024c.

Federico Battiston, Enrico Amico, Alain Barrat, Ginestra Bianconi, Guilherme Ferraz de Arruda, Benedetta Franceschiello, Iacopo Iacopini, Sonia Kéfi, Vito Latora, Yamir Moreno, et al. The physics of higher-order interactions in complex systems. *Nature physics*, 17(10):1093–1098, 2021.

Maximilian Beck, Korbinian Pöppel, Markus Spanring, Andreas Auer, Oleksandra Prudnikova, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. xlstm: Extended long short-term memory. In *Thirty-eighth Conference on Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2405.04517.

Austin R Benson, Rediet Abebe, Michael T Schaub, Ali Jadbabaie, and Jon Kleinberg. Simplicial closure and higher-order link prediction. *Proceedings of the National Academy of Sciences*, 115 (48):E11221–E11230, 2018.

Maciej Besta, Florian Scheidl, Lukas Gianinazzi, Grzegorz Kwasniewski, Shachar Klaiman, Jürgen Müller, and Torsten Hoefler. Demystifying Higher-Order Graph Neural Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(3):2544–2565, 3 2026. doi: 10.1109/TPAMI.2025.3637114.

Cristian Bodnar, Fabrizio Frasca, Nina Otter, Yuguang Wang, Pietro Lio, Guido F Montufar, and Michael Bronstein. Weisfeiler and lehman go cellular: Cw networks. *Advances in neural information processing systems*, 34:2625–2640, 2021a.

Cristian Bodnar, Fabrizio Frasca, Yuguang Wang, Nina Otter, Guido F Montufar, Pietro Lio, and Michael Bronstein. Weisfeiler and lehman go topological: Message passing simplicial networks. In *International conference on machine learning*, pp. 1026–1037. PMLR, 2021b.

Cristian Bodnar, Francesco Di Giovanni, Benjamin Paul Chamberlain, Pietro Lio, and Michael M. Bronstein. Neural sheaf diffusion: A topological perspective on heterophily and oversmoothing in GNNs. In *Advances in Neural Information Processing Systems*, 2022.

Steven L Brunton and J Nathan Kutz. *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022.

Gunnar Carlsson, Vin De Silva, and Dmitriy Morozov. Zigzag persistent homology and real-valued functions. In *Proceedings of the twenty-fifth annual symposium on Computational geometry*, pp. 247–256, 2009.

Gunnar Carlsson, VIN De Silva, Sara Kališnik, and Dmitriy Morozov. Parametrized homology via zigzag persistence. *Algebraic & Geometric Topology*, 19(2):657–700, 2019.

Jinyin Chen, Xueke Wang, and Xuanheng Xu. Gc-lstm: Graph convolution embedded lstm for dynamic network link prediction. *Applied Intelligence*, 52(7):7513–7528, 2022a.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Yuzhou Chen, Yulia Gel, and H Vincent Poor. Time-conditioned dances with simplicial complexes: Zigzag filtration curve based supra-hodge convolution networks for time-series forecasting. *Advances in Neural Information Processing Systems*, 35:8940–8953, 2022b.

Yuzhou Chen, Sotiris Batsakis, and H Vincent Poor. Higher-order spatio-temporal neural networks for covid-19 forecasting. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

Jae Choi, Yuzhou Chen, Huikyo Lee, Hyun Kim, and Yulia R Gel. Snn-pde: Learning dynamic pdes from data with simplicial neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 11561–11569, 2024.

Andrea Cini, Ivan Marisca, and Cesare Alippi. Filling the g_ap_s: Multivariate time series imputation by graph neural networks. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=kOu3-S3wJ7.

Tommaso Coletta, Robin Delabays, I Adagideli, and Ph Jacquod. Topologically protected loop flows in high voltage ac power grids. *New Journal of Physics*, 18(10):103042, 2016.

Gabriele D'Acunto and Claudio Battiloro. The relativity of causal knowledge. In Silvia Chiappa and Sara Magliacane (eds.), *Proceedings of the Forty-first Conference on Uncertainty in Artificial Intelligence*, volume 286 of *Proceedings of Machine Learning Research*, pp. 863–881. PMLR, 21–25 Jul 2025. URL https://proceedings.mlr.press/v286/d-acunto25a.html.

Stefania Ebli, Michaël Defferrard, and Gard Spreemann. Simplicial neural networks. In *Topological Data Analysis and Beyond workshop at NeurIPS*, 2020. URL https://arxiv.org/abs/2010.03633.

Edelsbrunner, Letscher, and Zomorodian. Topological persistence and simplification. *Discrete & computational geometry*, 28(4):511–533, 2002.

Aref Einizade, Fragkiskos Malliaros, and Jhony H Giraldo. Continuous product graph neural networks. *Advances in Neural Information Processing Systems*, 37:90226–90252, 2024.

Aref Einizade, Dorina Thanou, Fragkiskos D. Malliaros, and Jhony H. Giraldo. Continuous simplicial neural networks. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/55c944c9d908af245fa658b0e02b9003-Abstract-Conference.html.

Yam Eitan, Yoav Gelberg, Guy Bar-Shalom, Fabrizio Frasca, Michael Bronstein, and Haggai Maron. Topological blind spots: Understanding and extending topological deep learning through the lens of expressivity. *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

Fernando Gama, Joan Bruna, and Alejandro Ribeiro. Stability properties of graph neural networks. *IEEE Transactions on Signal Processing*, 68:5680–5695, 2020.

L. Giusti, C. Battiloro, P. Di Lorenzo, S. Sardellitti, and S. Barbarossa. Simplicial attention neural networks, 2022. URL https://arxiv.org/abs/2203.07485.

Lorenzo Giusti, Claudio Battiloro, Lucia Testa, Paolo Di Lorenzo, Stefania Sardellitti, and Sergio Barbarossa. Cell attention networks. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2023.

Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, pp. 922–929, 2019.

Mustafa Hajij, Kyle Istvan, and Ghada Zamzmi. Cell complex neural networks. *arXiv preprint arXiv:2010.00743*, 2020.

Mustafa Hajij, Ghada Zamzmi, Theodore Papamarkou, Nina Miolane, Aldo Guzmán-Sáenz, Karthikeyan Natesan Ramamurthy, Tolga Birdal, Tamal K. Dey, Soham Mukherjee, Shreyas N. Samaga, Neal Livesay, Robin Walters, Paul Rosen, and Michael T. Schaub. Topological deep learning: Going beyond graph data, 2023. URL https://arxiv.org/abs/2206.00606.

Mustafa Hajij, Mathilde Papillon, Florian Frantzen, Jens Agerberg, Ibrahem AlJabea, Rubén Ballester, Claudio Battiloro, Guillermo Bernárdez, Tolga Birdal, Aiden Brent, Peter Chin, Sergio Escalera, Simone Fiorellino, Odin Hoff Gardaa, Gurusankar Gopalakrishnan, Devendra Govil, Josef Hoppe, Maneel Reddy Karri, Jude Khouja, Manuel Lecha, Neal Livesay, Jan Meißner, Soham Mukherjee, Alexander Nikitin, Theodore Papamarkou, Jaro Prílepok, Karthikeyan Natesan Ramamurthy, Paul Rosen, Aldo Guzmán-Sáenz, Alessandro Salatiello, Shreyas N. Samaga, Simone Scardapane, Michael T. Schaub, Luca Scofano, Indro Spinelli, Lev Telyatnikov, Quang Truong, Robin Walters, Maosheng Yang, Olga Zaghen, Ghada Zamzmi, Ali Zia, and Nina Miolane. Topox: A suite of python packages for machine learning on topological domains. *Journal of Machine Learning Research*, 25(374):1–8, 2024. URL http://jmlr.org/papers/v25/24-0110.html.

J. Hansen and R. Ghrist. Toward a spectral theory of cellular sheaves. *Journal of Applied and Computational Topology*, 2019.

Jakob Hansen and Thomas Gebhart. Sheaf neural networks, 2020.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Josef Hoppe, Vincent P. Grande, and Michael T. Schaub. Don't be afraid of cell complexes! an introduction from an applied perspective, 2026. URL https://arxiv.org/abs/2506.09726.

Elvin Isufi, Geert Leus, Baltasar Beferull-Lozano, Sergio Barbarossa, and Paolo Di Lorenzo. Topological signal processing and learning: Recent advances and future challenges. *Signal Processing*, 233:109930, 2025.

Ming Jin, Huan Yee Koh, Qingsong Wen, Daniele Zambon, Cesare Alippi, Geoffrey I Webb, Irwin King, and Shirui Pan. A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection. *IEEE transactions on pattern analysis and machine intelligence*, 46(12):10466–10485, 2024.

Sanjukta Krishnagopal and Ginestra Bianconi. Spectral detection of simplicial communities via hodge laplacians. *Physical Review E*, 104(6):064303, 2021.

Joshin Krishnan, Rohan Money, Baltasar Beferull-Lozano, and Elvin Isufi. Simplicial vector autoregressive models. *IEEE Transactions on Signal Processing*, 72:5454–5469, 2024. doi: 10.1109/TSP.2024.3503063.

Manuel Lecha, Andrea Cavallo, Francesca Dominici, Elvin Isufi, and Claudio Battiloro. Higher-order topological directionality and directed simplicial neural networks. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025.

Manuel Lecha, Andrea Cavallo, Francesca Dominici, Ran Levi, Alessio Del Bue, Elvin Isufi, Pietro Morerio, and Claudio Battiloro. Directed semi-simplicial learning with applications to brain activity decoding. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=YR3CNvFfCr.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations (ICLR)*, 2018. URL https://arxiv.org/abs/1707.01926.

Lek-Heng Lim. Hodge laplacians on graphs. *Siam Review*, 62(3):685–715, 2020.

Hangchen Liu, Zheng Dong, Renhe Jiang, Jiewen Deng, Jinliang Deng, Quanjun Chen, and Xuan Song. Spatio-temporal adaptive embedding makes vanilla transformer sota for traffic forecasting. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pp. 4125–4129, 2023.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting, 2024. URL https://arxiv.org/abs/2310.06625.

Soumen Majhi, Matjaž Perc, and Dibakar Ghosh. Dynamics on higher-order networks: A review. *Journal of the Royal Society Interface*, 19(188):20220043, 2022.

Lorenzo Marinucci, Claudio Battiloro, and Paolo Di Lorenzo. Topological adaptive least mean squares algorithms over simplicial complexes. *IEEE Transactions on Signal and Information Processing over Networks*, 2026.

Edward C Mitchell, Brittany Story, David Boothe, Piotr J Franaszczuk, and Vasileios Maroulas. A topological deep learning framework for neural spike decoding. *Biophysical Journal*, 123(17): 2781–2789, 2024.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

Chao Pan, Siheng Chen, and Antonio Ortega. Spatio-temporal graph scattering transform, 2021. URL https://arxiv.org/abs/2012.03363.

Theodore Papamarkou, Tolga Birdal, Michael M. Bronstein, Gunnar E. Carlsson, Justin Curry, Yue Gao, Mustafa Hajij, Roland Kwitt, Pietro Lio, Paolo Di Lorenzo, Vasileios Maroulas, Nina Miolane, Farzana Nasrin, Karthikeyan Natesan Ramamurthy, Bastian Rieck, Simone Scardapane, Michael T Schaub, Petar Veličković, Bei Wang, Yusu Wang, Guowei Wei, and Ghada Zamzmi.

Position: Topological deep learning is the new frontier for relational learning. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 39529–39555. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/papamarkou24a.html.

Mathilde Papillon, Sophia Sanborn, Mustafa Hajij, and Nina Miolane. Architectures of topological deep learning: A survey of message-passing topological neural networks. *arXiv preprint arXiv:2304.10031*, 2023.

Mathilde Papillon, Guillermo Bernardez, Claudio Battiloro, and Nina Miolane. TopoTune: A framework for generalized combinatorial complex neural networks. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 47924–47952. PMLR, 13–19 Jul 2025a. URL https://proceedings.mlr.press/v267/papillon25a.html.

Mathilde Papillon, Sophia Sanborn, Johan Mathe, Louisa Cornelis, Abby Bertics, Domas Buracas, Hansen J Lillemark, Christian Shewmake, Fatih Dinc, Xavier Pennec, and Nina Miolane. Beyond euclid: an illustrated guide to modern machine learning with geometric, topological, and algebraic structures. *Machine Learning: Science and Technology*, 6(3):031002, aug 2025b. doi: 10.1088/2632-2153/adf375. URL https://dx.doi.org/10.1088/2632-2153/adf375.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

T. Mitchell Roddenberry, Nicholas Glaze, and Santiago Segarra. Principled simplicial neural networks for trajectory prediction. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9020–9029. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/roddenberry21a.html.

T. Mitchell Roddenberry, Michael T. Schaub, and Mustafa Hajij. Signal processing on cell complexes. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8852–8856, 2022. doi: 10.1109/ICASSP43922.2022.9747233.

Yulia Rubanova, Ricky TQ Chen, and David K Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. *Advances in neural information processing systems*, 32, 2019.

Luana Ruiz, Fernando Gama, and Alejandro Ribeiro. Gated graph recurrent neural networks. *IEEE Transactions on Signal Processing*, 68:6303–6318, 2020.

Mohammad Sabbaqi and Elvin Isufi. Graph-time convolutional neural networks: Architecture and theoretical analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12): 14625–14638, 2023.

Stefania Sardellitti, Sergio Barbarossa, and Lucia Testa. Topological signal processing over cell complexes. In *2021 55th Asilomar Conference on Signals, Systems, and Computers*, pp. 1558–1562. IEEE, 2021.

Michael T Schaub, Yu Zhu, Jean-Baptiste Seby, T Mitchell Roddenberry, and Santiago Segarra. Signal processing on higher-order networks: Livin'on the edge... and beyond. *Signal Processing*, 187:108149, 2021.

Youngjoo Seo, Michaël Defferrard, Pierre Vandergheynst, and Xavier Bresson. Structured sequence modeling with graph convolutional recurrent networks. In *International conference on neural information processing*, pp. 362–373. Springer, 2018.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Zezhi Shao, Zhao Zhang, Fei Wang, Wei Wei, and Yongjun Xu. Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting. In *Proceedings of the 31st ACM international conference on information & knowledge management*, pp. 4454–4458, 2022.

Lev Telyatnikov, Guillermo Bernardez, Marco Montagna, Mustafa Hajij, Martin Carrasco, Pavlo Vasylenko, Mathilde Papillon, Ghada Zamzmi, Michael T. Schaub, Jonas Verhellen, Pavel Snopov, Bertran Miquel-Oliver, Manel Gil-Sorribes, Alexis Molina, Victor Guallar, Theodore Long, Julian Suk, Patryk Rygiel, Alexander Nikitin, Giordan Escalona, Michael Banf, Dominik Filipiak, Max Schattauer, Liliya Imasheva, Alvaro Martinez, Halley Fritze, Marissa Masden, Valentina Sánchez, Manuel Lecha, Andrea Cavallo, Claudio Battiloro, Matt Piekenbrock, Mauricio Tec, George Dasoulas, Nina Miolane, Simone Scardapane, and Theodore Papamarkou. Topobench: A framework for benchmarking topological deep learning, 2025. URL https://arxiv.org/abs/2406.06642.

Hao Tian and Reza Zafarani. Higher-order networks representation and learning: A survey. *ACM SIGKDD Explorations Newsletter*, 26(1):1–18, 2024.

U.S. Environmental Protection Agency. EPANET: Software to model the hydraulic and water quality behavior of water distribution piping systems. https://www.epa.gov/water-research/epanet, 2000.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121*, 2019.

Maosheng Yang and Elvin Isufi. Convolutional learning on simplicial complexes. *arXiv preprint arXiv:2301.11163*, 2023.

Maosheng Yang, Elvin Isufi, and Geert Leus. Simplicial convolutional neural networks. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8847–8851. IEEE, 2022a.

Maosheng Yang, Elvin Isufi, Michael T Schaub, and Geert Leus. Simplicial convolutional filters. *IEEE Transactions on Signal Processing*, 70:4633–4648, 2022b.

Maosheng Yang, Viacheslav Borovitskiy, and Elvin Isufi. Hodge-compositional edge Gaussian processes. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li (eds.), *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 3754–3762. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/yang24e.html.

Ruochen Yang, Frederic Sala, and Paul Bogdan. Efficient representation learning for higher-order data with simplicial complexes. In Bastian Rieck and Razvan Pascanu (eds.), *Proceedings of the First Learning on Graphs Conference*, volume 198 of *Proceedings of Machine Learning Research*, pp. 13:1–13:21. PMLR, 09–12 Dec 2022c. URL https://proceedings.mlr.press/v198/yang22a.html.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

Ling Zhao, Yujiao Song, Chao Zhang, Yu Liu, Pu Wang, Tao Lin, Min Deng, and Haifeng Li. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 21(9):3848–3858, 2020. doi: 10.1109/TITS.2019.2935152.

Chuanpan Zheng, Xiaoliang Fan, Cheng Wang, and Jianzhong Qi. Gman: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI conference on artificial intelligence*, pp. 1234–1241, 2020.

Yuxuan Zhou, Shang Gao, Dandan Guo, Xiaohui Wei, Jon Rokne, and Hui Wang. A Survey of Change Point Detection in Dynamic Graphs . *IEEE Transactions on Knowledge & Data Engineering*, 37(03):1030–1048, 2025. ISSN 1558-2191. doi: 10.1109/TKDE.2024.3523857. URL https://doi.ieeecomputersociety.org/10.1109/TKDE.2024.3523857.

Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. In *Proceedings of the twentieth annual symposium on Computational geometry*, pp. 347–356, 2004.

APPENDIX

A Extended Background **18**

A.1 Notation and dimensions 18

A.2 Simplicial complexes and boundary incidence matrices 19

A.3 Hodge Laplacians 19

A.4 Dirac operator 19

A.5 Signals on simplicial complexes 20

A.6 Convolutional filters on simplicial complexes 20

B Extended Related Work **21**

C LightSTORM Architecture **22**

D Computational Complexity Analysis **23**

D.1 STORM: Time Complexity 23

D.2 LightSTORM: Time Complexity 24

D.3 Space Complexity 24

E Theoretical Analysis **24**

E.1 Permutation Equivariance Proof 25

E.2 Orientation Equivariance 26

E.3 Stability Proof 26

E.4 Generalization Bound via Rademacher Complexity 33

F Extended Experimental Details **43**

F.1 Dataset Details 44

F.2 Joint Multi-Order Forecasting Setup 44

F.3 Node-Only Forecasting Setup 47

F.4 Capacity Control for iTransformer 47

F.5 Dataset Topology Analysis 48

F.6 Hardware and Software 49

G Further Experimental Results **49**

G.1 Joint Multi-Order Forecasting 49

G.2 Node-Only Forecasting Results 51

G.3 Empirical Stability Under Admissible Topology Perturbations 51

H Ablation Studies **53**

H.1 Role of Topology 53

H.2 Injection Mode and Model Capacity 54

H.3 Role of Increasing Topological Order 56
H.4 Per-Order Forecasting Diagnostics 56

I Limitations and Future Work **57**

A Extended Background

We introduce the topological objects and operators used throughout, defining the domain, the signals that live on it, and the matrix operators that couple them.

A.1 Notation and dimensions

Table 7 summarizes the main notation and matrix dimensions used throughout the paper.

Table 7: Main notation and dimensions. Here $N = \sum_{k=0}^{K} n_k$, H is the hidden dimension, $G = 4$ denotes the number of recurrent gate branches $\{i, z, o, \tau\}$, where z indexes the candidate-cell branch, P is the STORM polynomial degree, J is the LightSTORM Dirac-polynomial degree, and S is the forecasting horizon. For cross-order terms, $\mathbf{U}^{(k)}$ denotes either the input signal $\mathbf{X}_t^{(k)}$ or the previous hidden state $\mathbf{h}_{t-1}^{(k)}$.

Symbol	Dimension	Meaning
$\mathcal{S}_k$	–	Set of k-simplices
n_k	–	Number of k-simplices, $n_k = \lvert\mathcal{S}_k\rvert$
K	–	Maximum simplex order
N	–	Total number of simplices, $N = \sum_{k=0}^{K} n_k$
d_k	–	Input feature dimension at order k
H	–	Hidden-state dimension
S	–	Forecasting horizon
$\mathbf{X}_t^{(k)}$	$n_k \times d_k$	Input signal at order k and time t
$\mathbf{h}_t^{(k)}$	$n_k \times H$	Hidden state at order k
$\mathbf{c}_t^{(k)}$	$n_k \times H$	Cell state at order k
$\widetilde{\mathbf{c}}_t^{(k)}$	$n_k \times H$	Candidate cell update from the z-branch
$\mathbf{i}_t^{(k)}, \mathbf{f}_t^{(k)}, \mathbf{o}_t^{(k)}$	$n_k \times H$	Input, forget, and output gates
$\boldsymbol{\tau}_t^{(k)}$	$n_k \times H$	Learned memory timescales
$\mathbf{G}_g^{(k)}$	$n_k \times H$	Gate pre-activation for $g \in \{i, z, o, \tau\}$
$\mathbf{B}_k$	$n_{k-1} \times n_k$	Boundary incidence matrix from order k to $k-1$
$\mathbf{B}_k^\top$	$n_k \times n_{k-1}$	Lifts signals from order $k-1$ to k
$\mathbf{B}_{k+1}$	$n_k \times n_{k+1}$	Projects signals from order $k+1$ to k
$\mathbf{L}_k^\downarrow$	$n_k \times n_k$	Lower Hodge Laplacian, $\mathbf{B}_k^\top \mathbf{B}_k$
$\mathbf{L}_k^\uparrow$	$n_k \times n_k$	Upper Hodge Laplacian, $\mathbf{B}_{k+1}\mathbf{B}_{k+1}^\top$
$\mathbf{L}_k$	$n_k \times n_k$	Full Hodge Laplacian, $\mathbf{L}_k^\downarrow + \mathbf{L}_k^\uparrow$
$\mathbf{D}, \mathbf{D}^{(d)}, \mathbf{D}^{(u)}$	$N \times N$	Dirac operator and its downward/upward components
$\widetilde{\mathbf{U}}_\downarrow^{(k)}$	$n_k \times d_{k-1}$ or $n_k \times H$	Lower cross-order signal, $\mathbf{B}_k^\top \mathbf{U}^{(k-1)}$
$\widetilde{\mathbf{U}}_\uparrow^{(k)}$	$n_k \times d_{k+1}$ or $n_k \times H$	Upper cross-order signal, $\mathbf{B}_{k+1}\mathbf{U}^{(k+1)}$
$\boldsymbol{\Theta}_{g,\mathrm{id}}^{(k)}$	$d_k \times H$ or $H \times H$	STORM self coefficient for input or hidden branch
$\boldsymbol{\Theta}_{g,\downarrow,p}^{(k)}, \boldsymbol{\Theta}_{g,\uparrow,p}^{(k)}$	$d_k \times H$ or $H \times H$	STORM intra-order Hodge coefficients
$\boldsymbol{\Theta}_{g,\mathrm{inj},\downarrow,p}^{(k)}$	$d_{k-1} \times H$ or $H \times H$	Lower-injection coefficient
$\boldsymbol{\Theta}_{g,\mathrm{inj},\uparrow,p}^{(k)}$	$d_{k+1} \times H$ or $H \times H$	Upper-injection coefficient
$\mathbf{b}_g^{(k)}$	H	Gate bias at order k
$\boldsymbol{\Theta}_{g,0}$	$H \times H$	LightSTORM shared identity coefficient
$\boldsymbol{\Theta}_{g,p}^{(d)}, \boldsymbol{\Theta}_{g,p}^{(u)}$	$H \times H$	LightSTORM downward/upward Dirac coefficients
$\mathbf{W}_{\mathrm{out}}^{(k)}$	$H \times Sd_k$	Per-order linear decoder matrix
$\mathbf{b}_{\mathrm{out}}^{(k)}$	Sd_k	Per-order decoder bias
$\widehat{\mathbf{X}}_{t+1:t+S}^{(k)}$	$n_k \times Sd_k$	Multi-step prediction, often reshaped as $n_k \times S \times d_k$

A.2 SIMPLICIAL COMPLEXES AND BOUNDARY INCIDENCE MATRICES

Let $\mathcal{V}$ be a finite vertex set. A k-simplex is a subset of $\mathcal{V}$ with cardinality $k+1$. Thus, a 0-simplex is a node, a 1-simplex is an edge between two nodes, and a 2-simplex is a filled triangle determined by three nodes. We denote the set of k-simplices by $\mathcal{S}_k = \{\sigma_1^{(k)}, \ldots, \sigma_{n_k}^{(k)}\}$, where $n_k = |\mathcal{S}_k|$.

A *finite simplicial complex* $\mathcal{K}$ of order K is a collection of simplices up to order K satisfying the inclusion property: whenever a simplex $\sigma \in \mathcal{K}$, every nonempty subset of its vertices also belongs to $\mathcal{K}$. For example, if a filled triangle belongs to $\mathcal{K}$, then its three edges and three nodes also belong to $\mathcal{K}$. In this work, we consider complexes of order two, consisting of nodes, edges, and filled triangles, although the definitions extend to arbitrary order.

Adjacencies of simplices. A $(k-1)$-face of a k-simplex is a subset of it with cardinality k, and a $(k+1)$-coface is a superset with cardinality $k+2$. Two k-simplices are *lower adjacent* if they share a common $(k-1)$-face, and *upper adjacent* if they are both faces of a common $(k+1)$-coface. Consequently, nodes have no lower adjacencies and their upper adjacencies recover the usual graph connectivity, while highest-order simplices have no upper adjacencies. For edges, lower adjacency means sharing an endpoint node, whereas upper adjacency means belonging to a common filled triangle.

After fixing an orientation for every simplex, relationships between adjacent simplex orders are encoded by the oriented boundary incidence matrices

$$\mathbf{B}_k \in \{-1, 0, +1\}^{n_{k-1} \times n_k}.$$

Their entries are

$$(\mathbf{B}_k)_{ij} = \begin{cases} +1 & \text{if } \sigma_i^{(k-1)} \text{ is a face of } \sigma_j^{(k)} \text{ with agreeing orientation,} \\ -1 & \text{if } \sigma_i^{(k-1)} \text{ is a face of } \sigma_j^{(k)} \text{ with opposite orientation,} \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

Thus $\mathbf{B}_1 \in \mathbb{R}^{n_0 \times n_1}$ maps each edge to its endpoint nodes, and $\mathbf{B}_2 \in \mathbb{R}^{n_1 \times n_2}$ maps each filled triangle to its boundary edges, with signs determined by the chosen orientations. The boundary operators satisfy the chain-complex identity

$$\mathbf{B}_{k-1}\mathbf{B}_k = \mathbf{0},$$

which expresses the algebraic fact that the boundary of a boundary is empty.

A.3 HODGE LAPLACIANS

From the boundary incidence matrices we build the Hodge Laplacians, which encode how simplices of the same order are connected through their lower and upper adjacencies. For boundary orders, we use the convention $\mathbf{B}_0 = \mathbf{0}$ and $\mathbf{B}_{K+1} = \mathbf{0}$, so that $\mathbf{L}_0^{\downarrow} = \mathbf{0}$ and $\mathbf{L}_K^{\uparrow} = \mathbf{0}$. For each $k \in \{0, \ldots, K\}$, we define

$$\mathbf{L}_k^{\downarrow} = \mathbf{B}_k^{\top}\mathbf{B}_k, \qquad \mathbf{L}_k^{\uparrow} = \mathbf{B}_{k+1}\mathbf{B}_{k+1}^{\top}, \qquad \mathbf{L}_k = \mathbf{L}_k^{\downarrow} + \mathbf{L}_k^{\uparrow}. \tag{10}$$

Here $\mathbf{L}_k^{\downarrow}$ captures lower adjacencies between two k-simplices that share a common $(k-1)$-face, while $\mathbf{L}_k^{\uparrow}$ captures upper adjacencies through co-membership in a common $(k+1)$-simplex. At the boundary orders, these terms vanish whenever the corresponding adjacent simplex order does not exist. The full Hodge Laplacian $\mathbf{L}_k \in \mathbb{R}^{n_k \times n_k}$ is symmetric and positive semidefinite. The special case $k=0$ recovers the standard graph Laplacian $\mathbf{L}_0 = \mathbf{L}_0^{\uparrow} = \mathbf{B}_1\mathbf{B}_1^{\top}$.

A.4 DIRAC OPERATOR

The Dirac operator collects all boundary incidence matrices into a single block matrix acting jointly on stacked signals across simplex orders. Under our convention $\mathbf{B}_k \in \mathbb{R}^{n_{k-1} \times n_k}$, the operator $\mathbf{B}_k$ maps order-k signals to order-$(k-1)$ signals, while $\mathbf{B}_k^{\top}$ maps order-$(k-1)$ signals to order-k signals. Therefore, for $K=2$, the Dirac operator is

$$\mathbf{D} = \begin{bmatrix} \mathbf{0} & \mathbf{B}_1 & \mathbf{0} \\ \mathbf{B}_1^{\top} & \mathbf{0} & \mathbf{B}_2 \\ \mathbf{0} & \mathbf{B}_2^{\top} & \mathbf{0} \end{bmatrix}. \tag{11}$$

The transpose blocks appear in the lower-left positions because each block row must have the dimension of the target simplex order. Acting on a stacked signal $\mathbf{X} = [\mathbf{X}^{(0)}; \mathbf{X}^{(1)}; \mathbf{X}^{(2)}]$, this gives

$$\mathbf{D}\mathbf{X} = \begin{bmatrix} \mathbf{B}_1\mathbf{X}^{(1)} \\ \mathbf{B}_1^\top\mathbf{X}^{(0)} + \mathbf{B}_2\mathbf{X}^{(2)} \\ \mathbf{B}_2^\top\mathbf{X}^{(1)} \end{bmatrix},$$

which is dimensionally consistent at every order. A key property is

$$\mathbf{D}^2 = \mathrm{diag}(\mathbf{L}_0, \mathbf{L}_1, \mathbf{L}_2),$$

since the cross-order terms vanish by $\mathbf{B}_1\mathbf{B}_2 = \mathbf{0}$. More generally, $\mathbf{D}^2 = \mathrm{diag}(\mathbf{L}_0, \ldots, \mathbf{L}_K)$.

The Dirac operator admits a decomposition into components that couple order 1 to its lower and upper neighboring orders. For $K{=}2$, we write

$$\mathbf{D}^{(d)} = \begin{bmatrix} \mathbf{0} & \mathbf{B}_1 & \mathbf{0} \\ \mathbf{B}_1^\top & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \qquad \mathbf{D}^{(u)} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B}_2 \\ \mathbf{0} & \mathbf{B}_2^\top & \mathbf{0} \end{bmatrix}, \tag{12}$$

so that $\mathbf{D} = \mathbf{D}^{(d)} + \mathbf{D}^{(u)}$. Here $\mathbf{D}^{(d)}$ couples nodes and edges through $\mathbf{B}_1$, while $\mathbf{D}^{(u)}$ couples edges and triangles through $\mathbf{B}_2$. Consequently, $(\mathbf{D}^{(d)})^2$ recovers the lower Hodge interactions associated with $\mathbf{B}_1$, and $(\mathbf{D}^{(u)})^2$ recovers the upper Hodge interactions associated with $\mathbf{B}_2$. LightSTORM (Section C) uses separate polynomial filters in $\mathbf{D}^{(d)}$ and $\mathbf{D}^{(u)}$ with a single shared identity coefficient:

$$\mathbf{G}_g[\mathbf{S}] = \mathbf{S}\mathbf{\Theta}_{g,0} + \sum_{p=1}^{J} (\mathbf{D}^{(d)})^p \mathbf{S}\, \mathbf{\Theta}_{g,p}^{(d)} + \sum_{p=1}^{J} (\mathbf{D}^{(u)})^p \mathbf{S}\, \mathbf{\Theta}_{g,p}^{(u)}. \tag{13}$$

The identity term is therefore not double-counted; only the nonzero powers of the downward and upward Dirac components use separate coefficients.

A.5 Signals on simplicial complexes

At each discrete time t, we observe a feature matrix at every order: $\mathbf{X}_t^{(k)} \in \mathbb{R}^{n_k \times d_k}$, where row i contains the d_k features of simplex $\sigma_i^{(k)}$. Examples include node pressures at order 0, edge flows at order 1, and triangle circulations at order 2. The topology $\mathcal{K}$ is fixed; only the signals $\mathbf{X}_t^{(k)}$ evolve in time. When needed, we stack all orders into $\mathbf{X}_t = [\mathbf{X}_t^{(0)}; \mathbf{X}_t^{(1)}; \ldots; \mathbf{X}_t^{(K)}] \in \mathbb{R}^{(\sum_k n_k) \times d}$, projecting each order to a common feature dimension d if necessary.

A.6 Convolutional filters on simplicial complexes

A simplicial convolutional filter at order k uses a self coefficient and *separate* learnable matrix coefficients for the lower and upper Laplacians:

$$g(\mathbf{L}_k^\downarrow, \mathbf{L}_k^\uparrow)\, \mathbf{X}_t^{(k)} = \mathbf{X}_t^{(k)} \mathbf{\Theta}_{\mathrm{id}} + \sum_{p=1}^{P} (\mathbf{L}_k^\downarrow)^p \mathbf{X}_t^{(k)}\, \mathbf{\Theta}_p^\downarrow + \sum_{p=1}^{P} (\mathbf{L}_k^\uparrow)^p \mathbf{X}_t^{(k)}\, \mathbf{\Theta}_p^\uparrow =: \mathbf{Y}_t^{(k)}. \tag{14}$$

The matrix coefficients allow the model to learn asymmetric, multi-channel contributions from lower and upper adjacent neighbors, a key expressive advantage over filters using a single shared polynomial in $\mathbf{L}_k$. Because both $\mathbf{L}_k^\downarrow$ and $\mathbf{L}_k^\uparrow$ are sparse, $(\mathbf{L}_k^\downarrow)^r$ and $(\mathbf{L}_k^\uparrow)^r$ mix simplices at most r hops apart, giving a local receptive field controlled by the polynomial degree P.

The Hodge Laplacians act on $\mathbf{X}_t^{(k)}$ by left multiplication, mixing features *within* order k. Cross-order transfer is achieved by the boundary operators: signals are lifted from order $k{-}1$ to k as $\widetilde{\mathbf{S}}_\downarrow^{(k)} = \mathbf{B}_k^\top \mathbf{X}_t^{(k-1)}$, and projected from order $k{+}1$ to k as $\widetilde{\mathbf{S}}_\uparrow^{(k)} = \mathbf{B}_{k+1} \mathbf{X}_t^{(k+1)}$, consistent with the notation in equation 6. Analogous polynomial filters in the Dirac operator propagate information jointly within and across all orders on the stacked signal $\mathbf{X}_t$ via separate polynomials in $\mathbf{D}^{(d)}$ and $\mathbf{D}^{(u)}$, as in equation 8, and form the basis of the LightSTORM filter design.

For further background on topological signal processing, we refer to Lim (2020); Barbarossa & Sardellitti (2020); Schaub et al. (2021) and surveys of TDL Isufi et al. (2025); Papamarkou et al. (2024).

B Extended Related Work

We organize the broader literature along four axes: pure sequence models, graph-based spatiotemporal models, static topological deep learning, and dynamic topological models.

Pure sequence models. Recurrent neural networks, and in particular LSTMs Hochreiter & Schmidhuber (1997) and their variants, have long been the dominant approach for multivariate time series forecasting. More recently, Transformer-based architectures Vaswani et al. (2017) and their spatiotemporal adaptations have demonstrated strong performance by leveraging self-attention over temporal or variable dimensions. In particular, iTransformer Liu et al. (2024) applies self-attention over the variable dimension rather than the temporal dimension, learning all-pairs spatial relationships from data but at quadratic cost in the number of variables and without any structural prior. The xLSTM architecture Beck et al. (2024) revisits recurrent models with exponential gating, recovering competitive performance against Transformers. Continuous-time extensions such as Neural ODEs Chen et al. (2018) and their recurrent variants Rubanova et al. (2019) motivate principled handling of irregular time steps through learned dynamics; STORM adopts this idea through the continuous-time forget gate derived from the analytical solution of a linear decay ODE, without requiring an external ODE solver. While these models are expressive temporal learners, they treat each variable as an independent or pairwise-coupled signal, ignoring the geometric and topological structure of the underlying domain. This limitation is particularly pronounced at long forecasting horizons, where structural inductive biases become increasingly valuable for maintaining predictive accuracy, as confirmed by our experimental results.

Graph-based spatiotemporal models. Graphs provide a natural relational inductive bias for multivariate time series on irregular domains. A common strategy combines graph convolutions or diffusion operators with temporal encoders, yielding hybrid architectures for forecasting and imputation Yu et al. (2017); Li et al. (2018); Zhao et al. (2020); Chen et al. (2022a). A second line of work more tightly fuses spatial and temporal processing through graph-recurrent updates, graph-aware attention, or structured spatiotemporal blocks Seo et al. (2018); Wu et al. (2019); Guo et al. (2019); Zheng et al. (2020); Ruiz et al. (2020). A third family constructs a joint space-time graph as a product of spatial and temporal graphs and performs convolution on the resulting structure Sabbaqi & Isufi (2023); Pan et al. (2021); Einizade et al. (2024). Across these families, incorporating spatial structure consistently improves performance on networked time series Jin et al. (2024); Cini et al. (2022); Zhou et al. (2025). However, these approaches remain fundamentally graph-based: interactions are modeled as pairwise relations, and the learning paradigm is node-centric, missing group interactions Battiston et al. (2021); Krishnagopal & Bianconi (2021); Tian & Zafarani (2024); Majhi et al. (2022) and signals intrinsically defined on edges, triangles, or higher-order objects Coletta et al. (2016). STORM overcomes this limitation by operating on signals at every simplex order through topology-aware recurrent gates.

Topological deep learning on static domains. Simplicial and cell complexes extend graphs by representing nodes, edges, faces, and higher-dimensional cells within a unified algebraic framework Schaub et al. (2021); Hoppe et al. (2026); Besta et al. (2026); Benson et al. (2018), building on work in topological signal processing Barbarossa & Sardellitti (2020); Roddenberry et al. (2022); Sardellitti et al. (2021); Isufi et al. (2025); Marinucci et al. (2026). This motivates Topological Deep Learning Papamarkou et al. (2024); Hajij et al. (2023); Papillon et al. (2023; 2025b). Simplicial Neural Networks generalize graph convolutions through Hodge-based filtering Ebli et al. (2020); Lim (2020). Message-passing frameworks extend relational aggregation within and across dimensions Bodnar et al. (2021b); Hajij et al. (2020); Bodnar et al. (2021a); Battiloro et al. (2024a). Convolutional architectures such as simplicial convolutional filters (SCF), simplicial convolutional neural networks (SCNN) and simplicial complex convolutional neural networks (SCCNN) implement localized multi-hop filtering and explicitly couple simplex orders Yang et al. (2022a;c); Yang & Isufi (2023); Roddenberry et al. (2021). Attention-based models have incorporated Dirac operators and cell complex attention to generalize graph attention to higher-order domains Battiloro et al. (2024b); Giusti et al. (2022; 2023). Expressivity analyses of TDL models have revealed structural blind spots relative to classical WL tests Bodnar et al. (2021b;a); Eitan et al. (2025). Further extensions include sheaf networks Hansen & Ghrist (2019); Hansen & Gebhart (2020); Bodnar et al. (2022); Battiloro et al. (2024c); Barbero et al. (2022); D'Acunto & Battiloro (2025), semi-simplicial

networks Lecha et al. (2026; 2025), Gaussian processes on simplicial domains Yang et al. (2024); Alain et al. (2024), continuous architectures Einizade et al. (2025), and software benchmarks for TDL Hajij et al. (2024); Telyatnikov et al. (2025); Papillon et al. (2025a). These works establish topology as a strong inductive bias for static tasks, but process one snapshot at a time with no temporal state evolution. STORM builds on the same convolutional operators and introduces recurrent hidden-state dynamics directly on the simplicial complex, jointly learning spatial and temporal dependencies across all simplex orders.

Topological models for time-evolving signals. Only recently have methods been proposed that explicitly combine topological structure with temporal dynamics. In Chen et al. (2022b), zigzag persistent homology Carlsson et al. (2009; 2019) is used to build time-conditioned topological descriptors, summarized through zigzag filtration curves Edelsbrunner et al. (2002); Zomorodian & Carlsson (2004) and fed into a supra-Hodge convolution block followed by a GRU. While this injects topological information into temporal prediction, the model operates on precomputed summaries rather than end-to-end recurrent dynamics over multi-order signals. In Chen et al. (2023), higher-order information is propagated to node features via Hodge Laplacians, incidence matrices, and attention before a GRU, but the model remains node-centric: topology enriches node representations rather than supporting recurrent evolution on multiple simplex orders. In Mitchell et al. (2024), simplicial convolutional layers Ebli et al. (2020); Yang et al. (2022a) are applied to node, edge, and triangle signals before a recurrent network, but the recurrent module acts on flattened embeddings rather than topology-aware hidden states. In Choi et al. (2024), Hodge-Laplacian filtering on simplicial complexes is established for PDE dynamics on physical domains, with the same discrete Hodge decomposition $\mathbf{L}_k = \mathbf{B}_k^\top \mathbf{B}_k + \mathbf{B}_{k+1}\mathbf{B}_{k+1}^\top$ underlying our SCF operators. The work demonstrates that boundary operators can effectively couple information across simplex orders; however, topology is encoded once per forward pass and temporal integration is delegated to an external Adams ODE solver. STORM builds on the same Hodge-theoretic foundation and extends it to the recurrent forecasting setting, embedding topological operators inside the gates at every timestep and handling irregular timing through a learned continuous-time forget gate without an external solver. In Krishnan et al. (2024), a linear simplicial VAR model extends VAR to multi-order signals through structured simplicial convolutional filters coupling nodes, edges, and faces. This captures linear higher-order dynamics but without gated hidden states, limiting its ability to model nonlinear long-range dependencies, as confirmed by our experimental results.

C LIGHTSTORM ARCHITECTURE

LightSTORM replaces the per-order SCFs used by STORM with polynomial filters in the Dirac operator. The Dirac operator decomposes as $\mathbf{D} = \mathbf{D}^{(d)} + \mathbf{D}^{(u)}$, where $\mathbf{D}^{(d)}$ and $\mathbf{D}^{(u)}$ are the downward and upward components defined in equation 12. These components couple neighboring simplex orders through the boundary operators, providing a shared cross-order filtering mechanism across the whole complex.

At time t, after mapping each order to the hidden dimension when needed, we stack the order-wise input signals along the simplex dimension into $\mathbf{X} \in \mathbb{R}^{N\times H}$, where $N = \sum_{k=0}^{K} n_k$. The previous hidden states are stacked in the same way into $\mathbf{h} \in \mathbb{R}^{N\times H}$. In the main text, the branch index on the filter coefficients is suppressed for readability; here we write it explicitly. The LightSTORM input-branch contribution for gate g is

$$\mathbf{G}_{X,g} = \mathbf{X}\mathbf{\Theta}_{X,g,0} + \sum_{p=1}^{J} (\mathbf{D}^{(d)})^p \mathbf{X}\, \mathbf{\Theta}^{(d)}_{X,g,p} + \sum_{p=1}^{J} (\mathbf{D}^{(u)})^p \mathbf{X}\, \mathbf{\Theta}^{(u)}_{X,g,p}. \tag{15}$$

The hidden-branch contribution is obtained analogously:

$$\mathbf{G}_{h,g} = \mathbf{h}\mathbf{\Theta}_{h,g,0} + \sum_{p=1}^{J} (\mathbf{D}^{(d)})^p \mathbf{h}\, \mathbf{\Theta}^{(d)}_{h,g,p} + \sum_{p=1}^{J} (\mathbf{D}^{(u)})^p \mathbf{h}\, \mathbf{\Theta}^{(u)}_{h,g,p}. \tag{16}$$

For each branch, the coefficients $\mathbf{\Theta}_{\cdot,g,0}$, $\mathbf{\Theta}^{(d)}_{\cdot,g,p}$, and $\mathbf{\Theta}^{(u)}_{\cdot,g,p}$ belong to $\mathbb{R}^{H\times H}$ and are shared across simplex orders. The identity term uses one shared coefficient, while separate coefficients are used

for the nonzero powers of the downward and upward Dirac components. This order-sharing is the main source of the parameter reduction in LightSTORM.

The stacked outputs $\mathbf{G}_{X,g}, \mathbf{G}_{h,g} \in \mathbb{R}^{N\times H}$ are then partitioned by rows into order-k blocks $\mathbf{G}_{X,g}^{(k)}, \mathbf{G}_{h,g}^{(k)} \in \mathbb{R}^{n_k\times H}$. These blocks are used in the same gate pre-activation as STORM:

$$\mathbf{G}_g^{(k)} = \mathbf{G}_{X,g}^{(k)} + \mathbf{G}_{h,g}^{(k)} + \mathbf{1}_{n_k}(\mathbf{b}_g^{(k)})^\top. \tag{17}$$

The recurrent update then follows the same cell equations equation 3–equation 4 and the same exponential forget parameterization in equation 5. As in STORM, the input, hidden, and both configurations control whether cross-order Dirac filtering is active in the input branch, the hidden branch, or both branches.

Table 8: Comparison of the two proposed architectures.

	STORM	**LightSTORM**
Topological filter	Per-order simplicial conv.	Shared Dirac polynomial
Cross-order coupling	Boundary injection	Dirac powers
Weight sharing	Across simplices, not orders	Across simplices and orders
Parameters per gate	$\mathcal{O}((K{+}1)(4P{+}3)H^2)$	$\mathcal{O}((2J{+}1)H^2)$

Table 8 summarizes both variants.

D Computational Complexity Analysis

We analyze the two proposed architectures. Let $N = \sum_{k=0}^{K} n_k$, $d = \max_k d_k$, H be the hidden dimension, $G = 4$ the number of gate branches, T the sequence length, and B the batch size. Elementwise operations, biases, and the final linear readout are lower-order terms and are omitted. Nonexistent lower or upper operators are omitted from all sums.

The implementation precomputes the required operator powers once, stores them in sparse CSR format, and reuses them throughout training. Consequently, the cost of a forward pass depends on the nonzero counts of the powers themselves, which may exceed those of the first-order operators. Define

$$\mathcal{M}_L := \sum_{k=0}^{K} \sum_{a\in\{\downarrow,\uparrow\}} \sum_{p=1}^{P} \operatorname{nnz}((\mathbf{L}_k^a)^p)\,, \qquad \mathcal{M}_B := \sum_{k=1}^{K} \operatorname{nnz}(\mathbf{B}_k), \tag{18}$$

$$\mathcal{M}_D := \sum_{a\in\{d,u\}} \sum_{j=1}^{J} \operatorname{nnz}\Big((\mathbf{D}^{(a)})^j\Big)\,. \tag{19}$$

These quantities describe the stored sparsity after the one-time precomputation; the preprocessing cost is not incurred during a forward pass.

D.1 STORM: Time Complexity

Consider first the hidden-state filter with cross-order injection in both directions. The within-order terms cost $BH\mathcal{M}_L$. Filtering the injected signals adds at most another $BH\mathcal{M}_L$, while applying each boundary operator in both directions costs $2BH\mathcal{M}_B$. Each gate uses at most $4P + 3$ dense $H \times H$ coefficient matrices per order: one identity term, $2P$ within-order terms, and $2(P + 1)$ injection terms. Thus one hidden-state update is bounded by

$$C_h^{\mathrm{STORM}} \le B\left[2H(\mathcal{M}_L + \mathcal{M}_B) + NG(4P + 3)H^2\right]. \tag{20}$$

This is an upper bound: boundary orders and configurations with only input or hidden injection contain fewer terms.

The input branch is evaluated over the full sequence before the recurrent loop. Replacing the input feature dimensions by d gives the corresponding upper bound, and the learned cross-order input projections add $\mathcal{O}(TBNd^2)$. Combining the input and hidden branches gives

$$C_{\mathrm{STORM}} = \mathcal{O}\big(TB\left[(\mathcal{M}_L + \mathcal{M}_B)(H + d) + NG(4P + 3)(H^2 + dH) + Nd^2\right]\big)\,. \tag{21}$$

The first term is sparse topological propagation and the second is contraction with the learned filter coefficients. In the experiments $d \ll H$, so the hidden-state contribution in equation 20 dominates.

D.2 LightSTORM: Time Complexity

LightSTORM first maps every order to H features and stacks the result into an $N \times H$ signal. One Dirac-polynomial branch costs

$$C_{\text{filter}}^{\text{LightSTORM}} \leq B\left[H\mathcal{M}_D + NG(2J+1)H^2\right]. \tag{22}$$

Let $R \in \{1, 2\}$ be the number of topology-filtered branches among the input and hidden branches: $R = 1$ for input-only or hidden-only injection and $R = 2$ for both. A branch without topological filtering is a dense map from H to GH. Including the initial input projections, the sequence cost is

$$C_{\text{LightSTORM}} = \mathcal{O}\big(TB\left[RH\mathcal{M}_D + NG(2+2JR)H^2 + NdH\right]\big). \tag{23}$$

The Dirac coefficients are shared across all simplex orders; sharing reduces the number of parameters, but each coefficient is still applied to all N simplices.

D.3 Space Complexity

Filter parameters. For the hidden branch with cross-order injection, STORM has at most $4P+3$ $H \times H$ coefficient matrices per gate and order. LightSTORM has $2J+1$ such matrices per topology-filtered branch, shared across orders. Hence

$$\Theta_h^{\text{STORM}} \leq (K+1)G(4P+3)H^2, \qquad \Theta_{\text{filter}}^{\text{LightSTORM}} = G(2J+1)H^2. \tag{24}$$

For $K = 2$ and $P = J = 2$, the corresponding upper-bound ratio is

$$\frac{\Theta_h^{\text{STORM}}}{\Theta_{\text{filter}}^{\text{LightSTORM}}} = \frac{(K+1)(4P+3)}{2J+1} = \frac{33}{5} \approx 6.6. \tag{25}$$

This is a filter-branch comparison rather than a ratio of complete models. Input projections, biases, the injection mode, and the forecasting head also contribute to the totals reported in Appendix F.2.5.

Activation memory. During inference, the recurrent states require $\mathcal{O}(BNH)$ memory. During training, storing activations for backpropagation through T steps requires $\mathcal{O}(TBNH(G+P))$ memory for STORM and $\mathcal{O}(TBNH(G+J))$ for LightSTORM, up to constants determined by the injection mode. The fixed sparse topology buffers require $\mathcal{O}(\mathcal{M}_L + \mathcal{M}_B)$ stored entries for STORM and $\mathcal{O}(\mathcal{M}_D)$ for LightSTORM.

Spatial scaling and attention. For fixed polynomial degree and bounded simplex degree, $\mathcal{M}_L$, $\mathcal{M}_B$, and $\mathcal{M}_D$ are linear in N. Equations equation 21 and equation 23 therefore scale linearly with the number of simplices. For context, the iTransformer baseline applies an independent encoder to each simplex order. Its self-attention term scales as

$$\mathcal{O}\left(LBH\sum_{k=0}^{K} n_k^2\right), \tag{26}$$

with attention-score memory proportional to $LBn_{\text{heads}}\sum_k n_k^2$. Thus, STORM and LightSTORM replace within-order all-pairs interactions by sparse topology-structured mixing. This comparison concerns asymptotic spatial scaling; wall-clock time also depends on sequence length, kernel implementation, and hardware utilization.

E Theoretical Analysis

We prove three guarantees for STORM: permutation equivariance (Section E.1), perturbation stability (Section E.3), and a finite-horizon generalization bound with explicit temporal dependence (Section E.4). Each result addresses a distinct desideratum of the architecture.

E.1 Permutation Equivariance Proof

We prove Proposition 1 for STORM. For each order $k \in \{0, \dots, K\}$, let $\mathbf{P}_k \in \{0,1\}^{N_k \times N_k}$ be a permutation matrix acting on the k-simplices. The permuted complex is defined by $\mathbf{B}'_k := \mathbf{P}_{k-1}\mathbf{B}_k\mathbf{P}_k^\top$ $(k \geq 1)$, which implies

$$\mathbf{L}_k^{\downarrow\prime} = \mathbf{P}_k\mathbf{L}_k^{\downarrow}\mathbf{P}_k^\top, \qquad \mathbf{L}_k^{\uparrow\prime} = \mathbf{P}_k\mathbf{L}_k^{\uparrow}\mathbf{P}_k^\top.$$

The permuted signals and states are $\mathbf{X}_t^{(k)\prime} := \mathbf{P}_k\mathbf{X}_t^{(k)}$, $\mathbf{h}_t^{(k)\prime} := \mathbf{P}_k\mathbf{h}_t^{(k)}$, $\mathbf{c}_t^{(k)\prime} := \mathbf{P}_k\mathbf{c}_t^{(k)}$.

Assumptions **A1** (Pointwise nonlinearities). All nonlinearities (σ, $\tanh$, softplus, exp, clamp) are applied entrywise, so $\varphi(\mathbf{P}_k\mathbf{Z}) = \mathbf{P}_k\varphi(\mathbf{Z})$.

A2 (Feature-only projections). All weight matrices $\mathbf{\Theta}$ act by right-multiplication: $(\mathbf{P}_k\mathbf{Z})\mathbf{\Theta} = \mathbf{P}_k(\mathbf{Z}\mathbf{\Theta})$.

Lemma E.1 (Conjugation of matrix powers)**.** *Let* $\mathbf{L}' = \mathbf{P}\mathbf{L}\mathbf{P}^\top$. *Then for every* $\ell \geq 0$, $(\mathbf{L}')^\ell = \mathbf{P}\mathbf{L}^\ell\mathbf{P}^\top$.

Proof. By induction. The case ℓ=0 gives $\mathbf{I} = \mathbf{P}\mathbf{I}\mathbf{P}^\top$. If $(\mathbf{L}')^\ell = \mathbf{P}\mathbf{L}^\ell\mathbf{P}^\top$, then

$$(\mathbf{L}')^{\ell+1} = (\mathbf{P}\mathbf{L}^\ell\mathbf{P}^\top)(\mathbf{P}\mathbf{L}\mathbf{P}^\top) = \mathbf{P}\mathbf{L}^{\ell+1}\mathbf{P}^\top. \qquad \square$$

Lemma E.2 (Equivariance of the STORM operator)**.** *Fix order* k. *Let* $\mathbf{Z}^{(k)}$ *denote either* $\mathbf{X}_t^{(k)}$ *or* $\mathbf{h}_{t-1}^{(k)}$. *The STORM gate pre-activation is*

$$\begin{aligned}\mathbf{G}_{g,t}^{(k)} = \mathbf{Z}^{(k)}\mathbf{\Theta}_{g,\text{self}}^{(k)} &+ \sum_{p=1}^{P}\mathbf{L}_{k,\downarrow}^{p}\mathbf{Z}^{(k)}\mathbf{\Theta}_{g,\downarrow,p}^{(k)} + \sum_{p=1}^{P}\mathbf{L}_{k,\uparrow}^{p}\mathbf{Z}^{(k)}\mathbf{\Theta}_{g,\uparrow,p}^{(k)} \\ &+ \sum_{p=0}^{P}\mathbf{L}_{k,\downarrow}^{p}(\mathbf{B}_k^\top\mathbf{Z}^{(k-1)})\mathbf{\Theta}_{g,\text{inj}\downarrow,p}^{(k)} + \sum_{p=0}^{P}\mathbf{L}_{k,\uparrow}^{p}(\mathbf{B}_{k+1}\mathbf{Z}^{(k+1)})\mathbf{\Theta}_{g,\text{inj}\uparrow,p}^{(k)} + \mathbf{1}(\mathbf{b}_g^{(k)})^\top. \quad (27)\end{aligned}$$

Let $\mathbf{G}_{g,t}^{(k)\prime}$ *be the same operator on the permuted complex with* $\mathbf{Z}^{(j)\prime} = \mathbf{P}_j\mathbf{Z}^{(j)}$. *Then* $\mathbf{G}_{g,t}^{(k)\prime} = \mathbf{P}_k\mathbf{G}_{g,t}^{(k)}$.

Proof. We verify equivariance term by term.

Intra-order terms. By A2, $(\mathbf{P}_k\mathbf{Z}^{(k)})\mathbf{\Theta} = \mathbf{P}_k(\mathbf{Z}^{(k)}\mathbf{\Theta})$. For Laplacian terms, by Lemma E.1 and $\mathbf{P}_k^\top\mathbf{P}_k = \mathbf{I}$: $(\mathbf{L}'_{k,\downarrow})^p(\mathbf{P}_k\mathbf{Z}^{(k)}) = \mathbf{P}_k\mathbf{L}_{k,\downarrow}^p\mathbf{Z}^{(k)}$, and analogously for $\mathbf{L}_{k,\uparrow}$.

Lower injection. Since $(\mathbf{B}'_k)^\top = \mathbf{P}_k\mathbf{B}_k^\top\mathbf{P}_{k-1}^\top$: $(\mathbf{B}'_k)^\top\mathbf{Z}^{(k-1)\prime} = \mathbf{P}_k\mathbf{B}_k^\top\mathbf{Z}^{(k-1)}$. Applying Laplacian powers and right multiplication preserves the leading $\mathbf{P}_k$.

Upper injection. Similarly, $\mathbf{B}'_{k+1}\mathbf{Z}^{(k+1)\prime} = \mathbf{P}_k\mathbf{B}_{k+1}\mathbf{Z}^{(k+1)}$.

Bias. Since $\mathbf{P}_k\mathbf{1} = \mathbf{1}$, the bias term is invariant. Summing all contributions yields the result. $\square$

Lemma E.3 (Equivariance of the recurrent cell update)**.** *Fix* k. *Assume* $\mathbf{G}_{g,t}^{(k)\prime} = \mathbf{P}_k\mathbf{G}_{g,t}^{(k)}$ *and* $\mathbf{h}_{t-1}^{(k)\prime} = \mathbf{P}_k\mathbf{h}_{t-1}^{(k)}$, $\mathbf{c}_{t-1}^{(k)\prime} = \mathbf{P}_k\mathbf{c}_{t-1}^{(k)}$. *Then* $\mathbf{c}_t^{(k)\prime} = \mathbf{P}_k\mathbf{c}_t^{(k)}$ *and* $\mathbf{h}_t^{(k)\prime} = \mathbf{P}_k\mathbf{h}_t^{(k)}$.

Proof. Gates. By A1, $\sigma(\mathbf{G}') = \mathbf{P}_k\sigma(\mathbf{G})$ and $\tanh(\mathbf{G}') = \mathbf{P}_k\tanh(\mathbf{G})$. The forget gate $\mathbf{f}_t^{(k)} = \exp(-\Delta t/\boldsymbol{\tau}_t^{(k)})$ is a composition of pointwise operations, hence also commutes with $\mathbf{P}_k$.

Hadamard products. For any $\mathbf{A}, \mathbf{B}$: $(\mathbf{P}_k\mathbf{A}) \odot (\mathbf{P}_k\mathbf{B}) = \mathbf{P}_k(\mathbf{A} \odot \mathbf{B})$.

State updates.

$$\mathbf{c}'_t = \mathbf{f}' \odot \mathbf{c}'_{t-1} + \mathbf{i}' \odot \tilde{\mathbf{c}}' = \mathbf{P}_k\mathbf{c}_t, \qquad \mathbf{h}'_t = \mathbf{o}' \odot \tanh(\mathbf{c}'_t) = \mathbf{P}_k\mathbf{h}_t. \qquad \square$$

Proof of Proposition 1

Proof. By induction. The base case holds by definition. For the inductive step, Lemma E.2 gives equivariance of all gate pre-activations. Given the inductive hypothesis on $(\mathbf{h}_{t-1}, \mathbf{c}_{t-1})$, Lemma E.3 implies the updated states $(\mathbf{h}_t, \mathbf{c}_t)$ are also equivariant. □

E.2 Orientation Equivariance

Permutation equivariance, proved in Appendix E.1, is the equivariance property used by STORM. We do not claim full orientation equivariance for the current gated architecture.

To see why, let an orientation change at order k be represented by a diagonal sign matrix $\mathbf{Q}_k$ with entries in $\{-1, +1\}$. Boundary operators transform as

$$\mathbf{B}'_k = \mathbf{Q}_{k-1}\mathbf{B}_k\mathbf{Q}_k,$$

which implies

$$(\mathbf{L}_k^{\downarrow})' = \mathbf{Q}_k\mathbf{L}_k^{\downarrow}\mathbf{Q}_k, \qquad (\mathbf{L}_k^{\uparrow})' = \mathbf{Q}_k\mathbf{L}_k^{\uparrow}\mathbf{Q}_k.$$

Thus, the bias-free linear Hodge and boundary filtering operators are orientation equivariant under the corresponding sign changes of the signals.

However, the full STORM recurrent cell is not orientation equivariant in general. Bias terms do not commute with arbitrary sign flips, and the sigmoid, softplus, clamp, and exponential operations used in the input, output, and continuous-time forget gates are not orientation invariant when applied directly to sign-changing pre-activations. Moreover, because these gates multiply the candidate, cell, and hidden states, odd nonlinearities alone are not sufficient: the multiplicative gate values must not introduce orientation-dependent sign changes. Therefore, the current model should be understood as operating with a fixed deterministic choice of simplex orientations. This is standard in our experiments: all incidence matrices are constructed once and kept fixed across training and evaluation.

Designing a fully orientation-equivariant gated recurrent architecture would require separating orientation-equivariant candidate dynamics from orientation-invariant gating signals, which we leave for future work.

E.3 Stability Proof

We prove the general stability result stated in Theorem 1. We then give a sufficient contraction condition and derive the uniform-in-time BIBO result as Theorem E.1.

Setting. Fix a simplicial complex $\mathcal{K}$ of maximum order $K \geq 0$. For each $k \in \{0, \dots, K\}$, let N_k denote the number of k-simplices, with Hodge Laplacians $\mathbf{L}_k^{\downarrow}, \mathbf{L}_k^{\uparrow} \in \mathbb{R}^{N_k \times N_k}$ and boundary operators $\mathbf{B}_k^{\top} \in \mathbb{R}^{N_k \times N_{k-1}}$, $\mathbf{B}_{k+1} \in \mathbb{R}^{N_k \times N_{k+1}}$. At each time t, the model processes inputs $\mathbf{X}_t^{(k)} \in \mathbb{R}^{N_k \times F_{\text{in}}}$, hidden states $\mathbf{h}_t^{(k)} \in \mathbb{R}^{N_k \times F_h}$, and cell states $\mathbf{c}_t^{(k)} \in \mathbb{R}^{N_k \times F_h}$, initialized at zero.

For compactness, we denote the STORM gate operator at order k on $\mathcal{K}$ by $\Phi_{\mathcal{K}}^{(k)}$, so that $\Phi_{\mathcal{K},H,g}^{(k)}$ and $\Phi_{\mathcal{K},X,g}^{(k)}$ refer to the hidden-state and input branches of gate g, respectively. The full one-step state map of STORM on $\mathcal{K}$, with input $\mathbf{X}_t$, is denoted by $\Phi_{\mathcal{K}}(\cdot, \mathbf{X}_t)$.

Norms. We write $\|\cdot\|_F$ for the Frobenius norm, $\|\mathbf{A}\|_\infty := \max_{i,j} |A_{ij}|$, and define

$$\bar{e}_H(t) := \max_k \left\| \mathbf{h}_t^{(k)} - \widehat{\mathbf{h}}_t^{(k)} \right\|_F, \qquad \bar{e}_C(t) := \max_k \left\| \mathbf{c}_t^{(k)} - \widehat{\mathbf{c}}_t^{(k)} \right\|_F.$$

We repeatedly use the Hadamard norm inequality:

$$\|\mathbf{A} \odot \mathbf{B}\|_F \leq \|\mathbf{A}\|_\infty \|\mathbf{B}\|_F. \tag{28}$$

For the STORM operator at order k mapping $\mathbf{Z}^{(\cdot)} = (\mathbf{Z}^{(k-1)}, \mathbf{Z}^{(k)}, \mathbf{Z}^{(k+1)})$ to a single-order output, we define

$$\left\| \Phi_{\mathcal{K}}^{(k)} \right\|_{\text{op}} := \sup_{\mathbf{Z}^{(\cdot)} \neq 0} \frac{\left\| \Phi_{\mathcal{K}}^{(k)}[\mathbf{Z}^{(\cdot)}] \right\|_F}{\max_{j \in \{k-1,k,k+1\}} \left\| \mathbf{Z}^{(j)} \right\|_F}.$$

At boundary orders, nonexistent terms are omitted from the maximum.

Let $\mathcal{P}$ denote the set of block-diagonal permutation matrices that relabel simplices consistently across orders. For stacked outputs, define

$$\left\|\mathbf{Y}-\widehat{\mathbf{Y}}\right\|_{\mathcal{P}} := \min_{\mathbf{P}\in\mathcal{P}} \left\|\mathbf{Y}-\mathbf{P}^{\top}\widehat{\mathbf{Y}}\right\|_{F}.$$

The admissible perturbation model supplies a fixed correspondence between the two complexes; in that aligned labeling, the identity permutation is admissible.

Architecture. For each gate $g \in \{i, z, o, \tau\}$ and order k, the gate pre-activation is

$$\mathbf{G}_{g,t}^{(k)} = \Phi_{\mathcal{K},H,g}^{(k)}\big[\mathbf{h}_{t-1}^{(\cdot)}\big] + \Phi_{\mathcal{K},X,g}^{(k)}\big[\mathbf{X}_{t}^{(\cdot)}\big] + \mathbf{b}_{g}^{(k)}, \tag{29}$$

where each $\Phi_{\mathcal{K}}^{(k)}$ computes

$$\begin{aligned}\Phi_{\mathcal{K}}^{(k)}[\mathbf{Z}^{(\cdot)}] := \mathbf{Z}^{(k)}\mathbf{\Theta}_{\mathrm{id}} &+ \sum_{p=1}^{P}(\mathbf{L}_{k}^{\downarrow})^{p}\mathbf{Z}^{(k)}\mathbf{\Theta}_{\downarrow,p} + \sum_{p=1}^{P}(\mathbf{L}_{k}^{\uparrow})^{p}\mathbf{Z}^{(k)}\mathbf{\Theta}_{\uparrow,p} \\ &+ \sum_{p=0}^{P}(\mathbf{L}_{k}^{\downarrow})^{p}(\mathbf{B}_{k}^{\top}\mathbf{Z}^{(k-1)})\mathbf{\Theta}_{\mathrm{inj},\downarrow,p} \\ &+ \sum_{p=0}^{P}(\mathbf{L}_{k}^{\uparrow})^{p}(\mathbf{B}_{k+1}\mathbf{Z}^{(k+1)})\mathbf{\Theta}_{\mathrm{inj},\uparrow,p}.\end{aligned} \tag{30}$$

At boundary orders $k = 0$ and $k = K$, nonexistent terms are omitted. The recurrent cell update is

$$\mathbf{i}_{t}^{(k)} = \sigma(\mathbf{G}_{i,t}^{(k)}), \qquad \mathbf{o}_{t}^{(k)} = \sigma(\mathbf{G}_{o,t}^{(k)}), \qquad \widetilde{\mathbf{c}}_{t}^{(k)} = \tanh(\mathbf{G}_{z,t}^{(k)}), \tag{31}$$

$$\boldsymbol{\tau}_{t}^{(k)} = \mathrm{clamp}\big(\mathrm{softplus}(\mathbf{G}_{\tau,t}^{(k)}), \tau_{\min}, \tau_{\max}\big), \tag{32}$$

$$\mathbf{f}_{t}^{(k)} = \exp(-\Delta t \oslash \boldsymbol{\tau}_{t}^{(k)}), \tag{33}$$

$$\mathbf{c}_{t}^{(k)} = \mathbf{f}_{t}^{(k)} \odot \mathbf{c}_{t-1}^{(k)} + \mathbf{i}_{t}^{(k)} \odot \widetilde{\mathbf{c}}_{t}^{(k)}, \tag{34}$$

$$\mathbf{h}_{t}^{(k)} = \mathbf{o}_{t}^{(k)} \odot \tanh(\mathbf{c}_{t}^{(k)}). \tag{35}$$

The readout

$$\mathbf{Y}_{t}^{(k)} = \mathbf{h}_{t}^{(k)}\mathbf{W}_{\mathrm{out}}^{(k)} + \mathbf{b}_{\mathrm{out}}^{(k)}$$

does not depend on $\mathcal{K}$.

Here $\odot$ denotes elementwise multiplication, and $\oslash$ denotes elementwise division; equivalently,

$$[\mathbf{f}_{t}^{(k)}]_{ij} = \exp\Big(-\Delta t/[\boldsymbol{\tau}_{t}^{(k)}]_{ij}\Big).$$

Admissible operator perturbations. The stability analysis considers a fixed oriented simplicial complex $\mathcal{K}$ and a small admissible perturbation of its weighted topological operators. Thus $\widehat{\mathcal{K}}$ is not an arbitrary second simplicial complex: it has the same simplex sets, orientations, and boundary sparsity pattern as $\mathcal{K}$, but the nonzero incidence weights are slightly perturbed.

For each k, let

$$\widehat{\mathbf{B}}_{k} = \mathbf{B}_{k} + \Delta\mathbf{B}_{k}, \qquad \mathrm{supp}(\Delta\mathbf{B}_{k}) \subseteq \mathrm{supp}(\mathbf{B}_{k}),$$

with

$$\|\Delta\mathbf{B}_{k}\|_{\mathrm{op}} \le \eta\, r_{k}, \qquad r_{k} := \|\mathbf{B}_{k}\|_{\mathrm{op}}.$$

Equivalently, the nonzero entries of $\mathbf{B}_{k}$ may be viewed as weighted incidences whose weights are perturbed by a small relative amount. For sufficiently small η, the incidence pattern and orientations are preserved.

The associated Hodge Laplacians are assumed to satisfy relative operator perturbation bounds

$$\left\|\widehat{\mathbf{L}}_{k}^{\downarrow} - \mathbf{L}_{k}^{\downarrow}\right\|_{\mathrm{op}} \le C_{L}\,\eta\,\left\|\mathbf{L}_{k}^{\downarrow}\right\|_{\mathrm{op}}, \qquad \left\|\widehat{\mathbf{L}}_{k}^{\uparrow} - \mathbf{L}_{k}^{\uparrow}\right\|_{\mathrm{op}} \le C_{L}\,\eta\,\left\|\mathbf{L}_{k}^{\uparrow}\right\|_{\mathrm{op}},$$

for a constant $C_{L} < \infty$. This covers the case where the perturbed Laplacians are built from the perturbed weighted boundary operators. We assume throughout that η is small enough for the perturbed operators to remain admissible.

Assumptions.

Assumption E.1 (Perturbation size)**.** *Fix $0 < \eta_0 < \infty$. The admissible perturbation model above holds with $0 \le \eta \le \eta_0$, and the incidence pattern and orientations are preserved.*

Assumption E.2 (Bounded boundary operators)**.** *There exist finite constants $r_{k,\downarrow}$ and $r_{k,\uparrow}$ such that*

$$\left\|\mathbf{B}_k^\top\right\|_{\mathrm{op}} \le r_{k,\downarrow}, \qquad \|\mathbf{B}_{k+1}\|_{\mathrm{op}} \le r_{k,\uparrow}.$$

Assumption E.3 (Finite polynomial filters)**.** *The polynomial degree P is finite, and all SCF coefficient matrices have finite operator norm. The degree and coefficients are fixed and identical in the two runs. No unit-norm constraint is imposed on the resulting SCF operators.*

Assumption E.4 (Lipschitz nonlinearities)**.** *The nonlinearities satisfy: σ is $\frac{1}{4}$-Lipschitz,* $\tanh$ *is* 1*-Lipschitz,* $\mathrm{softplus}$ *is* 1*-Lipschitz, and* clamp *is* 1*-Lipschitz.*

Assumption E.5 (Bounded inputs and time constants)**.** *There exist constants $B_X < \infty$, $0 < \tau_{\min} \le \tau_{\max}$, and $0 < \Delta_{\min} \le \Delta_{\max}$ such that*

$$\max_k \left\|\mathbf{X}_t^{(k)}\right\|_F \le B_X$$

for all t, and

$$0 < \tau_{\min} \le \boldsymbol{\tau}_t^{(k)} \le \tau_{\max}, \qquad 0 < \Delta_{\min} \le \Delta t \le \Delta_{\max}.$$

Consequently,

$$\mathbf{f}_t^{(k)} \le \gamma := \exp(-\Delta_{\min}/\tau_{\max}) < 1$$

entrywise.

Discussion on assumptions. Assumptions E.1 and E.2 specify the admissible perturbation class. The boundary-operator bounds are finite because the reference complex is fixed and finite. In Assumption E.3, the filters have finite degree and finitely many finite learned coefficients. Thus this assumption holds for every fixed STORM model, and no filter normalization is required. Assumption E.4 holds for the nonlinearities used by STORM. Finally, the experimental inputs form finite normalized sequences, the timescales are clamped to fixed positive bounds, and the experiments use $\Delta t = 1$, satisfying Assumption E.5. On the resulting bounded state set, the finite-dimensional one-step map has a finite Lipschitz constant q. The additional requirement $q < 1$ is only needed for the uniform-in-time result; Proposition E.1 gives a sufficient condition for it.

Definition E.1 (State metric and contractivity)**.** *For any $\lambda > 0$, define the weighted state metric*

$$d_w(S, S') := \max_k \left\|\mathbf{h}^{(k)} - \mathbf{h}^{(k)\prime}\right\|_F + \lambda \max_k \left\|\mathbf{c}^{(k)} - \mathbf{c}^{(k)\prime}\right\|_F.$$

We say that the one-step STORM map is q-Lipschitz in d_w if

$$d_w(\Phi_{\mathcal{K}}(S, \mathbf{X}), \Phi_{\mathcal{K}}(S', \mathbf{X})) \le q\, d_w(S, S')$$

for all states S, S' and admissible inputs $\mathbf{X}$. If $q < 1$, we say that the map is q-contractive.

SCF Stability Lemma.

Lemma E.4 (SCF stability under admissible operator perturbations)**.** *Under Assumptions E.1–E.3 and the admissible operator perturbation model above, for any multi-order signal with $\max_k \left\|\mathbf{Z}^{(k)}\right\|_F \le B$,*

$$\left\|\Phi_{\widehat{\mathcal{K}}}^{(k)}[\mathbf{Z}^{(\cdot)}] - \Phi_{\mathcal{K}}^{(k)}[\mathbf{Z}^{(\cdot)}]\right\|_F \le c_1 B\eta + c_2 B\eta^2. \tag{36}$$

Proof. The isolated self-term in the SCF does not depend on the topology and therefore cancels. We only need to control the Hodge-filtering and cross-order injection terms.

Intra-order terms. By the admissible perturbation model,

$$\left\|\widehat{\mathbf{L}}_k^{\downarrow} - \mathbf{L}_k^{\downarrow}\right\|_{\mathrm{op}} \le C_L \eta \left\|\mathbf{L}_k^{\downarrow}\right\|_{\mathrm{op}}, \qquad \left\|\widehat{\mathbf{L}}_k^{\uparrow} - \mathbf{L}_k^{\uparrow}\right\|_{\mathrm{op}} \le C_L \eta \left\|\mathbf{L}_k^{\uparrow}\right\|_{\mathrm{op}}.$$

For every power $\ell \geq 1$, standard telescoping gives

$$(\widehat{\mathbf{L}})^\ell - \mathbf{L}^\ell = \sum_{j=0}^{\ell-1} (\widehat{\mathbf{L}})^j (\widehat{\mathbf{L}} - \mathbf{L}) \mathbf{L}^{\ell-1-j}.$$

Write $\Lambda := \|\mathbf{L}\|_{\mathrm{op}}$ and $\overline{\Lambda} := (1 + C_L \eta_0)\Lambda$. The perturbation assumption gives $\left\|\widehat{\mathbf{L}}\right\|_{\mathrm{op}} \leq \overline{\Lambda}$. Hence

$$\left\|(\widehat{\mathbf{L}})^\ell - \mathbf{L}^\ell\right\|_{\mathrm{op}} \leq \ell C_L \eta \Lambda \max\{\Lambda, \overline{\Lambda}\}^{\ell-1} =: C_\ell \eta.$$

The constant C_ℓ is finite and uniform for $0 \leq \eta \leq \eta_0$. Summing over the finitely many polynomial orders and using the finite coefficient norms in Assumption E.3 gives constants $c_\downarrow^{(1)}$ and $c_\uparrow^{(1)}$ such that the lower and upper intra-order contributions are bounded by

$$(c_\downarrow^{(1)} + c_\uparrow^{(1)}) B \eta.$$

Cross-order injection. For the lower injection,

$$\widetilde{\mathbf{Z}}_\downarrow^{(k)} = \mathbf{B}_k^\top \mathbf{Z}^{(k-1)}, \qquad \widehat{\widetilde{\mathbf{Z}}}_\downarrow^{(k)} = \widehat{\mathbf{B}}_k^\top \mathbf{Z}^{(k-1)}.$$

Hence

$$\left\|\widehat{\widetilde{\mathbf{Z}}}_\downarrow^{(k)} - \widetilde{\mathbf{Z}}_\downarrow^{(k)}\right\|_F = \left\|\Delta \mathbf{B}_k^\top \mathbf{Z}^{(k-1)}\right\|_F \leq \eta r_k B.$$

The same argument applies to the upper injection through $\Delta \mathbf{B}_{k+1}$. For the $p = 0$ injection terms, this directly gives an $\mathcal{O}(B\eta)$ contribution. For $p \geq 1$, the difference also includes the perturbation of the Laplacian polynomial. With $\widetilde{\mathbf{Z}} := \mathbf{B}^\top \mathbf{Z}$ and $\widehat{\widetilde{\mathbf{Z}}} := \widehat{\mathbf{B}}^\top \mathbf{Z}$, decompose one such term as

$$(\widehat{\mathbf{L}})^p \widehat{\widetilde{\mathbf{Z}}} - \mathbf{L}^p \widetilde{\mathbf{Z}} = \big((\widehat{\mathbf{L}})^p - \mathbf{L}^p\big)\widetilde{\mathbf{Z}} + (\widehat{\mathbf{L}})^p \big(\widehat{\widetilde{\mathbf{Z}}} - \widetilde{\mathbf{Z}}\big)$$

and use $\left\|\widetilde{\mathbf{Z}}\right\|_F \leq r_k B$, $\left\|\widehat{\widetilde{\mathbf{Z}}} - \widetilde{\mathbf{Z}}\right\|_F \leq \eta r_k B$, $\left\|(\widehat{\mathbf{L}})^p\right\|_{\mathrm{op}} \leq \overline{\Lambda}^p$, and the power bound above. Each injection term is therefore bounded by a finite constant times $B\eta$, uniformly for $0 \leq \eta \leq \eta_0$. Summing over the finitely many polynomial orders and coefficient matrices gives a finite constant $c_{\mathrm{inj}}^{(1)}$.

Combining the intra-order and injection contributions gives

$$\left\|\Phi_{\widehat{\mathcal{K}}}^{(k)}[\mathbf{Z}^{(\cdot)}] - \Phi_{\mathcal{K}}^{(k)}[\mathbf{Z}^{(\cdot)}]\right\|_F \leq c_1 B\eta + c_2 B\eta^2,$$

with

$$c_1 = c_\downarrow^{(1)} + c_\uparrow^{(1)} + c_{\mathrm{inj}}^{(1)}, \qquad c_2 = 0.$$

□

Proof of Theorem 1.

Proof. Fix the simplex correspondence given by the admissible perturbation model and use it to align the two complexes at each simplex order. By permutation equivariance (Proposition 1), we may use this alignment without loss of generality. Minimizing over $\mathcal{P}$ can only make the output discrepancy smaller.

Let $\mathcal{S}_t$ and $\widehat{\mathcal{S}}_t$ denote the full STORM states on $\mathcal{K}$ and $\widehat{\mathcal{K}}$, respectively, and define

$$e_t := d_w(\mathcal{S}_t, \widehat{\mathcal{S}}_t).$$

The initial states are aligned, so $e_0 = 0$.

Step 1: Readout. For each order k,

$$\mathbf{Y}_t^{(k)} - \widehat{\mathbf{Y}}_t^{(k)} = (\mathbf{h}_t^{(k)} - \widehat{\mathbf{h}}_t^{(k)}) \mathbf{W}_{\mathrm{out}}^{(k)}.$$

With

$$c_{\text{out}} := \max_k \left\| \mathbf{W}_{\text{out}}^{(k)} \right\|_{\text{op}},$$

we have

$$\left\| \mathbf{Y}_t - \widehat{\mathbf{Y}}_t \right\|_{\mathcal{P}} \le c_{\text{out}} \sqrt{K+1}\, \bar{e}_H(t) \le c_{\text{out}} \sqrt{K+1}\, e_t. \tag{37}$$

Step 2: Bounded states. Since $\mathbf{f}_t^{(k)} \le \gamma < 1$ entrywise and $\left\| \mathbf{i}_t^{(k)} \odot \widetilde{\mathbf{c}}_t^{(k)} \right\|_\infty \le 1$, equation 34 gives

$$\left\| \mathbf{c}_t^{(k)} \right\|_\infty \le \gamma \left\| \mathbf{c}_{t-1}^{(k)} \right\|_\infty + 1.$$

Since $\mathbf{c}_0^{(k)} = 0$, this yields

$$\left\| \mathbf{c}_t^{(k)} \right\|_\infty \le B_c := \frac{1}{1-\gamma}.$$

Moreover, $\left\| \mathbf{h}_t^{(k)} \right\|_\infty \le 1$ because $\mathbf{h}_t^{(k)} = \mathbf{o}_t^{(k)} \odot \tanh(\mathbf{c}_t^{(k)})$.

Step 3: One-step perturbation. Introduce the intermediate state

$$\widetilde{\mathcal{S}}_t := \Phi_{\mathcal{K}}(\widehat{\mathcal{S}}_{t-1}, \mathbf{X}_t),$$

namely the state obtained by applying the unperturbed topology $\mathcal{K}$ to the perturbed previous state $\widehat{\mathcal{S}}_{t-1}$ with the same input. Then

$$e_t = d_w(\mathcal{S}_t, \widehat{\mathcal{S}}_t) \le d_w(\mathcal{S}_t, \widetilde{\mathcal{S}}_t) + d_w(\widetilde{\mathcal{S}}_t, \widehat{\mathcal{S}}_t).$$

By the assumed q-Lipschitz property of the state map in Definition E.1,

$$d_w(\mathcal{S}_t, \widetilde{\mathcal{S}}_t) \le q e_{t-1}.$$

For the second term, $\widetilde{\mathcal{S}}_t$ and $\widehat{\mathcal{S}}_t$ share the same previous state $\widehat{\mathcal{S}}_{t-1}$ but evolve under $\mathcal{K}$ and $\widehat{\mathcal{K}}$, respectively. For each gate $g \in \{i, z, o, \tau\}$ and order k, the pre-activation difference $\mathbf{G}_{g,t}^{(k)}(\mathcal{K}) - \mathbf{G}_{g,t}^{(k)}(\widehat{\mathcal{K}})$ is bounded by Lemma E.4 applied to both the input and hidden-state branches, with signal bounds B_X for the input branch and $B_H := \max_k \sqrt{N_k F_h}$ for the hidden-state branch, since $\|\mathbf{h}_t^{(k)}\|_\infty \le 1$ implies $\|\mathbf{h}_t^{(k)}\|_F \le \sqrt{N_k F_h}$. The factor B_H is absorbed into the constants κ, κ'. Propagating through the Lipschitz gate nonlinearities (Assumption E.4) and the Hadamard-product structure of the cell update, following the same term-by-term decomposition as in the contraction proof of Proposition E.1, yields finite constants κ, κ' such that

$$d_w(\widetilde{\mathcal{S}}_t, \widehat{\mathcal{S}}_t) \le \kappa\eta + \kappa'\eta^2. \tag{38}$$

Step 4: Error recursion. Iterating equation 38 and using $e_0 = 0$ gives

$$e_t \le (\kappa\eta + \kappa'\eta^2) \sum_{s=0}^{t-1} q^s. \tag{39}$$

Combining equation 37 and equation 39 yields

$$\left\| \mathbf{Y}_t - \widehat{\mathbf{Y}}_t \right\|_{\mathcal{P}} \le C_{\text{out}} (\kappa\eta + \kappa'\eta^2) \sum_{s=0}^{t-1} q^s,$$

where $C_{\text{out}} := c_{\text{out}} \sqrt{K+1}$. If $q \le 1$, then $\sum_{s=0}^{t-1} q^s \le t$, so

$$\left\| \mathbf{Y}_t - \widehat{\mathbf{Y}}_t \right\|_{\mathcal{P}} \le \mathsf{C}\, t\, \eta + \mathsf{C}'\, t\, \eta^2,$$

with $\mathsf{C} := C_{\text{out}}\kappa$ and $\mathsf{C}' := C_{\text{out}}\kappa'$. □

Contraction Condition.

Proposition E.1 (Sufficient condition for contraction)**.** *Let*

$$\gamma := \exp(-\Delta_{\min}/\tau_{\max}) < 1, \qquad W_{\text{op}} := \max_{k,g} \left\| \Phi^{(k)}_{\mathcal{K},H,g} \right\|_{\text{op}},$$

and define

$$\alpha := \left(c_F B_c + \frac{5}{4} \right) W_{\text{op}}, \qquad \beta := \frac{1}{4} W_{\text{op}}, \qquad c_F := \frac{\Delta_{\max}}{\tau^2_{\min}}, \qquad B_c := \frac{1}{1-\gamma}.$$

For two trajectories driven by the same inputs with

$$\left\| \mathbf{c}^{(k)}_t \right\|_\infty, \left\| \mathbf{c}^{(k)\prime}_t \right\|_\infty \le B_c,$$

the one-step error dynamics satisfy

$$\begin{pmatrix} \bar{e}_C(t) \\ \bar{e}_H(t) \end{pmatrix} \le \underbrace{\begin{pmatrix} \gamma & \alpha \\ \gamma & \alpha+\beta \end{pmatrix}}_{=:\mathbf{M}} \begin{pmatrix} \bar{e}_C(t-1) \\ \bar{e}_H(t-1) \end{pmatrix}. \tag{40}$$

A sufficient condition for $\rho(\mathbf{M}) < 1$ is

$$\gamma + \alpha + \beta < 1 + \gamma\beta. \tag{41}$$

Equivalently, it is sufficient that

$$W_{\text{op}} < W_{\text{crit}} := \frac{1-\gamma}{c_F B_c + \frac{3}{2} - \frac{\gamma}{4}}.$$

When this holds, for every $q \in (\rho(\mathbf{M}), 1)$, there exists $\lambda > 0$ such that the one-step STORM map is q-contractive in the metric d_w of Definition E.1.

Proof. Define the order-wise state errors

$$\bar{e}_C(t) := \max_k \left\| \mathbf{c}^{(k)}_t - \mathbf{c}^{(k)\prime}_t \right\|_F, \qquad \bar{e}_H(t) := \max_k \left\| \mathbf{h}^{(k)}_t - \mathbf{h}^{(k)\prime}_t \right\|_F.$$

Cell-state error. From equation 34,

$$\mathbf{c}^{(k)}_t - \mathbf{c}^{(k)\prime}_t = \underbrace{\mathbf{f}^{(k)}_t \odot \left(\mathbf{c}^{(k)}_{t-1} - \mathbf{c}^{(k)\prime}_{t-1} \right)}_{\text{(I)}} + \underbrace{\left(\mathbf{f}^{(k)}_t - \mathbf{f}^{(k)\prime}_t \right) \odot \mathbf{c}^{(k)\prime}_{t-1}}_{\text{(II)}} + \underbrace{\mathbf{i}^{(k)}_t \odot \widetilde{\mathbf{c}}^{(k)}_t - \mathbf{i}^{(k)\prime}_t \odot \widetilde{\mathbf{c}}^{(k)\prime}_t}_{\text{(III)}}.$$

Term (I). Since $\left\| \mathbf{f}^{(k)}_t \right\|_\infty \le \gamma$, we have

$$\|\text{(I)}\|_F \le \gamma \left\| \mathbf{c}^{(k)}_{t-1} - \mathbf{c}^{(k)\prime}_{t-1} \right\|_F \le \gamma \bar{e}_C(t-1).$$

Term (II). Using $\left\| \mathbf{c}^{(k)\prime}_{t-1} \right\|_\infty \le B_c$, we obtain

$$\|\text{(II)}\|_F \le B_c \left\| \mathbf{f}^{(k)}_t - \mathbf{f}^{(k)\prime}_t \right\|_F.$$

The map $\mathbf{G}_\tau \mapsto \mathbf{f}$ is c_F-Lipschitz, because the softplus and clamp maps are 1-Lipschitz and the derivative of $\tau \mapsto \exp(-\Delta t/\tau)$ is bounded by c_F on the admissible interval. Moreover, by definition of W_{op},

$$\left\| \Phi^{(k)}_{\mathcal{K},H,\tau} \right\|_{\text{op}} \le W_{\text{op}}.$$

Therefore

$$\|\text{(II)}\|_F \le B_c c_F W_{\text{op}} \bar{e}_H(t-1).$$

Term (III). Add and subtract $\mathbf{i}_t^{(k)\prime} \odot \widetilde{\mathbf{c}}_t^{(k)}$. Then

$$\|(\text{III})\|_F \le \left\| \left(\mathbf{i}_t^{(k)} - \mathbf{i}_t^{(k)\prime} \right) \odot \widetilde{\mathbf{c}}_t^{(k)} \right\|_F + \left\| \mathbf{i}_t^{(k)\prime} \odot \left(\widetilde{\mathbf{c}}_t^{(k)} - \widetilde{\mathbf{c}}_t^{(k)\prime} \right) \right\|_F .$$

Since $\left\|\widetilde{\mathbf{c}}_t^{(k)}\right\|_\infty \le 1$, $\left\|\mathbf{i}_t^{(k)\prime}\right\|_\infty \le 1$, σ is $1/4$-Lipschitz, $\tanh$ is 1-Lipschitz, and $\left\|\Phi^{(k)}_{\mathcal{K},H,i}\right\|_{\text{op}} \le W_{\text{op}}$, $\left\|\Phi^{(k)}_{\mathcal{K},H,c}\right\|_{\text{op}} \le W_{\text{op}}$, we obtain

$$\|(\text{III})\|_F \le \frac{5}{4} W_{\text{op}} \bar{e}_H(t-1).$$

Combining terms (I)–(III) and taking the maximum over k:

$$\bar{e}_C(t) \le \gamma \bar{e}_C(t-1) + \alpha \bar{e}_H(t-1).$$

Hidden-state error. From equation 35, adding and subtracting $\mathbf{o}_t^{(k)\prime} \odot \tanh(\mathbf{c}_t^{(k)})$:

$$\left\|\mathbf{h}_t^{(k)} - \mathbf{h}_t^{(k)\prime}\right\|_F \le \left\| \left(\mathbf{o}_t^{(k)} - \mathbf{o}_t^{(k)\prime} \right) \odot \tanh(\mathbf{c}_t^{(k)}) \right\|_F + \left\| \mathbf{o}_t^{(k)\prime} \odot \left[\tanh(\mathbf{c}_t^{(k)}) - \tanh(\mathbf{c}_t^{(k)\prime}) \right] \right\|_F .$$

Using the same Lipschitz bounds and $\left\|\Phi^{(k)}_{\mathcal{K},H,o}\right\|_{\text{op}} \le W_{\text{op}}$:

$$\bar{e}_H(t) \le \gamma \bar{e}_C(t-1) + (\alpha + \beta) \bar{e}_H(t-1).$$

Together with the cell-state bound, this gives equation 40.

Spectral analysis. The characteristic polynomial of $\mathbf{M}$ is $\mu^2 - p\mu + r = 0$ with $p := \gamma + \alpha + \beta$ and $r := \gamma\beta$. By the Jury criterion, $\rho(\mathbf{M}) < 1$ if $p < 1 + r$ and $r < 1$. The first condition is exactly equation 41, and it implies $r = \gamma\beta < 1$. Solving for W_{op}:

$$\left(c_F B_c + \frac{3}{2} - \frac{\gamma}{4} \right) W_{\text{op}} < 1 - \gamma,$$

which gives $W_{\text{op}} < W_{\text{crit}}$.

Contractive metric. Fix $q \in (\rho(\mathbf{M}), 1)$. Since $\mathbf{M}$ is entrywise nonnegative and $q > \rho(\mathbf{M})$, the matrix $q\mathbf{I} - \mathbf{M}^\top$ is a nonsingular M-matrix, so $\mathbf{v} := (q\mathbf{I} - \mathbf{M}^\top)^{-1}\mathbf{1}$ is entrywise positive. After rescaling, $\mathbf{v} = (\lambda, 1)^\top$ for some $\lambda > 0$, and $d_w(\mathcal{S}_t, \mathcal{S}'_t) = \lambda \bar{e}_C(t) + \bar{e}_H(t) = \mathbf{v}^\top \mathbf{e}(t)$. Using the error recursion:

$$d_w(\mathcal{S}_t, \mathcal{S}'_t) = \mathbf{v}^\top \mathbf{e}(t) \le \mathbf{v}^\top \mathbf{M} \mathbf{e}(t-1) = (\mathbf{M}^\top \mathbf{v})^\top \mathbf{e}(t-1) \le q\, d_w(\mathcal{S}_{t-1}, \mathcal{S}'_{t-1}).$$

□

Remark E.1 (Contraction condition)**.** *Proposition E.1 gives a sufficient condition for uniform-in-time stability. We did not enforce this condition during training, and our experiments did not use spectral normalization or a contraction penalty. The finite-horizon result in Theorem 1 does not require this condition.*

BIBO Stability.

Theorem E.1 (BIBO Stability)**.** *Under Assumptions E.1–E.5 and the contraction condition $W_{\text{op}} < W_{\text{crit}}$, running STORM with identical parameters on $\mathcal{K}$ and $\widehat{\mathcal{K}}$ from the same zero initial state and with the same input sequence yields constants $C_{\text{BIBO}}, C'_{\text{BIBO}} < \infty$, independent of t, such that, for every $t \ge 1$,*

$$\left\| \mathbf{Y}_t - \widehat{\mathbf{Y}}_t \right\|_{\mathcal{P}} \le C_{\text{BIBO}} \eta + C'_{\text{BIBO}} \eta^2.$$

Proof. The contraction condition $W_{\mathrm{op}} < W_{\mathrm{crit}}$ assumed in Theorem E.1 ensures, by Proposition E.1, that the one-step STORM state map is q-contractive in d_w for some $q < 1$. Applying Theorem 1 with this q and using

$$\sum_{s=0}^{t-1} q^s \le \frac{1}{1-q},$$

we obtain

$$\left\|\mathbf{Y}_t - \widehat{\mathbf{Y}}_t\right\|_{\mathcal{P}} \le \frac{C_{\mathrm{out}}\kappa}{1-q}\,\eta + \frac{C_{\mathrm{out}}\kappa'}{1-q}\,\eta^2 =: C_{\mathrm{BIBO}}\,\eta + C'_{\mathrm{BIBO}}\,\eta^2,$$

uniformly in t. □

Remark E.2 (Comparison with graph stability results)**.** *Related stability results for graph neural networks are given by Gama et al. (2020). For recurrent graph models, the GRNN stability bound of Ruiz et al. (2020) grows as $\mathcal{O}(t^2\eta)$: the t^2 term arises from state perturbations accumulating over time and from a topology-dependent readout. In STORM, the readout is topology-independent, and the recurrent forget gate controls how topology perturbations accumulate. Under the non-expansive condition $q \le 1$, Theorem 1 gives at most linear growth, $\mathcal{O}(t\eta)$. Under the stricter contraction condition of Proposition E.1, the geometric accumulation is summable and the bound becomes uniform in time, $\mathcal{O}(\eta)$, in the BIBO regime of Theorem E.1. Thus, stronger contraction improves robustness to persistent topology perturbations, although imposing too strong a contraction may reduce the model's ability to retain long-range state information.*

E.4 Generalization Bound via Rademacher Complexity

We prove the finite-horizon generalization bound in Theorem 2. The main idea is to control how a change in the shared STORM parameters affects the final prediction. We first show that the recurrent states remain bounded, then bound the sensitivity of one STORM update to changes in the state and in the recurrent parameters. Unrolling these bounds over T steps gives a Lipschitz bound for the predictor with respect to its parameters, which we use to control the Rademacher complexity of the loss class.

Setting. Let $\mathcal{K}$ be a fixed simplicial complex of maximum order K. Let $\mathcal{F}_T = \{f_\theta : \theta \in \Theta\}$ denote the class of STORM predictors mapping an input sequence $\mathbf{X}_{1:T}$ to $\mathbf{Y}_T = f_\theta(\mathbf{X}_{1:T})$ through the STORM recurrence equation 34–equation 35 and the linear readout.

For a predictor f_θ, let

$$R(f_\theta) := \mathbb{E}_{(\mathbf{X},\mathbf{Y}^*)\sim\mathcal{D}}[\ell(f_\theta(\mathbf{X}), \mathbf{Y}^*)]$$

denote the expected risk and

$$\widehat{R}_M(f_\theta) := \frac{1}{M}\sum_{m=1}^{M} \ell(f_\theta(\mathbf{X}^{(m)}, \mathbf{Y}^{*(m)}))$$

the empirical risk.

Assumptions.

Assumption E.6 (Independent trajectories)**.** *The training sample*

$$\left\{(\mathbf{X}^{(m)}_{1:T}, \mathbf{Y}^{*(m)})\right\}_{m=1}^{M}$$

consists of M input–target pairs drawn independently from the same distribution $\mathcal{D}$.

Assumption E.7 (Bounded inputs)**.** *For every simplex order k, there exists $B_{X,k} < \infty$ such that*

$$\|\mathbf{X}_t^{(k)}\|_F \le B_{X,k}$$

almost surely for every t.

Assumption E.8 (Bounded Lipschitz loss)**.** *The loss $\ell(\widehat{\mathbf{Y}}, \mathbf{Y}^*)$ takes values in $[0, B_\ell]$ and is L_ℓ-Lipschitz in its prediction argument with respect to the stacked Frobenius norm*

$$\|\mathbf{Y}\|_{\mathrm{stk}} := \left(\sum_{k=0}^{K} \|\mathbf{Y}^{(k)}\|_F^2\right)^{1/2}.$$

Assumption E.9 (Bounded parameter class)**.** *The complete trainable parameter class $\Theta \subset \mathbb{R}^d$, containing the recurrent and readout parameters, is nonempty, fixed independently of the observed training sample, and has finite Euclidean diameter*

$$D_\Theta := \sup_{\theta,\theta' \in \Theta} \|\theta - \theta'\|_2 < \infty.$$

The Euclidean norm is obtained by concatenating all scalar parameter entries. We write $\theta = (\theta_{\mathrm{rec}}, \theta_{\mathrm{out}})$. Since Θ is nonempty and has finite diameter, all parameter blocks are uniformly bounded over Θ.

Assumption E.10 (Initialization, timescales, and timesteps)**.** *The recurrent state is initialized at $\mathcal{S}_0 = \mathbf{0}$. The recurrent timescales satisfy entrywise*

$$0 < \tau_{\min} \le \boldsymbol{\tau}_t^{(k)} \le \tau_{\max} < \infty,$$

and

$$0 < \Delta_{\min} \le \Delta t \le \Delta_{\max} < \infty.$$

The STORM timescale map is the clamped softplus map defined in the model and is therefore 1-Lipschitz. The bounds and timestep specification are fixed independently of the parameters and the observed training sample, and parameter comparisons use the same timestep at each update.

Assumption E.11 (Fixed finite topology)**.** *The simplicial complex and its operators are fixed, finite, and independent of the observed training sample. The architecture, including the hidden width H, polynomial degree P, and injection configuration, is also fixed. Define*

$$\Lambda_{k,\downarrow} := \|\mathbf{L}_k^{\downarrow}\|_{\mathrm{op}}, \qquad \Lambda_{k,\uparrow} := \|\mathbf{L}_k^{\uparrow}\|_{\mathrm{op}},$$

and

$$r_{k,\downarrow} := \|\mathbf{B}_k^{\top}\|_{\mathrm{op}}, \qquad r_{k,\uparrow} := \|\mathbf{B}_{k+1}\|_{\mathrm{op}},$$

with nonexistent operators omitted.

Notation. For a collection of matrices, write

$$\|\mathbf{Z}\|_{F,\max} := \max_{0 \le k \le K} \|\mathbf{Z}^{(k)}\|_F.$$

For a matrix, $\|\mathbf{A}\|_\infty := \max_{i,j} |A_{ij}|$. We use

$$\|\mathbf{A} \odot \mathbf{B}\|_F \le \|\mathbf{A}\|_\infty \|\mathbf{B}\|_F$$

throughout.

Let $\Phi_\theta(\mathcal{S}, \mathbf{X})$ denote one STORM update; it depends only on θ_{rec}. For two recurrent states $\mathcal{S} = (\mathbf{h}, \mathbf{c})$ and $\mathcal{S}' = (\mathbf{h}', \mathbf{c}')$, define

$$\bar{e}_H(\mathcal{S}, \mathcal{S}') := \max_{0 \le k \le K} \|\mathbf{h}^{(k)} - \mathbf{h}^{(k)\prime}\|_F,$$

$$\bar{e}_C(\mathcal{S}, \mathcal{S}') := \max_{0 \le k \le K} \|\mathbf{c}^{(k)} - \mathbf{c}^{(k)\prime}\|_F,$$

and, for $\lambda > 0$,

$$d_w(\mathcal{S}, \mathcal{S}') := \bar{e}_H(\mathcal{S}, \mathcal{S}') + \lambda \bar{e}_C(\mathcal{S}, \mathcal{S}'). \tag{42}$$

Let $\eta_X, \eta_H \in \{0, 1\}$ indicate whether cross-order injection is active in the input and hidden branches, respectively. All constants below are uniform over Θ and the admissible inputs.

E.4.1 Bounded reachable states

Define

$$\gamma := \exp\left(-\frac{\Delta_{\min}}{\tau_{\max}}\right) < 1, \qquad B_c := \frac{1}{1-\gamma}. \tag{43}$$

Consider

$$\mathfrak{S} := \left\{ (\mathbf{h}, \mathbf{c}) : \max_k \|\mathbf{h}^{(k)}\|_\infty \le 1,\ \max_k \|\mathbf{c}^{(k)}\|_\infty \le B_c \right\}. \tag{44}$$

Lemma E.5 (Invariant bounded state set)**.** *For every admissible input and every $\theta \in \Theta$,*

$$\Phi_\theta(\mathfrak{S}, \mathbf{X}) \subseteq \mathfrak{S}.$$

Consequently, since $\mathcal{S}_0 = \mathbf{0} \in \mathfrak{S}$,

$$\|\mathbf{c}_t^{(k)}\|_\infty \le B_c, \qquad \|\mathbf{h}_t^{(k)}\|_\infty \le 1$$

for every t, k. In particular,

$$\|\mathbf{h}_t^{(k)}\|_F \le B_{H,k} := \sqrt{n_k H}. \tag{45}$$

Proof. By E.10,

$$0 < [\mathbf{f}_t^{(k)}]_{ij} = \exp\left(-\frac{\Delta t}{[\boldsymbol{\tau}_t^{(k)}]_{ij}}\right) \le \gamma.$$

Moreover,

$$0 < [\mathbf{i}_t^{(k)}]_{ij} < 1, \qquad |[\widetilde{\mathbf{c}}_t^{(k)}]_{ij}| \le 1.$$

Therefore, whenever $\|\mathbf{c}_{t-1}^{(k)}\|_\infty \le B_c$,

$$\begin{aligned}\|\mathbf{c}_t^{(k)}\|_\infty &\le \gamma \|\mathbf{c}_{t-1}^{(k)}\|_\infty + 1 \\ &\le \gamma B_c + 1 = B_c.\end{aligned}$$

For the hidden state,

$$\mathbf{h}_t^{(k)} = \mathbf{o}_t^{(k)} \odot \tanh(\mathbf{c}_t^{(k)}),$$

whose two factors have entrywise absolute value at most one. Thus

$$\|\mathbf{h}_t^{(k)}\|_\infty \le 1.$$

Hence $\mathfrak{S}$ is invariant. Since $\mathcal{S}_0 = \mathbf{0} \in \mathfrak{S}$, the result follows by induction. The Frobenius bound follows from

$$\|\mathbf{h}_t^{(k)}\|_F \le \sqrt{n_k H}\|\mathbf{h}_t^{(k)}\|_\infty.$$

□

E.4.2 FORGET-MAP SENSITIVITY

For fixed $\Delta > 0$, let

$$F_\Delta(\tau) = e^{-\Delta/\tau}.$$

Then

$$F'_\Delta(\tau) = \frac{\Delta}{\tau^2} e^{-\Delta/\tau}.$$

Hence

$$|F'_\Delta(\tau)| \le \frac{\Delta_{\max}}{\tau_{\min}^2}.$$

Alternatively, setting $y = \Delta/\tau$,

$$\frac{\Delta}{\tau^2} e^{-\Delta/\tau} = \frac{y^2 e^{-y}}{\Delta},$$

and

$$\sup_{y>0} y^2 e^{-y} = \frac{4}{e^2}.$$

Thus we may take

$$c_F := \min\left\{\frac{\Delta_{\max}}{\tau_{\min}^2}, \frac{4}{e^2 \Delta_{\min}}\right\}. \tag{46}$$

Since softplus and clamp are both 1-Lipschitz,

$$\|\mathbf{f} - \mathbf{f}'\|_F \le c_F \|\mathbf{G}_\tau - \mathbf{G}'_\tau\|_F. \tag{47}$$

E.4.3 STATE SENSITIVITY

With the time index omitted, the gate pre-activations are

$$\mathbf{G}_g^{(k)} = \Phi_{\mathcal{K},H,g}^{(k)}(\theta)[\mathbf{h}^{(\cdot)}] + \Phi_{\mathcal{K},X,g}^{(k)}(\theta)[\mathbf{X}^{(\cdot)}] + \mathbf{1}_{n_k}(\mathbf{b}_g^{(k)})^\top, \qquad g \in \{i, z, o, \tau\},$$

where the two branch operators are the simplicial filters defined in the method.

For a branch operator mapping a collection of source-order matrices to an order-k matrix, its induced norm is

$$\left\|\Phi_{\mathcal{K},H,g}^{(k)}(\theta)\right\|_{\mathrm{op}} := \sup_{\mathbf{Z}\neq 0} \frac{\|\Phi_{\mathcal{K},H,g}^{(k)}(\theta)[\mathbf{Z}]\|_F}{\|\mathbf{Z}\|_{F,\max}}.$$

For each gate $g \in \{i, z, o, \tau\}$, define a bound over the whole parameter class:

$$\overline{W}_g := \sup_{\theta\in\Theta} \max_{0\le k\le K} \left\|\Phi_{\mathcal{K},H,g}^{(k)}(\theta)\right\|_{\mathrm{op}}. \tag{48}$$

These quantities are finite by and E.11. For the STORM filters, an explicit upper bound is

$$\begin{aligned}
\omega_{g,k}(\theta) := \|\mathbf{\Theta}_{g,H,\mathrm{id}}^{(k)}\|_{\mathrm{op}} &\\
&+ \mathbf{1}_{\{k\ge 1\}} \sum_{p=1}^{P} \Lambda_{k,\downarrow}^p \|\mathbf{\Theta}_{g,H,\downarrow,p}^{(k)}\|_{\mathrm{op}} \\
&+ \mathbf{1}_{\{k\le K-1\}} \sum_{p=1}^{P} \Lambda_{k,\uparrow}^p \|\mathbf{\Theta}_{g,H,\uparrow,p}^{(k)}\|_{\mathrm{op}} \\
&+ \eta_H \mathbf{1}_{\{k\ge 1\}} r_{k,\downarrow} \sum_{p=0}^{P} \Lambda_{k,\downarrow}^p \|\mathbf{\Theta}_{g,H,\mathrm{inj},\downarrow,p}^{(k)}\|_{\mathrm{op}} \\
&+ \eta_H \mathbf{1}_{\{k\le K-1\}} r_{k,\uparrow} \sum_{p=0}^{P} \Lambda_{k,\uparrow}^p \|\mathbf{\Theta}_{g,H,\mathrm{inj},\uparrow,p}^{(k)}\|_{\mathrm{op}},
\end{aligned} \tag{49}$$

for which

$$\left\|\Phi_{\mathcal{K},H,g}^{(k)}(\theta)\right\|_{\mathrm{op}} \le \omega_{g,k}(\theta).$$

Define

$$\alpha := B_c c_F \overline{W}_\tau + \frac{1}{4}\overline{W}_i + \overline{W}_z, \qquad \beta := \frac{1}{4}\overline{W}_o. \tag{50}$$

Lemma E.6 (Uniform state sensitivity on the invariant set)**.** *For every $\lambda > 0$, every $\theta \in \Theta$, every admissible input $\mathbf{X}$, and every $\mathcal{S}, \mathcal{S}' \in \mathfrak{S}$,*

$$d_w(\Phi_\theta(\mathcal{S}, \mathbf{X}), \Phi_\theta(\mathcal{S}', \mathbf{X})) \le q_\lambda d_w(\mathcal{S}, \mathcal{S}'),$$

where

$$q_\lambda := \max\left\{\gamma\left(1 + \frac{1}{\lambda}\right), \alpha(1+\lambda) + \beta\right\}. \tag{51}$$

Proof. Let

$$\bar{e}_C = \bar{e}_C(\mathcal{S}, \mathcal{S}'), \qquad \bar{e}_H = \bar{e}_H(\mathcal{S}, \mathcal{S}').$$

For any order k,

$$\begin{aligned}
\mathbf{c}^{(k)+} - \mathbf{c}^{(k)\prime+} = \mathbf{f}^{(k)} \odot (\mathbf{c}^{(k)} - \mathbf{c}^{(k)\prime}) &\\
+ (\mathbf{f}^{(k)} - \mathbf{f}^{(k)\prime}) \odot \mathbf{c}^{(k)\prime} &\\
+ \mathbf{i}^{(k)} \odot \widetilde{\mathbf{c}}^{(k)} - \mathbf{i}^{(k)\prime} \odot \widetilde{\mathbf{c}}^{(k)\prime}.
\end{aligned}$$

The first term is bounded by

$$\gamma \bar{e}_C.$$

Since $\mathcal{S}' \in \mathfrak{S}$, the second satisfies

$$\begin{aligned}\|(\mathbf{f}^{(k)} - \mathbf{f}^{(k)\prime}) \odot \mathbf{c}^{(k)\prime}\|_F &\le B_c c_F \|\mathbf{G}_\tau^{(k)} - \mathbf{G}_\tau^{(k)\prime}\|_F \\ &\le B_c c_F \overline{W}_\tau \bar{e}_H.\end{aligned}$$

For the final term, add and subtract $\mathbf{i}^{(k)\prime} \odot \widetilde{\mathbf{c}}^{(k)}$. Using the $1/4$-Lipschitzness of sigmoid and the 1-Lipschitzness of $\tanh$,

$$\begin{aligned}&\|\mathbf{i}^{(k)} \odot \widetilde{\mathbf{c}}^{(k)} - \mathbf{i}^{(k)\prime} \odot \widetilde{\mathbf{c}}^{(k)\prime}\|_F \\ &\le \left(\frac{1}{4}\overline{W}_i + \overline{W}_z\right) \bar{e}_H.\end{aligned}$$

Hence

$$\bar{e}_C^+ \le \gamma \bar{e}_C + \alpha \bar{e}_H. \tag{52}$$

For the hidden state,

$$\mathbf{h}^{(k)+} = \mathbf{o}^{(k)} \odot \tanh(\mathbf{c}^{(k)+}),$$

which yields

$$\bar{e}_H^+ \le \bar{e}_C^+ + \frac{1}{4}\overline{W}_o \bar{e}_H \le \gamma \bar{e}_C + (\alpha + \beta)\bar{e}_H.$$

Therefore

$$\begin{aligned}d_w^+ &= \bar{e}_H^+ + \lambda \bar{e}_C^+ \\ &\le \gamma(1+\lambda)\bar{e}_C + [\alpha(1+\lambda) + \beta]\bar{e}_H \\ &\le q_\lambda(\lambda \bar{e}_C + \bar{e}_H),\end{aligned}$$

which proves the claim. □

Thus q_λ bounds how much a state difference can change in one update. When $q_\lambda < 1$, the update contracts in the distance d_w.

Lemma E.7 (Condition for $q_\lambda < 1$)**.** *There exists $\lambda > 0$ such that $q_\lambda < 1$ if and only if*

$$\alpha < (1-\gamma)(1-\beta), \tag{53}$$

equivalently

$$\gamma + \alpha + \beta < 1 + \gamma\beta.$$

Proof. The condition $q_\lambda < 1$ requires simultaneously

$$\lambda > \frac{\gamma}{1-\gamma}$$

and, for $\alpha > 0$,

$$\lambda < \frac{1-\beta}{\alpha} - 1.$$

An admissible λ exists exactly when

$$\frac{\gamma}{1-\gamma} < \frac{1-\beta}{\alpha} - 1,$$

which rearranges to

$$\alpha < (1-\gamma)(1-\beta).$$

If $\alpha = 0$, then $q_\lambda = \max\{\gamma(1+1/\lambda), \beta\}$, so a suitable λ exists exactly when $\beta < 1$. The equivalent second form follows by expansion. This characterizes when the stated bound q_λ is below one; the actual recurrence may contract even when this bound is at least one. □

Remark E.3 (Status of the contraction condition)**.** *Condition equation 53 characterizes when the uniform upper bound q_λ is below one. It is not required for the finite-horizon theorem and is used only for the sequence-length-independent case. Since q_λ is an upper bound, the actual recurrence may still contract even when $q_\lambda \ge 1$.*

E.4.4 SENSITIVITY TO THE SHARED PARAMETERS

The generalization argument requires a bound on how the predictor changes when its trainable parameters change. Because the recurrent parameters are shared across time, the same parameter perturbation affects every STORM update. We first bound its effect on one update and then accumulate this effect over the sequence using the state-sensitivity bound above.

To obtain the one-step bound, suppose a filter branch receives matrices with $\|\mathbf{Z}^{(k)}\|_F \le B_k$, and write $B = (B_0, \dots, B_K)$. For $\eta \in \{0, 1\}$, define

$$\begin{aligned}\Xi_k^2(B, \eta) := B_k^2 \Bigg[1 + \mathbf{1}_{\{k\ge 1\}} \sum_{p=1}^{P} \Lambda_{k,\downarrow}^{2p} + \mathbf{1}_{\{k\le K-1\}} \sum_{p=1}^{P} \Lambda_{k,\uparrow}^{2p}\Bigg] \\ + \eta \mathbf{1}_{\{k\ge 1\}} r_{k,\downarrow}^2 B_{k-1}^2 \sum_{p=0}^{P} \Lambda_{k,\downarrow}^{2p} \\ + \eta \mathbf{1}_{\{k\le K-1\}} r_{k,\uparrow}^2 B_{k+1}^2 \sum_{p=0}^{P} \Lambda_{k,\uparrow}^{2p}. \end{aligned} \tag{54}$$

Let

$$B_X = (B_{X,0}, \dots, B_{X,K}), \qquad B_H = (B_{H,0}, \dots, B_{H,K}),$$

and define

$$A_{\mathrm{rec}} := \max_k \left[\Xi_k^2(B_X, \eta_X) + \Xi_k^2(B_H, \eta_H) + n_k\right]^{1/2}. \tag{55}$$

The term n_k accounts for the broadcast recurrent bias, since

$$\|\mathbf{1}_{n_k}(\mathbf{b} - \mathbf{b}')^\top\|_F = \sqrt{n_k}\|\mathbf{b} - \mathbf{b}'\|_2.$$

The two Ξ_k^2 terms account for the input and hidden filters. Together with the bias term, they bound the effect of changing all parameters of one gate. Let θ_g denote all recurrent filter matrices and the bias associated with gate g.

Lemma E.8 (Gate sensitivity to parameter perturbations)**.** *For every fixed $\mathcal{S} \in \mathfrak{S}$, every admissible input $\mathbf{X}$, every $\theta, \theta' \in \Theta$, and every $g \in \{i, z, o, \tau\}$,*

$$\max_k \|\mathbf{G}_g^{(k)}(\theta) - \mathbf{G}_g^{(k)}(\theta')\|_F \le A_{\mathrm{rec}}\|\theta_g - \theta_g'\|_2.$$

Proof. At a fixed target order, every trainable filter contribution has the form

$$\mathbf{A}_j \mathbf{Z}^{(s_j)} \mathbf{\Theta}_j.$$

Holding $\mathbf{Z}$ fixed,

$$\|\mathbf{A}_j \mathbf{Z}^{(s_j)}(\mathbf{\Theta}_j - \mathbf{\Theta}_j')\|_F \le \|\mathbf{A}_j\|_{\mathrm{op}}\|\mathbf{Z}^{(s_j)}\|_F\|\mathbf{\Theta}_j - \mathbf{\Theta}_j'\|_F.$$

The topology operators satisfy

$$\|(\mathbf{L}_k^{\downarrow})^p\|_{\mathrm{op}} \le \Lambda_{k,\downarrow}^p, \qquad \|(\mathbf{L}_k^{\uparrow})^p\|_{\mathrm{op}} \le \Lambda_{k,\uparrow}^p,$$

and

$$\|(\mathbf{L}_k^{\downarrow})^p \mathbf{B}_k^\top\|_{\mathrm{op}} \le \Lambda_{k,\downarrow}^p r_{k,\downarrow},$$

$$\|(\mathbf{L}_k^{\uparrow})^p \mathbf{B}_{k+1}\|_{\mathrm{op}} \le \Lambda_{k,\uparrow}^p r_{k,\uparrow}.$$

Using and Lemma E.5, and then applying Cauchy–Schwarz jointly over all filter and bias blocks, gives

$$\begin{aligned}&\|\mathbf{G}_g^{(k)}(\theta) - \mathbf{G}_g^{(k)}(\theta')\|_F \\ &\le \left[\Xi_k^2(B_X, \eta_X) + \Xi_k^2(B_H, \eta_H) + n_k\right]^{1/2} \|\theta_g - \theta_g'\|_2.\end{aligned}$$

Taking the maximum over k proves the claim. □

Lemma E.9 (One-step parameter sensitivity)**.** *For every fixed $\mathcal{S} \in \mathfrak{S}$, every admissible input $\mathbf{X}$, and every pair $\theta, \theta' \in \Theta$,*

$$d_w(\Phi_\theta(\mathcal{S}, \mathbf{X}), \Phi_{\theta'}(\mathcal{S}, \mathbf{X})) \le L_\Phi \|\theta_{\mathrm{rec}} - \theta'_{\mathrm{rec}}\|_2,$$

where

$$L_\Phi := A_{\mathrm{rec}} \sqrt{(1+\lambda)^2 \left[(B_c c_F)^2 + \frac{17}{16}\right] + \frac{1}{16}}. \tag{56}$$

Proof. Since the state and input are fixed, Lemma E.8 gives

$$\|\mathbf{f} - \mathbf{f}'\|_{F,\max} \le c_F A_{\mathrm{rec}} \|\theta_\tau - \theta'_\tau\|_2,$$
$$\|\mathbf{i} - \mathbf{i}'\|_{F,\max} \le \frac{1}{4} A_{\mathrm{rec}} \|\theta_i - \theta'_i\|_2,$$
$$\|\widetilde{\mathbf{c}} - \widetilde{\mathbf{c}}'\|_{F,\max} \le A_{\mathrm{rec}} \|\theta_z - \theta'_z\|_2,$$

and

$$\|\mathbf{o} - \mathbf{o}'\|_{F,\max} \le \frac{1}{4} A_{\mathrm{rec}} \|\theta_o - \theta'_o\|_2.$$

Since the fixed previous cell state belongs to $\mathfrak{S}$,

$$\bar{e}_C^+ \le A_{\mathrm{rec}} \left[B_c c_F \|\theta_\tau - \theta'_\tau\|_2 + \frac{1}{4}\|\theta_i - \theta'_i\|_2 + \|\theta_z - \theta'_z\|_2 \right].$$

Therefore, by Cauchy–Schwarz,

$$\bar{e}_C^+ \le A_{\mathrm{rec}} \sqrt{(B_c c_F)^2 + \frac{17}{16}}\, \|\theta_{\{i,z,\tau\}} - \theta'_{\{i,z,\tau\}}\|_2.$$

For the hidden state,

$$\bar{e}_H^+ \le \bar{e}_C^+ + \frac{1}{4} A_{\mathrm{rec}} \|\theta_o - \theta'_o\|_2.$$

Hence

$$\begin{aligned} d_w^+ &= \bar{e}_H^+ + \lambda \bar{e}_C^+ \\ &\le (1+\lambda)\bar{e}_C^+ + \frac{1}{4} A_{\mathrm{rec}} \|\theta_o - \theta'_o\|_2. \end{aligned}$$

A final application of Cauchy–Schwarz over the four disjoint gate parameter blocks gives the stated L_Φ. □

E.4.5 Temporal sensitivity with shared parameters

Define

$$\Psi_T(q) := \sum_{j=0}^{T-1} q^j = \begin{cases} \dfrac{1-q^T}{1-q}, & q \ne 1, \\ T, & q = 1. \end{cases} \tag{57}$$

Lemma E.10 (Shared-parameter temporal sensitivity)**.** *Let two STORM models with recurrent parameters θ_{rec} and θ'_{rec} receive the same input trajectory and share $\mathcal{S}_0 = \mathbf{0}$. Then*

$$d_w(\mathcal{S}_T(\theta_{\mathrm{rec}}), \mathcal{S}_T(\theta'_{\mathrm{rec}})) \le L_\Phi \Psi_T(q_\lambda) \|\theta_{\mathrm{rec}} - \theta'_{\mathrm{rec}}\|_2.$$

Proof. By Lemma E.5, both trajectories remain in $\mathfrak{S}$. Let

$$e_t := d_w(\mathcal{S}_t(\theta_{\mathrm{rec}}), \mathcal{S}_t(\theta'_{\mathrm{rec}})).$$

Then $e_0 = 0$. Introduce

$$\widetilde{\mathcal{S}}_t := \Phi_{\theta_{\mathrm{rec}}}(\mathcal{S}_{t-1}(\theta'_{\mathrm{rec}}), \mathbf{X}_t).$$

The triangle inequality, followed by Lemmas E.6 and E.9, gives

$$e_t \le q_\lambda e_{t-1} + L_\Phi \|\theta_{\mathrm{rec}} - \theta'_{\mathrm{rec}}\|_2.$$

Since $e_0 = 0$,

$$e_T \le L_\Phi \sum_{j=0}^{T-1} q_\lambda^j \|\theta_{\mathrm{rec}} - \theta'_{\mathrm{rec}}\|_2.$$

□

The same parameter difference $\theta_{\mathrm{rec}} - \theta'_{\mathrm{rec}}$ appears at every step. Thus the proof preserves STORM's shared recurrent parameters and does not introduce independent parameters across time.

E.4.6 READOUT AND PREDICTOR SENSITIVITY

Write

$$\mathbf{Y}_T^{(k)} = \mathbf{h}_T^{(k)} \mathbf{W}_{\text{out}}^{(k)} + \mathbf{1}_{n_k} (\mathbf{b}_{\text{out}}^{(k)})^\top.$$

Since Θ is bounded, define

$$U_k := \sup_{\theta \in \Theta} \|\mathbf{W}_{\text{out}}^{(k)}\|_{\text{op}}, \qquad C_{\text{out}} := \left(\sum_{k=0}^{K} U_k^2 \right)^{1/2},$$

and

$$A_{\text{out}} := \max_k \sqrt{B_{H,k}^2 + n_k} = \max_k \sqrt{n_k (H+1)}.$$

Lemma E.11 (Predictor sensitivity)**.** *For the complete parameter vector $\theta = (\theta_{\text{rec}}, \theta_{\text{out}})$,*

$$\|f_\theta(\mathbf{X}_{1:T}) - f_{\theta'}(\mathbf{X}_{1:T})\|_{\text{stk}} \le L_T \|\theta - \theta'\|_2,$$

where

$$L_T := \sqrt{A_{\text{out}}^2 + \left[C_{\text{out}} L_\Phi \Psi_T(q_\lambda) \right]^2}. \tag{58}$$

Proof. Holding the readout parameters fixed,

$$\begin{aligned} &\|\Delta \mathbf{Y}_{\text{state}}\|_{\text{stk}}^2 \\ &= \sum_{k=0}^{K} \|(\mathbf{h}_T^{(k)} - \mathbf{h}_T^{(k)\prime}) \mathbf{W}_{\text{out}}^{(k)}\|_F^2 \\ &\le C_{\text{out}}^2 d_w(\mathcal{S}_T, \mathcal{S}'_T)^2. \end{aligned}$$

Therefore

$$\|\Delta \mathbf{Y}_{\text{state}}\|_{\text{stk}} \le C_{\text{out}} L_\Phi \Psi_T(q_\lambda) \|\theta_{\text{rec}} - \theta'_{\text{rec}}\|_2.$$

At a fixed hidden state, for each order,

$$\begin{aligned} &\|\mathbf{h}^{(k)} (\mathbf{W}_{\text{out}}^{(k)} - \mathbf{W}_{\text{out}}^{(k)\prime}) + \mathbf{1}_{n_k} (\mathbf{b}_{\text{out}}^{(k)} - \mathbf{b}_{\text{out}}^{(k)\prime})^\top\|_F \\ &\le \sqrt{B_{H,k}^2 + n_k} \left(\|\mathbf{W}_{\text{out}}^{(k)} - \mathbf{W}_{\text{out}}^{(k)\prime}\|_F^2 + \|\mathbf{b}_{\text{out}}^{(k)} - \mathbf{b}_{\text{out}}^{(k)\prime}\|_2^2 \right)^{1/2}. \end{aligned}$$

Summing the squared bounds over k and taking the square root gives

$$\|\Delta \mathbf{Y}_{\text{out}}\|_{\text{stk}} \le A_{\text{out}} \|\theta_{\text{out}} - \theta'_{\text{out}}\|_2.$$

Adding these two contributions gives

$$\|f_\theta(\mathbf{X}) - f_{\theta'}(\mathbf{X})\|_{\text{stk}} \le C_{\text{out}} L_\Phi \Psi_T(q_\lambda) \|\theta_{\text{rec}} - \theta'_{\text{rec}}\|_2 + A_{\text{out}} \|\theta_{\text{out}} - \theta'_{\text{out}}\|_2.$$

Cauchy–Schwarz over the two parameter blocks proves the result. This argument does not require Θ to be a product of separate recurrent and readout parameter classes. □

By E.8, the scalar loss class

$$\mathcal{G}_T := \{(\mathbf{X}, \mathbf{Y}^*) \mapsto \ell(f_\theta(\mathbf{X}), \mathbf{Y}^*) : \theta \in \Theta\}$$

satisfies

$$|g_\theta(z) - g_{\theta'}(z)| \le L_\ell L_T \|\theta - \theta'\|_2. \tag{59}$$

E.4.7 RADEMACHER COMPLEXITY OF THE LOSS CLASS

For a real-valued function class $\mathcal{G}$ and a sample $S = (z_1, \ldots, z_M)$, the empirical Rademacher complexity Mohri et al. (2018); Bartlett & Mendelson (2002); Shalev-Shwartz & Ben-David (2014) is

$$\widehat{\mathfrak{R}}_S(\mathcal{G}) := \frac{1}{M} \mathbb{E}_{\boldsymbol{\sigma}} \left[\sup_{g \in \mathcal{G}} \sum_{m=1}^{M} \sigma_m g(z_m) \right],$$

where $\sigma_1, \ldots, \sigma_M$ are independent Rademacher random variables taking values in $\{-1, +1\}$ with equal probability.

We apply this definition to the scalar loss class

$$\mathcal{G}_T := \{(\mathbf{X}, \mathbf{Y}^*) \mapsto \ell(f_\theta(\mathbf{X}), \mathbf{Y}^*) : \theta \in \Theta\}.$$

By E.8 and Lemma E.11,

$$|g_\theta(z) - g_{\theta'}(z)| \le L_\ell L_T \|\theta - \theta'\|_2. \tag{60}$$

We use this parameter sensitivity to bound the Rademacher complexity of $\mathcal{G}_T$. The following argument applies to any scalar function class satisfying such a Lipschitz bound and gives a dependence of order

$$L D_\Theta \sqrt{\frac{d}{M}}.$$

Lemma E.12 (Finite-class Rademacher bound)**.** *Let $\mathcal{V} \subset \mathbb{R}^M$ be a nonempty finite set satisfying $\|v\|_2 \le R$ for every $v \in \mathcal{V}$. Then*

$$\frac{1}{M}\mathbb{E}_{\boldsymbol{\sigma}} \max_{v \in \mathcal{V}} \langle \boldsymbol{\sigma}, v \rangle \le \frac{R}{M}\sqrt{2 \log |\mathcal{V}|}.$$

Proof. If $R = 0$, every $v \in \mathcal{V}$ is zero and the conclusion is immediate. If $|\mathcal{V}| = 1$, then

$$\mathbb{E}_{\boldsymbol{\sigma}} \langle \boldsymbol{\sigma}, v \rangle = 0,$$

and the conclusion is again immediate. Assume $R > 0$ and $|\mathcal{V}| \ge 2$. For $a > 0$,

$$\mathbb{E}_{\boldsymbol{\sigma}} e^{a\langle \boldsymbol{\sigma}, v \rangle} = \prod_{m=1}^{M} \cosh(a v_m) \le e^{a^2 R^2/2}.$$

Hence

$$\mathbb{E}_{\boldsymbol{\sigma}} \max_{v \in \mathcal{V}} \langle \boldsymbol{\sigma}, v \rangle \le \frac{\log |\mathcal{V}|}{a} + \frac{a R^2}{2}.$$

Choosing

$$a = \frac{\sqrt{2 \log |\mathcal{V}|}}{R}$$

gives the claim after division by M. □

Lemma E.13 (Lipschitz parameterization bound)**.** *Let $\Theta \subset \mathbb{R}^d$ be nonempty with finite Euclidean diameter D_Θ, and suppose*

$$\mathcal{G} = \{g_\theta : \theta \in \Theta\}$$

satisfies

$$|g_\theta(z) - g_{\theta'}(z)| \le L \|\theta - \theta'\|_2.$$

Then

$$\widehat{\mathfrak{R}}_S(\mathcal{G}) \le 15 L D_\Theta \sqrt{\frac{d}{M}}. \tag{61}$$

Proof. If $D_\Theta = 0$ or $L = 0$, the class contains a single function and its empirical Rademacher complexity is zero. Otherwise, fix $\theta_0 \in \Theta$.

For $j \ge 0$, set

$$r_j = D_\Theta 2^{-j},$$

so $r_0 = D_\Theta$. For each $j \ge 1$, choose a maximal subset $\mathcal{N}_j \subset \Theta$ whose distinct points have distance greater than r_j. Every point of Θ is within r_j of $\mathcal{N}_j$. To bound its size, place disjoint balls of radius $r_j/2$ around its points. They all lie in the ball centered at θ_0 with radius $D_\Theta + r_j/2$. Comparing volumes gives

$$|\mathcal{N}_j| \le \left(1 + \frac{2D_\Theta}{r_j}\right)^d = (1 + 2^{j+1})^d.$$

Set $\mathcal{N}_0 = \{\theta_0\}$. For each $\theta \in \Theta$, choose $\pi_j(\theta) \in \mathcal{N}_j$ satisfying

$$\|\theta - \pi_j(\theta)\|_2 \le r_j,$$

and set $\pi_0(\theta) = \theta_0$.

Subtracting g_{θ_0} leaves the empirical Rademacher complexity unchanged, because its signed sum has expectation zero. For any finite $J \geq 1$,

$$g_\theta - g_{\theta_0} = \sum_{j=1}^{J} \left[g_{\pi_j(\theta)} - g_{\pi_{j-1}(\theta)}\right] + \left[g_\theta - g_{\pi_J(\theta)}\right].$$

The final term contributes at most Lr_J to the normalized signed sum, because

$$\frac{1}{M}\left|\sum_{m=1}^{M} \sigma_m[g_\theta(z_m) - g_{\pi_J(\theta)}(z_m)]\right| \leq Lr_J.$$

For a level-j pair

$$u = \pi_j(\theta), \qquad v = \pi_{j-1}(\theta),$$

we have

$$\|u - v\|_2 \leq r_j + r_{j-1} = 3r_j.$$

Thus the corresponding evaluation vector has Euclidean norm at most

$$3Lr_j\sqrt{M}.$$

There are at most $|\mathcal{N}_j||\mathcal{N}_{j-1}|$ such vectors. Here we only consider pairs that arise from the same θ. For $j \geq 1$, the covering bounds imply

$$2\log\big(|\mathcal{N}_j||\mathcal{N}_{j-1}|\big) \leq 2d(2j+3)\log 2 \leq d(4j+6).$$

Taking the supremum of the sum to be at most the sum of the suprema, then applying Lemma E.12 to each level, gives

$$\widehat{\mathfrak{R}}_S(\mathcal{G}) \leq 3LD_\Theta\sqrt{\frac{d}{M}}\sum_{j=1}^{J} 2^{-j}\sqrt{4j+6} + LD_\Theta 2^{-J}.$$

Since

$$\sqrt{4j+6} \leq j+3$$

for $j \geq 1$,

$$\sum_{j=1}^{\infty} 2^{-j}\sqrt{4j+6} \leq \sum_{j=1}^{\infty} 2^{-j}(j+3) = 5.$$

Letting $J \to \infty$ proves

$$\widehat{\mathfrak{R}}_S(\mathcal{G}) \leq 15LD_\Theta\sqrt{\frac{d}{M}}.$$

□

Applying Lemma E.13 with $L = L_\ell L_T$ gives

$$\widehat{\mathfrak{R}}_S(\mathcal{G}_T) \leq 15L_\ell L_T D_\Theta\sqrt{\frac{d}{M}}. \tag{62}$$

E.4.8 Proof of Theorem 2

For completeness, Theorem 2 states that under **E.6– E.11**, for every $\delta \in (0,1)$, with probability at least $1-\delta$, uniformly over $\theta \in \Theta$,

$$R(f_\theta) \leq \widehat{R}_M(f_\theta) + CL_\ell D_\Theta L_T\sqrt{\frac{d}{M}} + B_\ell\sqrt{\frac{\log(1/\delta)}{2M}},$$

for a numerical constant $C > 0$. If there exists $\lambda > 0$ such that $q_\lambda < 1$, then $\Psi_T(q_\lambda) \leq (1-q_\lambda)^{-1}$ and the temporal factor in the bound is independent of T.

Proof. Define

$$\Gamma(S) := \sup_{g \in \mathcal{G}_T} \left[\mathbb{E}g(Z) - \frac{1}{M} \sum_{m=1}^{M} g(Z_m) \right].$$

Let $S' = (Z'_1, \ldots, Z'_M)$ be an independent sample from the same distribution. Replacing the expected loss by its expectation over S', and then moving the supremum inside that expectation, gives the first inequality below. Independently exchanging each pair (Z_m, Z'_m) does not change its joint distribution. Introducing random signs for these exchanges and separating the two sums gives

$$\begin{aligned} \mathbb{E}_S \Gamma(S) &\leq \mathbb{E}_{S,S'} \sup_{g \in \mathcal{G}_T} \frac{1}{M} \sum_{m=1}^{M} [g(Z'_m) - g(Z_m)] \\ &\leq 2\mathbb{E}_S[\widehat{\mathfrak{R}}_S(\mathcal{G}_T)]. \end{aligned}$$

Using equation 62,

$$\mathbb{E}_S \Gamma(S) \leq 30 L_\ell L_T D_\Theta \sqrt{\frac{d}{M}}.$$

Since every $g \in \mathcal{G}_T$ takes values in $[0, B_\ell]$, replacing one training pair changes $\Gamma(S)$ by at most B_ℓ/M. McDiarmid's inequality therefore gives, with probability at least $1 - \delta$,

$$\Gamma(S) \leq 30 L_\ell L_T D_\Theta \sqrt{\frac{d}{M}} + B_\ell \sqrt{\frac{\log(1/\delta)}{2M}}.$$

By the definition of $\Gamma(S)$, this holds uniformly over $\theta \in \Theta$. Thus Theorem 2 holds with the numerical constant

$$C = 30.$$

□

The factor 15 comes from the covering bound and the factor 2 from symmetrization. Neither depends on the architecture.

E.4.9 Interpretation

Remark E.4 (Dependence on sequence length)**.** *The factor $\Psi_T(q_\lambda)$ describes how the effect of a parameter perturbation can accumulate over T recurrent updates. In particular,*

$$\Psi_T(q_\lambda) = \begin{cases} \dfrac{1 - q_\lambda^T}{1 - q_\lambda}, & q_\lambda \neq 1, \\ T, & q_\lambda = 1. \end{cases}$$

Thus it remains bounded when $q_\lambda < 1$, grows linearly when $q_\lambda = 1$, and grows geometrically as an upper bound when $q_\lambda > 1$. This describes the sensitivity bound only; it does not imply the same growth in the predictions, forecasting error, or realized generalization gap.

When $q_\lambda < 1$,

$$L_T \leq \sqrt{A_{\text{out}}^2 + \left[\frac{C_{\text{out}} L_\Phi}{1 - q_\lambda} \right]^2},$$

which gives the sequence-length-independent case in Theorem 2.

F Extended Experimental Details

Predictions for STORM and LightSTORM are produced by a per-order linear decoder $\hat{\mathbf{X}}^{(k)}_{t+1:t+S} = \mathbf{h}^{(k)}_t \mathbf{W}^{(k)}_{\text{out}} + \mathbf{b}^{(k)}_{\text{out}}$, where $\mathbf{W}^{(k)}_{\text{out}} \in \mathbb{R}^{H \times S d_k}$, so that all S forecast steps are produced in one forward pass.

F.1 Dataset Details

PeMS-D4 and PeMS-D8. Nodes correspond to traffic sensors and carry traffic flow measurements. Edges represent oriented relations between connected sensors, and triangles represent three-sensor higher-order groups in the constructed simplicial complex. Following Chen et al. (2022b), we use PeMS-D4 and PeMS-D8 as traffic forecasting benchmarks. For each dataset, we construct a fixed clique complex from the supplied sensor graph: graph links form the edges and graph 3-cliques form the triangles. Since traffic flow is observed only at nodes, we lift it to higher orders by unsigned incidence averaging,

$$\mathbf{X}_t^{(1)} = \frac{1}{2}|\mathbf{B}_1|^\top \mathbf{X}_t^{(0)}, \qquad \mathbf{X}_t^{(2)} = \frac{1}{3}|\mathbf{B}_2|^\top \mathbf{X}_t^{(1)}.$$

Thus an edge carries the mean flow of its two endpoints and a triangle carries the mean flow of its three vertices. Orientations are retained in the boundary operators used by the model but do not affect the lifted signals.

Water (EPANET). Nodes correspond to junctions and carry pressure signals, while edges correspond to pipes and carry flow signals. These node and edge signals are provided directly by the hydraulic simulation U.S. Environmental Protection Agency (2000), requiring no signal lifting. The dataset has no triangles ($N_2 = 0$).

Feature normalization. Features for each simplex order are normalized independently using training set statistics (zero mean, unit variance), computed separately for nodes, edges, and triangles.

Operator normalization. For training, each nonzero lower and upper Hodge Laplacian is divided by its spectral norm, $\widetilde{\mathbf{L}}_k^a = \mathbf{L}_k^a / \|\mathbf{L}_k^a\|_{\mathrm{op}}$, for $a \in \{\downarrow, \uparrow\}$. Boundary matrices are left unnormalized.

Dataset sizes and temporal subsets. PeMS-D4 (T=16,992) covers traffic flow readings collected every 5 minutes over January and February 2018 ($59 \times 24 \times 12 = 16{,}992$ timesteps) and is used in full. PeMS-D8 (T=17,856) covers July and August 2016 ($62 \times 24 \times 12 = 17{,}856$ timesteps) and is used in full. Water (T=4,480) is used in full. All datasets are split chronologically into training (70%), validation (15%), and testing (15%) sets relative to their respective total length T.

F.1.1 Scalability Discussion

The datasets used in our evaluation are of moderate scale, with the largest simplicial complex containing $N = 676$ simplices (PeMS-D4). As analyzed in Appendix D, for fixed polynomial degree and bounded simplex degree, STORM and LightSTORM have sequence costs that scale linearly in both T and N. Under these conditions, the total number of nonzeros in the required operator powers is also linear in N. In comparison, self-attention applied independently at each simplex order has a spatial term proportional to $\sum_k n_k^2$. Thus, the proposed architectures have more favorable asymptotic spatial scaling, although our experiments do not directly evaluate the large-N regime.

The PeMS datasets are standard benchmarks in the spatiotemporal forecasting literature Li et al. (2018); Wu et al. (2019); Liu et al. (2024) and enable comparison with prior work. Water, while smaller ($N = 76$), provides a physically grounded setting in which edge signals arise from the underlying hydraulic dynamics rather than signal lifting. The evaluated datasets therefore cover different moderate-size complexes, from $N = 76$ to $N = 676$.

F.2 Joint Multi-Order Forecasting Setup

F.2.1 Baseline Selection Rationale

Baselines were selected to systematically assess whether topological structure provides a benefit for dynamical systems forecasting, spanning a spectrum from no structural prior to full simplicial topology. The selection follows four categories of increasing structural richness.

Topology-unaware sequence models (LSTM, xLSTM, Transformer, iTransformer) represent strong temporal learners with no spatial inductive bias, that treat the multivariate signal as a generic vector.

Graph-based models (GCN, GCN-xLSTM) add pairwise spatial structure via graph convolution but remain restricted to node-level signals and pairwise interactions, isolating the contribution of higher-order topology over graph-only structure.

Static simplicial models (SC-VAR, SCRNN) incorporate full simplicial topology but do not implement topology-aware recurrent gating. SC-VAR isolates the contribution of static topological filtering without gated recurrent memory, while SCRNN follows a two-step design: it first processes topological signals, then flattens them and applies a recurrent backbone, rather than processing topology and temporal memory jointly.

Our models (STORM, LightSTORM) combine full simplicial topology with gated nonlinear recurrence, representing the full proposed design.

This progression allows us to attribute performance differences to specific architectural components: topology vs. no topology, higher-order vs. graph-only, gated memory vs. static filtering, separate vs. joint processing.

F.2.2 Hyperparameter Configuration

STORM and LightSTORM. Both models use one linear topological filter per gate branch: one branch for the current input and one branch for the previous hidden state. The filter order is swept over $P \in \{1, 2, 3\}$ for STORM and $J \in \{1, 2, 3\}$ for LightSTORM, with hidden dimension $H \in \{16, 32\}$. Three cross-order injection modes are evaluated for STORM/LightSTORM: *input*, *hidden*, and *both*. The best configuration per dataset is selected by validation NMSE.

Baselines. Transformer and iTransformer use 2 layers, 4 attention heads, feedforward dimension 32, and dropout 0.1. The main tables report iTransformer with $H{=}32$; Appendix F.4 additionally reports a larger $H{=}64$ variant as a capacity control. The LSTM baseline uses 2 temporal layers with hidden dimension swept over $\{16, 32, 48\}$, and the reported configuration uses $H{=}48$. The xLSTM baseline is a topology-free, per-order implementation of the stabilized scalar-memory sLSTM update Beck et al. (2024). It uses one recurrent cell per simplex order with hidden dimension $H{=}48$, followed by the same direct multi-step readout used by the other recurrent models. GCN-xLSTM uses 2 GCN layers followed by 2 temporal layers. The GCN baseline uses 4 graph-convolution layers with hidden dimension $H{=}64$. SC-VAR and SCRNN are selected by validation NMSE using the same validation protocol as the other baselines.

Table 9 summarizes all hyperparameters.

Table 9: Hyperparameter configuration. The learning rate 10^{-3} was selected by sweep over $\{10^{-3}, 3 \times 10^{-4}, 5 \times 10^{-4}, 10^{-4}\}$ applied uniformly to all models.

Hyperparameter	**Value**
Learning rate	10^{-3}
Batch size	32
Optimizer	Adam ($\beta_1{=}0.9$, $\beta_2{=}0.999$, weight decay 0)
Max epochs	300
Learning-rate warmup	Linear, 10 epochs, from 10^{-5} to 10^{-3}
Validation interval	5 epochs
Early stopping patience	15 validation checks (validation MSE)
Gradient clipping	0.5 (global norm)
Filter order P (STORM) / J (LightSTORM)	$\{1, 2, 3\}$
Hidden dimension H	$\{16, 32\}$ (ours); $\{16, 32, 48\}$ (baselines)
Injection mode (STORM/LightSTORM)	{input, hidden, both}
Layers (Transformer, iTransformer)	2
Attention heads	4
Independent runs	5 (mean $\pm$ std reported)

F.2.3 Training Loss

For joint multi-order forecasting, we prevent simplex orders with more elements from dominating gradient updates over orders with fewer elements (e.g., nodes over triangles) by using a balanced

forecasting loss. It computes the mean squared error independently per order and averages the results. For each order

$$\mathcal{L}^{(k)} = \frac{1}{|\mathcal{B}|\, S\, n_k\, d_k} \sum_{m\in\mathcal{B}} \sum_{s=1}^{S} \sum_{i=1}^{n_k} \sum_{j=1}^{d_k} \left(\hat{Y}_{s,i,j}^{(k,m)} - Y_{s,i,j}^{(k,m)} \right)^2 . \tag{63}$$

A weighted variant $\mathcal{L}_{\text{balanced}} = \sum_k w_k \mathcal{L}^{(k)} / \sum_k w_k$ is supported but was not used in the joint multi-order experiments (equal weights across orders). The node-only benchmark instead uses the objective in equation 67.

F.2.4 Evaluation Metrics

For the joint multi-order experiments, we report final-step forecasting metrics at prediction step S. This evaluation focuses on the error at the target horizon rather than averaging over intermediate steps. Unless otherwise stated, metrics are computed by pooling squared errors over all evaluated simplex orders. For per-order diagnostics, the same definitions are applied separately to each simplex order. Evaluation uses direct multi-step prediction with no autoregressive rollout.

Mean Squared Error (MSE). For simplex order k, the final-step MSE at forecasting step S is

$$\text{MSE}_S^{(k)} = \frac{1}{|\mathcal{T}|\, n_k\, d_k} \sum_{t\in\mathcal{T}} \sum_{i=1}^{n_k} \sum_{j=1}^{d_k} \left(\hat{Y}_{t,i,S,j}^{(k)} - Y_{t,i,S,j}^{(k)} \right)^2 . \tag{64}$$

MSE measures the average squared prediction error at the target forecasting step. Lower is better.

Normalized Mean Squared Error (NMSE). For simplex order k, the final-step NMSE is

$$\text{NMSE}_S^{(k)} = \frac{\sum_{t\in\mathcal{T}} \sum_{i=1}^{n_k} \sum_{j=1}^{d_k} \left(\hat{Y}_{t,i,S,j}^{(k)} - Y_{t,i,S,j}^{(k)} \right)^2}{\sum_{t\in\mathcal{T}} \sum_{i=1}^{n_k} \sum_{j=1}^{d_k} \left(Y_{t,i,S,j}^{(k)} - \bar{Y}_S^{(k)} \right)^2}, \tag{65}$$

where $\bar{Y}_S^{(k)}$ is the mean target value at order k and forecasting step S, computed over the evaluation set.

When reporting a single pooled NMSE across all evaluated simplex orders, we pool numerator and denominator before taking the ratio:

$$\text{NMSE}_S = \frac{\sum_k \sum_{t\in\mathcal{T}} \sum_{i=1}^{n_k} \sum_{j=1}^{d_k} \left(\hat{Y}_{t,i,S,j}^{(k)} - Y_{t,i,S,j}^{(k)} \right)^2}{\sum_k \sum_{t\in\mathcal{T}} \sum_{i=1}^{n_k} \sum_{j=1}^{d_k} \left(Y_{t,i,S,j}^{(k)} - \bar{Y}_S^{(k)} \right)^2}. \tag{66}$$

NMSE normalizes prediction error by target variance, making it scale-free and comparable across datasets and simplex orders. A value of 1 corresponds to the error of a constant mean predictor, while values below 1 indicate that the model captures temporal structure. Lower is better.

F.2.5 Parameter Count

Table 10 reports representative parameter counts at $S = 10$, including the forecasting head. For fixed numbers of simplex orders, hidden dimensions, feature dimensions, and filter degrees, the trainable filter parameters of STORM and LightSTORM do not depend on the number of simplices: increasing the number of nodes, edges, or triangles changes the computational and memory cost, but not the number of filter weights.

LightSTORM further reduces the parameter count by sharing its Dirac-polynomial filters across simplex orders. At hidden dimension $d = 32$ with both-branch injection, this reduces the parameter count from 192.4K to 59.0K on PeMS and from 128.3K to 58.6K on Water, corresponding to reductions of approximately $3.3\times$ and $2.2\times$, respectively.

Table 10: Representative parameter counts at $S = 10$, including the forecasting head. d denotes the hidden dimension.

Model	PeMS	Water
iTrans., $d = 32$	53.3K	35.5K
iTrans., $d = 64$	204.9K	136.6K
Transformer	14.0K	26.7K
LSTM, $d = 48$	87.4K	58.3K
xLSTM, $d = 48$	30.3K	20.2K
GCN-xLSTM	40.0K	26.6K
SC-VAR	2.6K	1.5K
SCRNN	11.3K	12.7K
STORM in, $d = 32$	94.1K	62.7K
STORM hid, $d = 32$	139.4K	126.2K
STORM both, $d = 32$	192.4K	128.3K
STORM both, $d = 16$	50.2K	33.4K
L-STORM in, $d = 32$	34.4K	34.0K
L-STORM hid, $d = 32$	26.2K	34.0K
L-STORM both, $d = 32$	59.0K	58.6K
L-STORM both, $d = 16$	15.2K	15.0K

F.3 Node-Only Forecasting Setup

We use the same PeMS-D4 and PeMS-D8 complexes, preprocessing, and data splits as in the main experiments, but restrict the prediction target and loss to node signals. All models receive the same observed node history. STORM and LightSTORM retain their multi-order internal representation, with higher-order inputs constructed from the observed node history as described in Section 5. We evaluate horizons $S \in \{5, 10, 20, 40, 60, 80\}$. Each model directly predicts the full forecast horizon. Training uses stride-32 windows, while validation and test use stride-one windows. Model selection is based on validation node error. Unlike joint-forecasting experiments, here we use a normalized node MSE over every forecast step, node, and feature:

$$\mathcal{L}_{\text{node}} = \frac{1}{|\mathcal{B}| S n_0 d_0} \sum_{m \in \mathcal{B}} \sum_{s=1}^{S} \sum_{i=1}^{n_0} \sum_{j=1}^{d_0} \big(\widehat{Y}_{sij}^{(0,m)} - Y_{sij}^{(0,m)}\big)^2. \tag{67}$$

Higher-order states remain active, but only node predictions are used for training.

Hyperparameter selection. For STORM and LightSTORM, we use the same hyperparameter search space as in the joint forecasting experiments: hidden width $H \in \{16, 32\}$, polynomial order $P \in \{1, 2, 3\}$, and topology injection in the input, hidden, or both branches. Configurations are selected separately for each model, dataset, and forecast horizon using validation node MSE. The only difference from the joint forecasting setting is the node-only objective in Eq. equation 67.

Baseline configurations. For the node-forecasting baselines, we use the official implementations where available and follow their reference architectures; DCRNN uses the BasicTS PyTorch reproduction. For each baseline, we evaluate four training configurations obtained by combining the reference loss or normalized node MSE with the reference learning rate or 0.3 times that rate. Configurations are selected using validation node MSE under the same forecasting protocol, after which the selected model is trained over five seeds and test performance is averaged across runs. The main architecture and reference training settings are summarized in Table 11.

F.4 Capacity Control for iTransformer

Since full STORM has more parameters than several sequence baselines, we also evaluate larger iTransformer variants as a capacity control. In the main tables, iTransformer is reported with H=32, matching the hidden width used for the main STORM and LightSTORM configurations. We additionally evaluate iTransformer with H=64, which substantially increases its parameter count. Table 12 shows that increasing iTransformer capacity improves performance only marginally and does

Table 11: Architecture and reference training settings for the node-forecasting baselines. WD denotes weight decay. For each model, the hyperparameter search also considers 0.3 times the listed learning rate.

Model	Architecture	LR	WD	Batch
Graph WaveNet	Residual/dilation width 32, skip width 256, end width 512, dropout 0.3; fixed and adaptive graph supports	10^{-3}	10^{-4}	64
STID	3 layers; feature and node-embedding widths 32	2×10^{-3}	10^{-4}	64
STAEformer	3 layers, 4 heads, FFN width 256, adaptive embedding width 80, dropout 0.1	10^{-3}	5×10^{-4} / 1.5×10^{-3}	16
DCRNN	2 recurrent layers of width 64, diffusion order 2; scheduled-sampling decay 2000	3×10^{-3}	0	64

not close the long-horizon gap on the traffic datasets. On PeMS-D8, iTransformer with $H{=}64$ reaches final-step NMSE 0.2374 at $S{=}40$ and 0.4581 at $S{=}80$, compared with 0.1592 and 0.2310 for the best STORM variants. On PeMS-D4, the larger iTransformer reaches 0.2851 at $S{=}40$ and 0.5343 at $S{=}80$, compared with 0.2247 and 0.3164 for STORM. Thus, the traffic gains are not explained by simply increasing the parameter budget of a topology-unaware sequence model. On Water, the larger iTransformer is more competitive, consistent with the smaller gains observed on the node-edge dataset.

Table 12: Capacity control for iTransformer. Terminal-step pooled NMSE (lower is better) for $H{=}32$ and $H{=}64$. Entries report mean and standard deviation over five seeds. Parameter counts are total parameters at the corresponding horizon.

Dataset	Model	$S{=}5$	$S{=}10$	$S{=}20$	$S{=}40$	$S{=}60$	$S{=}80$
PeMS-D8	iTransformer, $H{=}32$	0.0388 ± 0.0001	0.0604 ± 0.0005	0.1248 ± 0.0034	0.2451 ± 0.0049	0.3439 ± 0.0055	0.4656 ± 0.0168
	iTransformer, $H{=}64$	0.0390 ± 0.0003	0.0592 ± 0.0005	0.1210 ± 0.0012	0.2374 ± 0.0045	0.3485 ± 0.0142	0.4581 ± 0.0104
	Params, $H{=}32$	52.8K	53.3K	54.3K	56.3K	58.3K	60.2K
	Params, $H{=}64$	203.9K	204.9K	206.8K	210.7K	214.6K	218.5K
PeMS-D4	iTransformer, $H{=}32$	0.0494 ± 0.0001	0.0715 ± 0.0005	0.1444 ± 0.0011	0.2933 ± 0.0048	0.3991 ± 0.0086	0.5274 ± 0.0090
	iTransformer, $H{=}64$	0.0490 ± 0.0006	0.0710 ± 0.0002	0.1408 ± 0.0021	0.2851 ± 0.0047	0.3991 ± 0.0073	0.5343 ± 0.0046
	Params, $H{=}32$	52.8K	53.3K	54.3K	56.3K	58.3K	60.2K
	Params, $H{=}64$	203.9K	204.9K	206.8K	210.7K	214.6K	218.5K
Water	iTransformer, $H{=}32$	0.0683 ± 0.0019	0.2118 ± 0.0067	0.5455 ± 0.0076	0.6249 ± 0.0085	0.6315 ± 0.0030	0.7180 ± 0.0093
	iTransformer, $H{=}64$	0.0647 ± 0.0011	0.2119 ± 0.0200	0.5263 ± 0.0126	0.5893 ± 0.0076	0.6079 ± 0.0072	0.6926 ± 0.0076
	Params, $H{=}32$	35.2K	35.5K	36.2K	37.5K	38.8K	40.2K
	Params, $H{=}64$	136.0K	136.6K	137.9K	140.5K	143.1K	145.7K

F.5 Dataset Topology Analysis

Tables 13 and 14 summarize the topological structure of the three datasets and the nonzero entries of their operators. PeMS-D8 has fewer nodes than PeMS-D4 but more triangles (80 vs. 29) and denser upper-order structure, as reflected by $\mathrm{nnz}(B_2)$, $\mathrm{nnz}(L_1^{\mathrm{upr}})$, and $\mathrm{nnz}(L_2)$. PeMS-D4 also contains nontrivial edge–triangle interactions, whereas Water has no triangles ($N_2{=}0$) and therefore contains only node–edge coupling.

Table 13: Topological structure and signal statistics. Signal statistics computed over the first 100 timesteps. "Near-zero curl %" denotes the fraction of triangle signal values with $|x| < 0.01$. For PeMS, triangle signals are node averages rather than boundary curls; the reported percentage therefore refers to near-zero triangle signal values.

Dataset	N_0	N_1	N_2	$\mathrm{nnz}(L_1^{\mathrm{lwr}})$+$\mathrm{nnz}(L_1^{\mathrm{upr}})$	$\mathrm{nnz}(L_2)$	Edge Std	Tri Std	Near-zero curl %	Near-zero grad %
PeMS-D4	307	340	29	1,724	51	0.426	0.363	0.03%	0.21%
PeMS-D8	170	274	80	2,678	314	0.986	1.029	0.47%	0.42%
Water	36	40	0	262	0	0.927	—	—	0.82%

Table 14: Nonzero entry counts for all topological operators.

Dataset	$\text{nnz}(B_1)$	$\text{nnz}(B_2)$	$\text{nnz}(L_0)$	$\text{nnz}(L_1^{\text{lwr}})$	$\text{nnz}(L_1^{\text{upr}})$	$\text{nnz}(L_2)$
PeMS-D4	680	87	987	1,474	250	51
PeMS-D8	548	240	718	2,044	634	314
Water	80	0	88	262	0	0

F.6 Hardware and Software

All experiments are conducted on a SLURM-managed HPC cluster. Standard experiments ($S \leq 20$) use NVIDIA V100 (16GB) or RTX 6000 (24GB) GPUs. Long-horizon experiments ($S \geq 40$) use NVIDIA A40 (48GB) GPUs. The software stack uses PyTorch Paszke et al. (2019).

G Further Experimental Results

G.1 Joint Multi-Order Forecasting

This section reports the complete forecasting tables for the joint multi-order forecasting, including the Transformer and GCN baselines omitted from the main text for space. The complete results are shown in Tables 15, 16, and 17. All other settings match the corresponding experiments in Section 5.

Table 15: NMSE (lower is better) at forecast step S on PeMS-D8 (mean $\pm$ std, 5 seeds). **Bold**: best overall. Underline: second best. Last row: blue/red = gain/loss of our best vs. best baseline (%).

Model	S=5	S=10	S=20	S=40	S=60	S=80
STORM (both)	0.0404 ± 0.0005	**0.0541** ± 0.0007	0.0963 ± 0.0004	0.1596 ± 0.0041	0.1959 ± 0.0033	**0.2310** ± 0.0059
STORM (hidden)	0.0402 ± 0.0005	0.0560 ± 0.0004	**0.0953** ± 0.0019	**0.1592** ± 0.0093	0.2008 ± 0.0065	0.2370 ± 0.0125
LightSTORM (both)	0.0403 ± 0.0003	0.0558 ± 0.0006	0.0982 ± 0.0037	0.1652 ± 0.0037	**0.1951** ± 0.0105	0.2462 ± 0.0067
LightSTORM (hidden)	0.0403 ± 0.0004	0.0552 ± 0.0008	0.0983 ± 0.0029	0.1624 ± 0.0055	0.2063 ± 0.0090	0.2511 ± 0.0177
iTransformer	**0.0388** ± 0.0001	0.0604 ± 0.0005	0.1248 ± 0.0038	0.2451 ± 0.0055	0.3439 ± 0.0062	0.4656 ± 0.0188
xLSTM	0.0403 ± 0.0003	0.0625 ± 0.0006	0.1359 ± 0.0022	0.3593 ± 0.0016	0.5687 ± 0.0065	0.7092 ± 0.0009
Transformer	0.0426 ± 0.0007	0.0653 ± 0.0012	0.1427 ± 0.0033	0.3726 ± 0.0087	0.5720 ± 0.0033	0.6879 ± 0.0061
LSTM	0.0392 ± 0.0004	0.0622 ± 0.0009	0.1373 ± 0.0007	0.3616 ± 0.0085	0.5640 ± 0.0064	0.6958 ± 0.0066
GCN	0.1051 ± 0.0022	0.1495 ± 0.0026	0.3006 ± 0.0025	0.6032 ± 0.0021	0.7594 ± 0.0018	0.8419 ± 0.0033
GCN-xLSTM	0.0404 ± 0.0002	0.0622 ± 0.0006	0.1333 ± 0.0029	0.2909 ± 0.0064	0.4427 ± 0.0053	0.5623 ± 0.0092
SC-VAR	0.0508 ± 0.0002	0.0966 ± 0.0007	0.2407 ± 0.0003	0.5675 ± 0.0004	0.7904 ± 0.0001	0.8934 ± 0.0007
SCRNN	0.8706 ± 0.0004	0.8726 ± 0.0003	0.8753 ± 0.0006	0.8817 ± 0.0060	0.8820 ± 0.0033	0.8856 ± 0.0020
Gain vs. best baseline	−3.6%	+10.4%	+23.6%	+35.0%	+43.3%	+50.4%

Table 16: NMSE (lower is better) at forecast step S on PeMS-D4 (mean $\pm$ std, 5 seeds). **Bold**: best overall. Underline: second best. Last row: blue/red = gain/loss of our best vs. best baseline (%).

Model	S=5	S=10	S=20	S=40	S=60	S=80
STORM (both)	0.0495 ± 0.0002	0.0725 ± 0.0004	0.1214 ± 0.0013	**0.2247** ± 0.0033	**0.2686** ± 0.0031	**0.3164** ± 0.0017
STORM (hidden)	0.0494 ± 0.0003	0.0728 ± 0.0008	**0.1206** ± 0.0020	0.2259 ± 0.0006	0.2759 ± 0.0077	0.3287 ± 0.0179
LightSTORM (both)	0.0494 ± 0.0004	0.0735 ± 0.0005	0.1244 ± 0.0008	0.2297 ± 0.0051	0.2886 ± 0.0051	0.3458 ± 0.0061
LightSTORM (hidden)	**0.0491** ± 0.0002	0.0729 ± 0.0009	0.1253 ± 0.0023	0.2449 ± 0.0069	0.3093 ± 0.0133	0.3626 ± 0.0074
iTransformer	0.0494 ± 0.0001	**0.0715** ± 0.0005	0.1444 ± 0.0012	0.2933 ± 0.0054	0.3991 ± 0.0097	0.5274 ± 0.0100
xLSTM	0.0498 ± 0.0002	0.0788 ± 0.0005	0.1676 ± 0.0014	0.4170 ± 0.0015	0.6550 ± 0.0042	0.7567 ± 0.0060
Transformer	0.0516 ± 0.0002	0.0829 ± 0.0003	0.1740 ± 0.0029	0.4298 ± 0.0039	0.6640 ± 0.0065	0.7605 ± 0.0018
LSTM	0.0499 ± 0.0002	0.0789 ± 0.0004	0.1690 ± 0.0005	0.4202 ± 0.0036	0.6500 ± 0.0052	0.7589 ± 0.0044
GCN	0.0999 ± 0.0026	0.1564 ± 0.0017	0.3116 ± 0.0007	0.6327 ± 0.0003	0.8214 ± 0.0011	0.9063 ± 0.0003
GCN-xLSTM	0.0495 ± 0.0004	0.0779 ± 0.0004	0.1573 ± 0.0012	0.3641 ± 0.0022	0.5420 ± 0.0043	0.6453 ± 0.0077
SC-VAR	0.0623 ± 0.0004	0.1173 ± 0.0009	0.2717 ± 0.0005	0.6046 ± 0.0004	0.8303 ± 0.0002	0.9339 ± 0.0004
SCRNN	0.8036 ± 0.0002	0.8057 ± 0.0012	0.8137 ± 0.0010	0.8187 ± 0.0023	0.8151 ± 0.0013	0.8257 ± 0.0011
Gain vs. best baseline	+0.6%	−1.4%	+16.5%	+23.4%	+32.7%	+40.0%

G.1.1 STORM and LightSTORM vs. sequence models

Figure 4 compares STORM and LightSTORM with topology-unaware sequence models: LSTM, xLSTM, Transformer, and iTransformer. The comparison shows that the long-horizon gains are not

Table 17: NMSE (lower is better) at forecast step S on Water (mean $\pm$ std, 5 seeds). **Bold**: best overall. Underline: second best. Last row: blue/red = gain/loss of our best vs. best baseline (%).

Model	S=5	S=10	S=20	S=40	S=60	S=80
STORM (both)	0.0679 $\pm$ 0.0010	0.2772 $\pm$ 0.0035	0.3519 $\pm$ 0.0144	0.5517 $\pm$ 0.0143	**0.6135** $\pm$ 0.0125	**0.6417** $\pm$ 0.0113
STORM (hidden)	**0.0655** $\pm$ 0.0027	0.2720 $\pm$ 0.0098	**0.3509** $\pm$ 0.0122	**0.5471** $\pm$ 0.0176	0.6222 $\pm$ 0.0123	0.6548 $\pm$ 0.0105
LightSTORM (both)	0.0697 $\pm$ 0.0010	0.3016 $\pm$ 0.0061	0.4143 $\pm$ 0.0175	0.5495 $\pm$ 0.0103	0.6379 $\pm$ 0.0097	0.6884 $\pm$ 0.0144
LightSTORM (hidden)	0.0696 $\pm$ 0.0020	0.3054 $\pm$ 0.0053	0.4144 $\pm$ 0.0227	0.5604 $\pm$ 0.0077	0.6321 $\pm$ 0.0163	0.6768 $\pm$ 0.0055
iTransformer	0.0683 $\pm$ 0.0021	**0.2118** $\pm$ 0.0075	0.5455 $\pm$ 0.0085	0.6249 $\pm$ 0.0095	0.6315 $\pm$ 0.0033	0.7180 $\pm$ 0.0104
xLSTM	0.0745 $\pm$ 0.0018	0.2809 $\pm$ 0.0088	0.5884 $\pm$ 0.0123	0.6303 $\pm$ 0.0060	0.7035 $\pm$ 0.0098	0.7687 $\pm$ 0.0046
Transformer	0.1012 $\pm$ 0.0010	0.3400 $\pm$ 0.0230	0.5490 $\pm$ 0.0029	0.6091 $\pm$ 0.0054	0.6654 $\pm$ 0.0052	0.7073 $\pm$ 0.0062
LSTM	0.0658 $\pm$ 0.0039	0.2380 $\pm$ 0.0083	0.5576 $\pm$ 0.0081	0.5984 $\pm$ 0.0061	0.6469 $\pm$ 0.0065	0.7090 $\pm$ 0.0088
GCN	0.3428 $\pm$ 0.0019	0.6259 $\pm$ 0.0014	0.8590 $\pm$ 0.0013	0.9400 $\pm$ 0.0023	0.7277 $\pm$ 0.0009	0.9274 $\pm$ 0.0012
GCN-xLSTM	0.0739 $\pm$ 0.0041	0.3252 $\pm$ 0.0207	0.5305 $\pm$ 0.0158	0.5990 $\pm$ 0.0101	0.6512 $\pm$ 0.0109	0.7182 $\pm$ 0.0070
SC-VAR	0.2353 $\pm$ 0.0109	0.5703 $\pm$ 0.0077	0.8588 $\pm$ 0.0056	0.8928 $\pm$ 0.0035	0.7244 $\pm$ 0.0004	0.9276 $\pm$ 0.0088
SCRNN	0.6467 $\pm$ 0.0035	0.6617 $\pm$ 0.0047	0.6577 $\pm$ 0.0074	0.7245 $\pm$ 0.0249	0.7266 $\pm$ 0.0308	0.7644 $\pm$ 0.0297
Gain vs. best baseline	+0.5%	−28.4%	+33.9%	+8.6%	+2.9%	+9.3%

due to a single weak baseline. On the traffic datasets, iTransformer is the strongest sequence baseline at short forecasting steps, but its error grows faster as the forecasting step increases. At S=80, iTransformer reaches final-step NMSE 0.466 on PeMS-D8 and 0.527 on PeMS-D4, while STORM remains below 0.32 on both datasets. LightSTORM follows the same trend with a smaller filter parameterization, remaining close to STORM on PeMS-D8 and ahead of the sequence baselines at medium and long steps on both traffic datasets. On Water, sequence baselines are competitive at short steps, especially around S=10, but STORM becomes more accurate from medium forecasting steps onward.

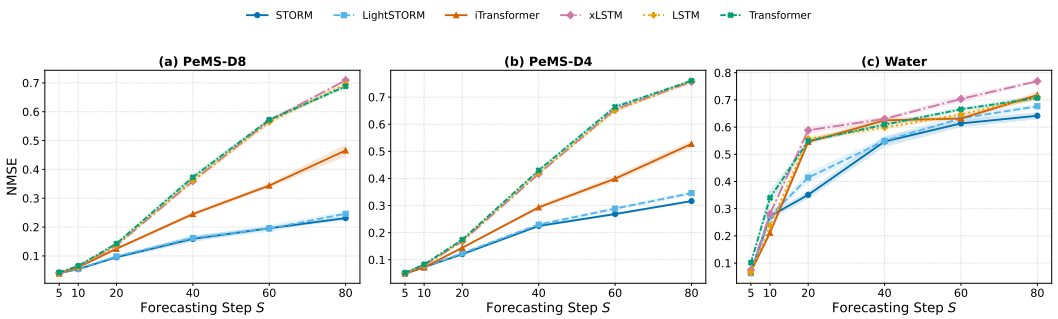


Figure 4: Comparison with topology-unaware sequence baselines. NMSE is reported at forecasting step S with ± 1 std over 5 seeds. Lower is better. For STORM and LightSTORM, each point reports the best evaluated injection variant at that forecasting step. On the traffic datasets, the sequence baselines are competitive at short steps but degrade more rapidly at medium and long steps. On Water, sequence baselines remain strong at short steps, while STORM is more accurate from medium steps onward.

G.1.2 STORM AND LIGHTSTORM VS. SPATIALLY-AWARE MODELS

Figure 5 compares STORM and LightSTORM with baselines that use graph or simplicial structure: GCN, GCN-xLSTM, SC-VAR, and SCRNN. Among these models, GCN-xLSTM is generally the strongest spatial baseline, showing that combining graph structure with recurrence is beneficial. However, its node-centric pairwise representation does not close the gap to STORM at medium and long forecasting steps on the traffic datasets. The simplicial baselines SC-VAR and SCRNN also remain substantially less accurate in this setting, suggesting that using simplicial structure alone is not sufficient without a gated recurrent mechanism that updates memory through multi-order topology. On Water, the gap is smaller, consistent with the absence of triangles, but STORM remains competitive with or better than the strongest spatial baselines from medium steps onward.

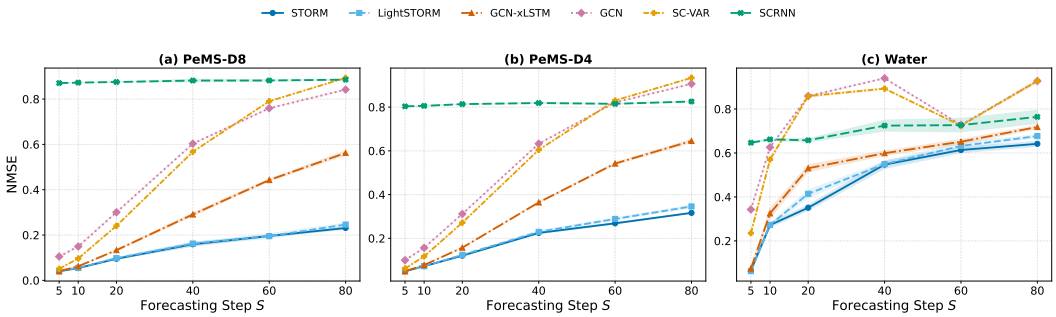


Figure 5: Comparison with graph-based and simplicial baselines. NMSE is reported at forecasting step S with ± 1 std over 5 seeds. Lower is better. For STORM and LightSTORM, each point reports the best evaluated injection variant at that forecasting step. GCN-xLSTM is the strongest spatially-aware baseline in most cases, but it does not close the medium- and long-step gap on the traffic datasets. The results support combining gated recurrence with multi-order topological filtering rather than using spatial structure or recurrent memory in isolation.

G.2 Node-Only Forecasting Results

Tables 18 and 19 report the complete node-forecasting results. Final-step NMSE evaluates the prediction at the furthest forecast step S, while all-step NMSE aggregates the error over the full predicted sequence. All results are averaged over five runs.

Table 18: PeMS-D4 node forecasting. Final-step and all-step NMSE (mean $\pm$ standard deviation over five runs; lower is better).

Model	$S = 5$	$S = 10$	$S = 20$	$S = 40$	$S = 60$	$S = 80$
Final-step NMSE						
STORM	0.0545 ± 0.0005	0.0783 ± 0.0012	0.1258 ± 0.0016	0.2196 ± 0.0022	0.2568 ± 0.0072	0.2859 ± 0.0058
LightSTORM	0.0553 ± 0.0002	0.0776 ± 0.0009	0.1223 ± 0.0022	0.2143 ± 0.0030	0.2495 ± 0.0072	0.2885 ± 0.0151
Graph WaveNet	0.0567 ± 0.0005	0.0836 ± 0.0031	0.1494 ± 0.0055	0.2959 ± 0.0127	0.3982 ± 0.0325	0.5130 ± 0.0271
STID	0.0557 ± 0.0002	0.0837 ± 0.0035	0.1427 ± 0.0024	0.2700 ± 0.0036	0.3538 ± 0.0043	0.3996 ± 0.0064
STAEformer	0.0553 ± 0.0002	0.0799 ± 0.0005	0.1292 ± 0.0052	0.1709 ± 0.0085	0.1831 ± 0.0130	0.2196 ± 0.0252
DCRNN	0.0554 ± 0.0002	0.0869 ± 0.0028	0.1999 ± 0.0309	0.4091 ± 0.0175	0.6563 ± 0.0215	0.8598 ± 0.0661
All-step NMSE						
STORM	0.0446 ± 0.0002	0.0578 ± 0.0004	0.0783 ± 0.0006	0.1234 ± 0.0016	0.1553 ± 0.0033	0.1736 ± 0.0033
LightSTORM	0.0452 ± 0.0001	0.0570 ± 0.0004	0.0763 ± 0.0010	0.1204 ± 0.0007	0.1495 ± 0.0026	0.1703 ± 0.0062
Graph WaveNet	0.0455 ± 0.0002	0.0594 ± 0.0016	0.0868 ± 0.0025	0.1560 ± 0.0053	0.2103 ± 0.0128	0.2607 ± 0.0105
STID	0.0459 ± 0.0003	0.0594 ± 0.0015	0.0856 ± 0.0009	0.1457 ± 0.0015	0.1969 ± 0.0014	0.2372 ± 0.0031
STAEformer	0.0454 ± 0.0001	0.0569 ± 0.0003	0.0770 ± 0.0018	0.1030 ± 0.0029	0.1178 ± 0.0031	0.1351 ± 0.0089
DCRNN	0.0451 ± 0.0001	0.0608 ± 0.0017	0.1070 ± 0.0164	0.2012 ± 0.0083	0.3224 ± 0.0153	0.4428 ± 0.0303

G.3 Empirical Stability Under Admissible Topology Perturbations

We evaluate the finite-horizon stability result by perturbing the weighted boundary operators while keeping the trained model, inputs, and initial states fixed. For each simplex order, we sample a perturbation direction $\mathbf{z}_k$ and define

$$\mathbf{D}_k = \mathrm{diag}(\exp(a\mathbf{z}_k)), \qquad \widehat{\mathbf{B}}_k = \mathbf{D}_{k-1}^{-1}\mathbf{B}_k\mathbf{D}_k.$$

The amplitude a is chosen so that

$$\eta = \max_k \frac{\|\widehat{\mathbf{B}}_k - \mathbf{B}_k\|_{\mathrm{op}}}{\|\mathbf{B}_k\|_{\mathrm{op}}} \in \{10^{-4}, 10^{-3}, 10^{-2}\}.$$

We use five perturbation directions for each dataset and perturbation magnitude. This construction preserves the simplex correspondence, incidence support and signs, and the chain identity. The Hodge operators are then recomputed from the perturbed boundary operators. All perturbations

Table 19: PeMS-D8 node forecasting. Final-step and all-step NMSE (mean ± standard deviation over five runs; lower is better).

Model	$S = 5$	$S = 10$	$S = 20$	$S = 40$	$S = 60$	$S = 80$
Final-step NMSE						
STORM	0.0432 ± 0.0005	0.0586 ± 0.0020	0.1001 ± 0.0038	0.1599 ± 0.0080	0.1935 ± 0.0056	0.2283 ± 0.0147
LightSTORM	0.0439 ± 0.0002	0.0585 ± 0.0004	0.0998 ± 0.0012	0.1438 ± 0.0093	0.1832 ± 0.0122	0.2007 ± 0.0089
Graph WaveNet	0.0448 ± 0.0018	0.0620 ± 0.0013	0.1181 ± 0.0037	0.2332 ± 0.0073	0.3415 ± 0.0232	0.4346 ± 0.0696
STID	0.0414 ± 0.0004	0.0599 ± 0.0008	0.1058 ± 0.0020	0.2053 ± 0.0026	0.3027 ± 0.0034	0.3673 ± 0.0056
STAEformer	0.0427 ± 0.0001	0.0644 ± 0.0004	0.1209 ± 0.0023	0.1925 ± 0.0082	0.2334 ± 0.0152	0.1965 ± 0.0325
DCRNN	0.0440 ± 0.0005	0.0670 ± 0.0029	0.1395 ± 0.0029	0.3285 ± 0.0295	0.5349 ± 0.0341	0.7199 ± 0.0482
All-step NMSE						
STORM	0.0342 ± 0.0004	0.0437 ± 0.0012	0.0633 ± 0.0012	0.0928 ± 0.0020	0.1152 ± 0.0029	0.1343 ± 0.0041
LightSTORM	0.0348 ± 0.0003	0.0433 ± 0.0002	0.0623 ± 0.0012	0.0857 ± 0.0025	0.1079 ± 0.0038	0.1216 ± 0.0039
Graph WaveNet	0.0337 ± 0.0011	0.0449 ± 0.0006	0.0684 ± 0.0008	0.1234 ± 0.0034	0.1765 ± 0.0086	0.2250 ± 0.0365
STID	0.0339 ± 0.0004	0.0440 ± 0.0006	0.0639 ± 0.0008	0.1128 ± 0.0012	0.1600 ± 0.0012	0.2028 ± 0.0026
STAEformer	0.0339 ± 0.0001	0.0445 ± 0.0002	0.0671 ± 0.0007	0.1074 ± 0.0021	0.1269 ± 0.0061	0.1258 ± 0.0123
DCRNN	0.0350 ± 0.0006	0.0470 ± 0.0010	0.0785 ± 0.0015	0.1598 ± 0.0099	0.2565 ± 0.0138	0.3556 ± 0.0204

satisfy the admissibility conditions used in Theorem 1, including the corresponding Hodge-operator perturbation bound. To measure the effect of the perturbation directly, for test sample w and forecast step s we compute

$$d_{w,s} = \left(\sum_{k \in \mathcal{K}_{\text{out}}} \left\| \widehat{\mathbf{Y}}^{(k)}_{w,s} - \mathbf{Y}^{(k)}_{w,s} \right\|_F^2 \right)^{1/2},$$

where $\mathbf{Y}$ and $\widehat{\mathbf{Y}}$ denote the predictions obtained with the clean and perturbed topology, respectively. We report the final-horizon discrepancy averaged over the test set and across perturbation directions, together with the corresponding clean and perturbed NMSE. The results are reported in Table 20. Across all three datasets, the output discrepancy grows approximately proportionally to η. Increasing η from 10^{-4} to 10^{-2} increases the discrepancy by $98.5\times$ on PeMS-D4, $96.1\times$ on PeMS-D8, and $96.7\times$ on Water. This is consistent with the linear dependence on η in the finite-horizon bound of Theorem 1. At the same time, predictive NMSE changes only marginally over the tested perturbation range. These results provide empirical support for the stability behavior predicted by the theorem under its admissible perturbation model.

Table 20: Complete final-step stability results at $S = 80$. Output discrepancy and ΔNMSE are mean ± sample standard deviation over five perturbation directions. Clean NMSE is fixed across directions. NMSE is pooled across active simplex orders.

Dataset	η	Final output discrepancy	Clean NMSE	Perturbed NMSE	ΔNMSE
PeMS-D4	10^{-4}	0.0080 ± 0.0009	0.310179	0.310173	-0.000006 ± 0.000013
	10^{-3}	0.0808 ± 0.0098	0.310179	0.310123	-0.000056 ± 0.000126
	10^{-2}	0.7922 ± 0.0858	0.310179	0.310049	-0.000130 ± 0.001156
PeMS-D8	10^{-4}	0.0072 ± 0.0007	0.237332	0.237327	-0.000005 ± 0.000009
	10^{-3}	0.0697 ± 0.0054	0.237332	0.237294	-0.000037 ± 0.000089
	10^{-2}	0.6872 ± 0.0579	0.237332	0.237544	$+0.000212 \pm 0.001064$
Water	10^{-4}	0.0017 ± 0.0004	0.646768	0.646766	-0.000002 ± 0.000004
	10^{-3}	0.0172 ± 0.0036	0.646768	0.646748	-0.000020 ± 0.000036
	10^{-2}	0.1659 ± 0.0337	0.646768	0.646865	$+0.000097 \pm 0.000400$

Table 21: NMSE pooled over all 80 forecast steps. Perturbed NMSE and ΔNMSE are mean ± sample standard deviation over five perturbation directions.

Dataset	η	Clean NMSE	Perturbed NMSE	ΔNMSE
PeMS-D4	10^{-4}	0.180714	0.180712 ± 0.000003	$-2.24 \times 10^{-6} \pm 3.48 \times 10^{-6}$
	10^{-3}	0.180714	0.180693 ± 0.000035	$-2.12 \times 10^{-5} \pm 3.48 \times 10^{-5}$
	10^{-2}	0.180714	0.180716 ± 0.000294	$+2.32 \times 10^{-6} \pm 2.94 \times 10^{-4}$
PeMS-D8	10^{-4}	0.135537	0.135536 ± 0.000002	$-1.61 \times 10^{-6} \pm 2.08 \times 10^{-6}$
	10^{-3}	0.135537	0.135521 ± 0.000021	$-1.60 \times 10^{-5} \pm 2.12 \times 10^{-5}$
	10^{-2}	0.135537	0.135580 ± 0.000261	$+4.24 \times 10^{-5} \pm 2.61 \times 10^{-4}$
Water	10^{-4}	0.480166	0.480166 ± 0.000003	$+1.92 \times 10^{-7} \pm 2.65 \times 10^{-6}$
	10^{-3}	0.480166	0.480171 ± 0.000027	$+4.45 \times 10^{-6} \pm 2.67 \times 10^{-5}$
	10^{-2}	0.480166	0.480482 ± 0.000244	$+3.16 \times 10^{-4} \pm 2.44 \times 10^{-4}$

Table 22: Scaling of the final-output discrepancy. The sensitivity column is the mean discrepancy divided by the realized perturbation magnitude.

Dataset	η	Final discrepancy	Discrepancy$/\eta$
PeMS-D4	10^{-4}	0.0080 ± 0.0009	80.40
	10^{-3}	0.0808 ± 0.0098	80.79
	10^{-2}	0.7922 ± 0.0858	79.22
PeMS-D8	10^{-4}	0.0072 ± 0.0007	71.50
	10^{-3}	0.0697 ± 0.0054	69.71
	10^{-2}	0.6872 ± 0.0579	68.72
Water	10^{-4}	0.0017 ± 0.0004	17.16
	10^{-3}	0.0172 ± 0.0036	17.16
	10^{-2}	0.1659 ± 0.0337	16.59

H Ablation Studies

We study how the topological operators and architectural choices contribute to STORM and LightSTORM.

H.1 Role of Topology

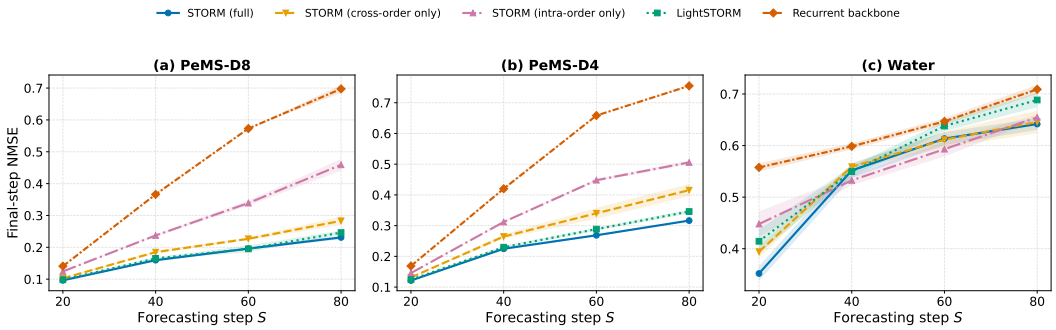


Figure 6: Topology ablation: final-step NMSE at $S \in \{20, 40, 60, 80\}$ (mean ± 1 std over seeds; lower is better).

We compare full STORM and LightSTORM with three ablations: an intra-order-only variant that retains Hodge-Laplacian filtering but removes cross-order coupling, a cross-order-only variant that retains boundary coupling but removes Hodge-Laplacian filtering, and a recurrent backbone with no topological operators. STORM and LightSTORM use the *both*-injection configuration. Tables 23–

25 and Figure 6 report the results. On PeMS-D4 and PeMS-D8, the topology-aware variants separate increasingly from the recurrent backbone as the forecasting horizon grows. Cross-order coupling contributes most strongly among the individual mechanisms: cross-order-only consistently outperforms intra-order-only at long horizons, while full STORM performs best overall, suggesting that Hodge filtering and boundary coupling are complementary. LightSTORM remains close to full STORM despite sharing its Dirac-filter parameters across simplex orders. On Water, the differences among topology variants are smaller and their ordering varies with the forecasting horizon. Nevertheless, the topology-aware variants outperform the recurrent backbone at the longest horizon. Overall, the ablation shows that the long-horizon gains are not reproduced by the topology-free recurrent backbone and that cross-order coupling plays an important role.

Table 23: Topology ablation on PeMS-D8: final-step NMSE (mean ± std; lower is better). STORM uses the *both* configuration. **Bold**: best. Underline: second best.

Model	**Topology**	S=20	S=40	S=60	S=80
STORM	Full	**0.0963**± 0.0004	**0.1596**± 0.0041	0.1959± 0.0033	**0.2310**± 0.0059
STORM	Cross-order only	0.1027± 0.0025	0.1848± 0.0046	0.2267± 0.0045	0.2831± 0.0130
STORM	Intra-order only	0.1243± 0.0010	0.2372± 0.0034	0.3389± 0.0082	0.4590± 0.0185
LightSTORM	Shared Dirac	0.0982± 0.0037	0.1652± 0.0037	**0.1951**± 0.0105	0.2462± 0.0067
Recurrent backbone	None	0.1408± 0.0003	0.3662± 0.0026	0.5729± 0.0002	0.6974± 0.0128

Table 24: Topology ablation on PeMS-D4: final-step NMSE (mean ± std; lower is better). STORM uses the *both* configuration. **Bold**: best. Underline: second best.

Model	**Topology**	S=20	S=40	S=60	S=80
STORM	Full	**0.1214**± 0.0013	**0.2247**± 0.0033	**0.2686**± 0.0031	**0.3164**± 0.0017
STORM	Cross-order only	0.1314± 0.0010	0.2645± 0.0072	0.3401± 0.0177	0.4155± 0.0198
STORM	Intra-order only	0.1456± 0.0019	0.3119± 0.0026	0.4479± 0.0046	0.5057± 0.0061
LightSTORM	Shared Dirac	0.1244± 0.0008	0.2297± 0.0051	0.2886± 0.0051	0.3458± 0.0061
Recurrent backbone	None	0.1687± 0.0012	0.4200± 0.0040	0.6583± 0.0005	0.7548± 0.0035

Table 25: Topology ablation on Water: final-step NMSE (mean ± std; lower is better). No triangles ($N_2 = 0$). STORM uses the *both* configuration. **Bold**: best. Underline: second best.

Model	**Topology**	S=20	S=40	S=60	S=80
STORM	Full	**0.3519**± 0.0144	0.5517± 0.0143	0.6135± 0.0125	**0.6417**± 0.0113
STORM	Cross-order only	0.3941± 0.0115	0.5589± 0.0057	0.6115± 0.0154	0.6458± 0.0218
STORM	Intra-order only	0.4478± 0.0246	**0.5325**± 0.0094	**0.5930**± 0.0143	0.6547± 0.0054
LightSTORM	Shared Dirac	0.4144± 0.0227	0.5495± 0.0103	0.6379± 0.0097	0.6884± 0.0144
Recurrent backbone	None	0.5576± 0.0081	0.5984± 0.0061	0.6469± 0.0065	0.7090± 0.0088

H.2 Injection Mode and Model Capacity

Tables 26–28 compare the injection modes and hidden dimensions of STORM and LightSTORM at $S \in \{10, 40, 80\}$. On PeMS-D4 and PeMS-D8, injecting topology into the hidden state, through either hidden or both injection, substantially outperforms input-only injection at long forecasting horizons. The differences are much smaller at $S = 10$, suggesting that topological interactions within the recurrent state become more important as the forecasting horizon increases. On Water, the preferred injection mode varies with the horizon and the differences are less consistent. Increasing the hidden dimension from $H = 16$ to $H = 32$ generally improves performance, particularly at medium and long horizons, but the effect is not uniform across configurations. At the same time, reduced-width STORM variants remain competitive, suggesting that the gains are not explained by

recurrent capacity alone. Parameter counts are reported in Appendix F.2.5. LightSTORM retains much of STORM's performance on the traffic datasets while using substantially fewer parameters. At $S = 80$, the best LightSTORM configuration is within approximately 7% of STORM on PeMS-D8 and 9% on PeMS-D4. Overall, these ablations suggest that hidden-state topological coupling contributes to the long-horizon gains, while LightSTORM provides a more parameter-efficient alternative to the full per-order parameterization.

Table 26: Injection-mode ablation on PeMS-D8: NMSE (mean ± std; lower is better). Blue/green bold mark the best STORM/LightSTORM configuration per horizon.

Model	Mode	H	S=10	S=40	S=80
STORM	input	16	0.0616± 0.0006	0.2956± 0.0047	0.5655± 0.0088
		32	0.0587± 0.0007	0.2233± 0.0083	0.3913± 0.0053
	hidden	16	0.0578± 0.0014	0.1908± 0.0018	0.3256± 0.0200
		32	0.0560± 0.0004	**0.1592**± 0.0093	0.2370± 0.0125
	both	16	0.0573± 0.0001	0.1880± 0.0110	0.3376± 0.0098
		32	**0.0541**± 0.0007	0.1596± 0.0041	**0.2310**± 0.0059
LightSTORM	input	16	0.0636± 0.0004	0.3570± 0.0057	0.6919± 0.0100
		32	0.0626± 0.0008	0.3105± 0.0066	0.5867± 0.0046
	hidden	16	0.0588± 0.0001	0.2123± 0.0043	0.4065± 0.0064
		32	**0.0552**± 0.0008	**0.1624**± 0.0055	0.2511± 0.0177
	both	16	0.0600± 0.0006	0.2155± 0.0106	0.3736± 0.0104
		32	0.0558± 0.0006	0.1652± 0.0037	**0.2462**± 0.0067

Table 27: Injection-mode ablation on PeMS-D4: NMSE (mean ± std; lower is better). Blue/green bold mark the best STORM/LightSTORM configuration per horizon.

Model	Mode	H	S=10	S=40	S=80
STORM	input	16	0.0762± 0.0005	0.3662± 0.0022	0.6643± 0.0159
		32	0.0750± 0.0003	0.2934± 0.0105	0.4708± 0.0166
	hidden	16	0.0771± 0.0004	0.2864± 0.0004	0.4660± 0.0031
		32	0.0728± 0.0008	0.2259± 0.0006	0.3287± 0.0179
	both	16	0.0762± 0.0004	0.2823± 0.0043	0.4589± 0.0340
		32	**0.0725**± 0.0004	**0.2247**± 0.0033	**0.3164**± 0.0017
LightSTORM	input	16	0.0794± 0.0011	0.4215± 0.0033	0.6103± 0.0156
		32	0.0782± 0.0005	0.3853± 0.0024	0.3486± 0.0143
	hidden	16	0.0776± 0.0004	0.3130± 0.0074	0.4919± 0.0080
		32	**0.0729**± 0.0009	0.2449± 0.0069	0.3626± 0.0074
	both	16	0.0764± 0.0004	0.2938± 0.0067	0.4919± 0.0080
		32	0.0735± 0.0005	**0.2297**± 0.0051	**0.3458**± 0.0061

Table 28: Injection-mode ablation on Water: NMSE (mean ± std; lower is better). No triangles ($N_2 = 0$). Blue/green bold mark the best STORM/LightSTORM configuration per horizon.

Model	Mode	H	S=10	S=40	S=80
STORM	input	16	0.2813± 0.0049	0.6022± 0.0072	0.7281± 0.0075
		32	0.2989± 0.0101	**0.5463**± 0.0090	0.6602± 0.0112
	hidden	16	0.3276± 0.0140	0.5904± 0.0102	0.7021± 0.0093
		32	**0.2720**± 0.0098	0.5471± 0.0176	0.6548± 0.0105
	both	16	0.3234± 0.0120	0.5872± 0.0083	0.6982± 0.0136
		32	0.2772± 0.0035	0.5517± 0.0143	**0.6417**± 0.0113
LightSTORM	input	16	0.3067± 0.0150	0.6485± 0.0077	0.7644± 0.0104
		32	**0.2726**± 0.0063	0.5986± 0.0125	0.7324± 0.0096
	hidden	16	0.3101± 0.0066	0.6096± 0.0122	0.7183± 0.0076
		32	0.3054± 0.0053	0.5604± 0.0077	**0.6768**± 0.0055
	both	16	0.3199± 0.0146	0.6338± 0.0143	0.7362± 0.0045
		32	0.3016± 0.0061	**0.5495**± 0.0103	0.6884± 0.0144

H.3 Role of Increasing Topological Order

We compare five progressively structured variants: a topology-free node-only recurrent model; graph preprocessing, where a graph filter is applied before the recurrence; graph recurrence, where graph filters are embedded inside the recurrent gates; node–edge recurrence, which adds edge states and node–edge coupling; and node–edge–triangle recurrence, which additionally includes triangle states and order-two coupling. All variants use hidden width 32 and degree-three filters where applicable. We report final-step node NMSE, since node predictions are shared by all variants.

The results show a clear progression as topological structure is incorporated more deeply into the recurrent representation. Graph preprocessing alone provides little improvement over the topology-free model, whereas embedding graph filters inside the recurrent gates substantially improves forecasting. Adding coupled edge states yields a further gain in every setting. Including triangle states improves performance again in five of the six dataset–horizon combinations, with PeMS-D4 at $S = 80$ as the only exception. Overall, these results indicate that the gains increase as higher-order information is incorporated into the recurrent dynamics, with coupled node–edge and node–edge–triangle representations providing the strongest performance.

Table 29: Topology/order ladder: final-step node NMSE (mean $\pm$ standard deviation over three runs; lower is better). **Bold**: best. Underline: second best.

Dataset	S	No topology	Graph preproc.	Graph recurrent	Node–edge	Node–edge–triangle
PeMS-D4	20	0.1767 ± 0.0015	0.1778 ± 0.0042	0.1416 ± 0.0040	0.1260 ± 0.0018	$\mathbf{0.1242 \pm 0.0018}$
	40	0.4244 ± 0.0024	0.4221 ± 0.0008	0.2934 ± 0.0006	0.2312 ± 0.0073	$\mathbf{0.2295 \pm 0.0030}$
	80	0.7580 ± 0.0035	0.7490 ± 0.0034	0.4703 ± 0.0036	$\mathbf{0.3089 \pm 0.0038}$	0.3223 ± 0.0036
PeMS-D8	20	0.1429 ± 0.0047	0.1425 ± 0.0029	0.1169 ± 0.0053	0.1003 ± 0.0024	$\mathbf{0.0991 \pm 0.0013}$
	40	0.3424 ± 0.0051	0.3461 ± 0.0023	0.1972 ± 0.0090	0.1628 ± 0.0068	$\mathbf{0.1591 \pm 0.0010}$
	80	0.6597 ± 0.0090	0.6525 ± 0.0202	0.3635 ± 0.0286	0.2404 ± 0.0136	$\mathbf{0.2218 \pm 0.0106}$

H.4 Per-Order Forecasting Diagnostics

To better understand how forecasting performance is distributed across simplex orders, we report final-step NMSE separately for node, edge, and triangle signals on the two traffic datasets. We focus on medium and long forecasting horizons, $S{=}40$ and $S{=}80$, where the main results show the largest separation between topology-aware models and recurrent baselines. Unlike the pooled NMSE in the main tables, per-order NMSE normalizes errors within each simplex order and is therefore intended as a diagnostic view rather than a replacement for the main metric.

Tables 30–31 show that STORM variants remain competitive across node, edge, and triangle orders. On PeMS-D8, STORM gives the lowest or tied-lowest errors across all orders at $S{=}80$, and remains close to the best method at $S{=}40$. On PeMS-D4, STORM gives the lowest node and edge errors at both horizons, while the lowest triangle error is obtained by STORM (hidden) at $S{=}40$ and LightSTORM (both) at $S{=}80$. Overall, these diagnostics suggest that the improvements observed in the pooled metrics are not confined to node-level prediction, but also extend to the induced edge- and triangle-level signals.

Table 30: Per-order final-step NMSE on PeMS-D4 at $S{=}40$ and $S{=}80$. Values are means over 5 seeds. Lower is better. **Bold**: best mean. Underline: second-best mean.

Model	$S{=}40$			$S{=}80$		
	Nodes ($k{=}0$)	Edges ($k{=}1$)	Triangles ($k{=}2$)	Nodes ($k{=}0$)	Edges ($k{=}1$)	Triangles ($k{=}2$)
STORM (both)	**0.2318**	**0.2200**	0.2030	**0.3278**	**0.3120**	0.2433
STORM (hidden)	0.2324	0.2228	**0.1930**	0.3380	0.3264	0.2535
LightSTORM (both)	0.2360	0.2265	0.1990	0.3560	0.3453	**0.2377**
LightSTORM (hidden)	0.2500	0.2432	0.2075	0.3769	0.3730	0.2803
iTransformer	0.2982	0.2878	0.3046	0.5307	0.5273	0.4904
xLSTM	0.4230	0.4105	0.4296	0.7504	0.7621	0.7615
GCN-xLSTM	0.3347	0.3840	0.4482	0.5863	0.6902	0.7532
SC-VAR	0.5827	0.6183	0.6822	0.9027	0.9619	0.9383

Table 31: Per-order final-step NMSE on PeMS-D8 at $S{=}40$ and $S{=}80$. Values are means over 5 seeds. Lower is better. **Bold**: best mean. Underline: second-best or tied second-best mean.

Model	$S{=}40$			$S{=}80$		
	Nodes ($k{=}0$)	Edges ($k{=}1$)	Triangles ($k{=}2$)	Nodes ($k{=}0$)	Edges ($k{=}1$)	Triangles ($k{=}2$)
STORM (both)	<u>0.1600</u>	<u>0.1622</u>	**0.1496**	**0.2272**	**0.2379**	**0.2153**
STORM (hidden)	**0.1598**	**0.1612**	<u>0.1510</u>	<u>0.2316</u>	<u>0.2439</u>	<u>0.2244</u>
LightSTORM (both)	0.1630	0.1686	0.1584	0.2352	0.2569	<u>0.2244</u>
LightSTORM (hidden)	0.1609	0.1659	0.1532	0.2438	0.2627	0.2265
iTransformer	0.2440	0.2460	0.2443	0.4488	0.4742	0.4718
xLSTM	0.3466	0.3646	0.3681	0.6621	0.7303	0.7374
GCN-xLSTM	0.2371	0.3104	0.3391	0.4453	0.6063	0.6627
SC-VAR	0.5220	0.5840	0.6080	0.7946	0.9392	0.9483

I Limitations and Future Work

STORM is developed for signals evolving over a fixed simplicial complex. This is the problem setting considered in this work and is suitable when the interaction structure is known and doesn't change over time. A natural extension could be working on topologies that evolve in time, considering, for example, to use time-indexed boundary and Hodge operators. The more challenging case would be when simplices appear or disappear, since the recurrent states must then be aligned across complexes with different simplex sets.

A second limitation concerns the availability of native higher-order signals in the current stage of the field. On the traffic datasets, measurements are observed at nodes and higher-order edge and triangle signals are induced from node-level observations. In contrast, Water provides native node and edge measurements but no triangle-level signals. While our experiments show that the multi-order recurrent representation can be beneficial even when higher-order inputs are constructed from node observations, additional benchmarks with native multi-order measurements would help clarify when such measurements provide further benefits and in which settings higher-order structure is most useful.

The model is permutation equivariant but not orientation equivariant. Simplex orientations must therefore be fixed consistently, and changing them can alter the learned representations and predictions. Developing recurrent architectures that are equivariant to orientation changes is an interesting direction for future work.

Finally, the theoretical analysis is developed under explicit sufficient assumptions on operator perturbations, boundedness, and recurrent sensitivity. These assumptions yield finite-horizon stability and generalization guarantees, while the empirical perturbation experiments support the predicted stability behavior. A natural direction for future work is to extend these results to broader perturbation models and weaker assumptions.