OMNISHEAF: UNIFYING SHEAF HYPERGRAPH NETWORKS VIA EFFICIENT UNIVERSAL APPROXIMATORS

Gabriele Onorato* Northeastern University
Nathaniel D. Bastian Johns Hopkins University
Francesco Restuccia Northeastern University

ABSTRACT

Higher-order relations among multiple entities frequently appear in fields such as neuroscience, radio allocation and computer networks. These relations can be modeled via *hypergraphs*, i.e., topological structures that extend traditional graphs via *hyperedges* connecting multiple nodes at once. Recent work has introduced Sheaf Hypergraph Networks (SHNs), which further augment this topological structure via a *cellular sheaf*. The latter attaches d-dimensional vector spaces named *stalks* to each node and hyperedge, as well as learnable linear transformations called *restriction maps* between them. The key limitation of current sheaf methods for hypergraphs is that they compute the *graph clique*, which scales *quadratically* with the number of elements of each hyperedge. Ultimately, this causes loss of valuable topological information and makes computation prohibitive. In this paper, we introduce OMNISHEAF (OS), a new Sheaf Neural Network (SNN) that for the first time works directly over hypergraphs via multiset functions over stalk spaces. We mathematically prove that OS generalizes SNNs, SHNs, ALLSET, DHNN, and Dir-GNN via its universal approximators and that it can be made directionality-aware over hyperedges via both a complex phase and tail-head indices. Extensive experiments across 25 benchmarks against hypergraph networks, set-based architectures, and sheaf models demonstrate that OS achieves up to $+14.37$ higher accuracy on undirected node classification benchmarks (on Senate, while reaching 95.58% on Congress) and $+4.11$ on directed ones, while on ROMAN-EMPIRE it trains $7.1\times$ faster than DSHNLight, uses $3.6\times$ less peak GPU memory and $14.9\times$ fewer training FLOPs at equal stalk dimension.

1 INTRODUCTION

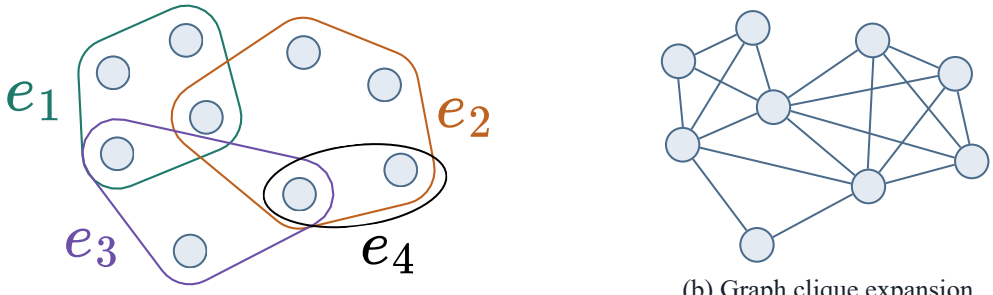
Hypergraphs extend graphs to higher-order relations and enable nodes to be connected via hyperedges, i.e., a set-type relation between multiple nodes. Hypergraphs have several applications in numerous fields such as computer networking (Liu et al., 2024), neuroscience (Guo et al., 2021) and radio allocation (Zhang et al., 2016), just to name a few. Learning over these structures is usually performed through Graph Neural Networks (GNNs) (Gori et al., 2005; Scarselli et al., 2009; Kipf & Welling, 2017). As illustrated in Figure 1, pairwise graph reductions eliminate polyadic interactions. As such, Hypergraph Neural Neworks (HNNs) are usually preferable over GNNs (Feng et al., 2019; Yadati et al., 2019; Dong et al., 2020).

Cellular Sheaves. Similar to GNNs, HNNs suffer from *oversmoothing*, i.e., the phenomenon where repeated message-passing causes node representations to converge to an indistinguishable state as network depth increases (Chen et al., 2022; Rusch et al., 2023). Recent work has reduced this issue with *cellular sheaves* (Curry, 2014), formed by stalks (d-dimensional vector spaces) over nodes and edges, and linear restriction maps between them (Hansen & Gebhart, 2020). Such augmented graph topology is then used by Sheaf Neural Networks (SNNs) and Sheaf Hypergraph Networks (SHNs) to capture higher-order relationship effectively and reduce oversmoothing (Bodnar et al., 2022; Duta et al., 2023; Mule et al., 2026; Fiorini et al., 2026).

Key Issues. The SHNs introduced by Duta et al. (2023) compute the *graph clique* of the hypergraph, i.e. replacing each hyperedge e with a pairwise complete subgraph $K_{|e|}$ before running graph sheaf diffusion (Bodnar et al., 2022). This projection is structurally lossy and non-injective. This

*Corresponding author: onorato.g@northeastern.edu

means that distinct polyadic motifs, as illustrated in Figure 1, collapse into identical pairwise graphs, thus discarding the multi-way boundary topology of HNNs. Algebraically, factoring joint polyadic relations into pairwise compositions $\mathcal{F}_{v \trianglelefteq e}^{\top} \mathcal{F}_{u \trianglelefteq e}$ across expanded edges introduces spurious cycle holonomies, which fails to guarantee the positive semi-definiteness of the resulting sheaf Laplacian necessary for the convergence of the diffusion process, which Mule et al. (2026) recovered via a complex-valued Laplacian. Another key issue is that the graph clique scales quadratically, i.e., for each hyperedge $e \in \mathcal{E}$ the cost of the graph clique is $\mathcal{O}(\sum_{e \in \mathcal{E}} |e|^2)$, thus making computation infeasible on large hypergraphs (Agarwal et al., 2006), as shown in Figure 2.



(a) Hypergraph

(b) Graph clique expansion of the hypergraph

Figure 1: *The graph clique of a hypergraph.* **(a)** A hypergraph composed of 4 relations. **(b)** The graph clique expansion of a hypergraph: here the hyperedge e_4 is lost due to it being a subset of e_2, thus losing topological information.

Proposed Approach. This paper radically departs from existing work by introducing OS, a new neural network designed for sheaf learning over directed and undirected hypergraphs. Rather than flattening hyperedges into graphs or pooling over unaligned private node coordinates, OS formulates two-stage message passing via multiset functions across stalk spaces (Wagstaff et al., 2019). In the forward phase, node representations are transported into the common hyperedge stalk $\mathcal{F}(e)$ via incidence restriction maps $\mathcal{F}_{u \trianglelefteq e}$. In the backward phase, hyperedge representations are co-projected back to node stalks alongside their local transport blocks $\mathcal{F}_{u \trianglelefteq e}^{\top} \mathcal{F}_{u \trianglelefteq e}$. This joint aggregation enables us to perform sheaf learning without computing global graph Laplacians or pairwise clique interactions. In Section 3.2, we prove that OS generalizes both directed and undirected SNNs (Bodnar et al., 2022; Barbero et al., 2022; Fiorini et al., 2026) and SHNs (Duta et al., 2023; Mule et al., 2026), while also avoiding the graph clique computation entirely. As such, OS reduces the cost from $\mathcal{O}(\sum_{e \in E} |e_i|^2)$ to $\mathcal{O}(\sum_{e \in E} |e_i|)$ (Theorem 4). We further introduce OmniDeepSheaf (ODS) and OmniSheafTransformer (OST) that are proven to be universal approximators of OS (Theorem 8). Finally, we show in Section 3.4 that OS can be made directionality-aware via complex-valued phase (Fiorini et al., 2026) or with tail-head indices (Rossi et al., 2024), thus obtaining OmniPhaseSheaf (OPS) and OmniCopreSheaf (OCS) respectively.

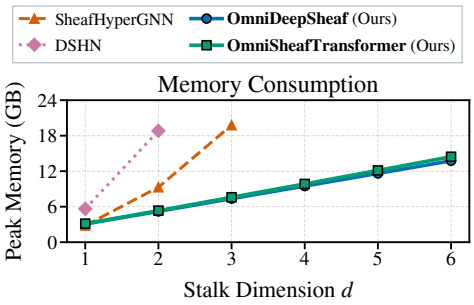


Figure 2: *Cost on Roman-empire as the stalk dimension d grows.* OS grows linearly in d in peak training memory, while DSHN and SheafHyperGNN grow quadratically and exhaust the 24 GB card at $d = 3$ and $d = 4$.

Summary of Novel Contributions:

❶ We introduce OMNISHEAF along with two universal approximators, *OmniDeepSheaf* and *OmniSheafTransformer* (Theorem 8 and Corollary 9). We prove that they generalize all current hypergraph sheaf models (Duta et al., 2023), graph SNNs (Bodnar et al., 2022; Barbero et al., 2022), and AllSet (Chien et al., 2022) (Theorems 5, 6 and 7).

❷ **Linear Space and Time Complexity:** We prove that OMNISHEAF reduces the complexity of sheaf message passing from $\mathcal{O}(\sum_{e \in \mathcal{E}} |e|^2)$ to $\mathcal{O}(\sum_{e \in \mathcal{E}} |e|)$, thus avoiding the quadratic graph clique expansion and avoiding the out-of-memory bottlenecks of prior sheaf methods (Theorem 4).

❸ **Directionality-Aware OmniSheaf:** We introduce *OmniPhaseSheaf* (via complex phases) and OmniCopreSheaf (via tail-head indices) for directed hypergraphs, proving they generalize Dir-

GNN (Rossi et al., 2024), DHNN (Fiorini et al., 2024), DSNN (Fiorini et al., 2026), and DSHN (Mule et al., 2026) (Theorems 10 and 11).

❹ **Empirical Performance & Efficiency:** Across 25 benchmarks spanning directed, undirected, synthetic, and molecular tasks, OS sets the best reported accuracy on the three committee hypergraphs, on both molecular tasks and on seven of the twelve directed benchmarks, gaining up to $+14.37$ on Senate and $+4.11$ on Roman-empire and reaching 100% on synthetic hypergraph benchmarks, while trailing the strongest sheaf baselines by at most 2.3 points on the homophilic citation benchmarks. On an NVIDIA L4 GPU and on ROMAN-EMPIRE, OMNISHEAF trains $7.1\times$ faster than DSHN at $d = 2$ while using $3.6\times$ less peak memory, cuts a training step from 19,286 to 1292 GFLOPs at $d = 6$ ($14.9\times$), and reaches $d = 6$ where DSHN and SheafHyperGCN already exhaust a 24 GB GPU at $d = 3$. On small dense hypergraphs such as TELEGRAM the clique baselines remain cheaper, and we report this explicitly in Section 4.4.

2 RELATED WORK

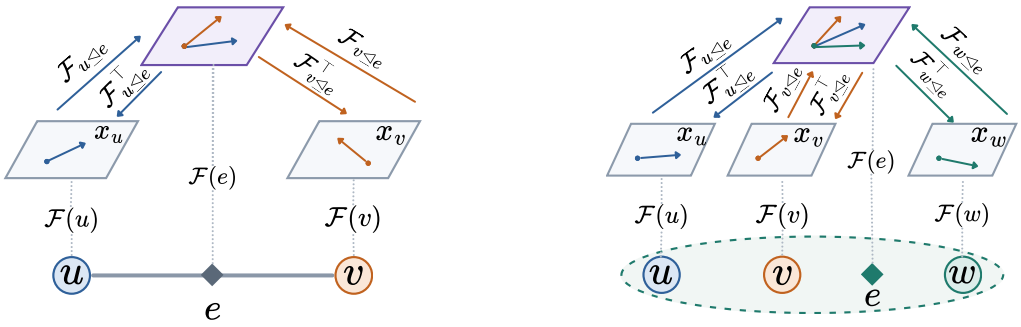
Graph Neural Networks. GNNs (Gori et al., 2005; Scarselli et al., 2009) are the most common framework for modeling pairwise interactions. They operate by propagating messages along edges (Kipf & Welling, 2017; Veličković et al., 2018). However they are fundamentally limited to binary connections and fail to capture higher-order, multi-entity relationships (Morris et al., 2019).

Hypergraph Neural Networks. HNNs overcome the limitations of GNN by processing multi-node hyperedges to capture complex group interactions. Foundational methods approached this topology from different directions: HyperGNN (Feng et al., 2019) introduced hypergraph convolutions based on a defined hypergraph Laplacian, while HyperGCN (Yadati et al., 2019) approximated hyperedge interactions through non-linear graph mediator structures. Subsequent architectures refined the message-passing mechanisms. HNHN (Dong et al., 2020) implemented asymmetric incidence matrix normalizations, and HCHA (Bai et al., 2021) incorporated attention mechanisms to dynamically weight node-hyperedge relations. To address the severe oversmoothing encountered in deep networks, UniGCNII (Huang & Yang, 2021) leverages identity mapping and residual connections, LEGCN (Yang et al., 2022) utilizes line graph expansions to maintain structural integrity, HyperND (Prokopchik et al., 2022) applies continuous neural diffusion processes, ED-HNN (Wang et al., 2023a) relies on equivariant hypergraph diffusion, and PhenomNN (Wang et al., 2023b) defines a hypergraph energy function to constrain node representations. For applications involving asymmetric data flows, GeDi-HNN (Fiorini et al., 2024) introduces generalized directed operators over hypergraphs, while DHNN (Fiorini et al., 2024) computes distinct head and tail projections to model directed hyperedges explicitly.

Set-Based Learning. Moving away from matrix expansions, recent paradigms frame hyperedges as multisets. However, representing set-valued functions introduces fundamental bottlenecks; universal permutation-invariant functions require strict latent dimensionality and continuity constraints (Wagstaff et al., 2019; Tabaghi & Wang, 2024). Foundational models like DeepSets (Zaheer et al., 2017) and SetTransformer (Lee et al., 2019) navigate these limits. Chien et al. (2022) integrated these functions into the AllDeepSets and AllSetTransformer architectures to operate as universal approximators directly on the hypergraph topology. However AllSet is strictly scalar-valued and lacks topological augmentation such as those provided by cellular sheaves.

Hypergraph Sheaves. Cellular sheaves augment topological spaces to increase expressivity (Curry, 2014). Formally, a cellular sheaf $\mathcal{F}$ over a hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{E} \subseteq 2^{\mathcal{V}}$; with the notation $u \trianglelefteq e$ for $u \in e$ attaches a d-dimensional vector space called *stalk* $\mathcal{F}(v) \in \mathbb{R}^d$ to each node $v \in \mathcal{V}$ and $\mathcal{F}(e) \in \mathbb{R}^d$ to each hyperedge $e \in \mathcal{E}$, with restriction maps $\mathcal{F}_{v \trianglelefteq e}$ and $\mathcal{F}^{\top}_{v \trianglelefteq e}$ (both $\in \mathbb{R}^{d \times d}$) for each $v \in e$, stacking nodes yields the 0-cochain space $C^0(\mathcal{H}, \mathcal{F}) \coloneqq \bigoplus_{v \in \mathcal{V}} \mathcal{F}(v)$ while hyperedge stalks produces the 1-cochain space $C^1(\mathcal{H}, \mathcal{F}) \coloneqq \bigoplus_{e \in \mathcal{E}} \mathcal{F}(e)$, transport over the sheaf between node-hyperedge relations is performed by combining restriction maps, i.e., if we want to move a node feature $x_v \in \mathcal{F}(v)$ to $\mathcal{F}(u)$ with both $u, v \in e$ we apply two restriction maps $\mathcal{F}^{\top}_{u \trianglelefteq e}(\mathcal{F}_{v \trianglelefteq e} x_v)$ in sequence, which can be interpreted in opinion theory as moving from the private opinion space $\mathcal{F}(v)$ to the shared space $\mathcal{F}(e)$ and vice-versa (Hansen & Ghrist, 2021). Duta et al. (2023) designed SheafHyperGNN and SheafHyperGCN to apply sheaf operators on undirected hypergraphs, with linear and nonlinear sheaf graph laplacians respectively (Zaghen et al., 2024). Mule et al. (2026) developed DSHN and the scalable DSHNLight to capture asymmetric flows on directed hypergraphs

via a complex-valued graph sheaf laplacian. However all these sheaf models compute the graph clique which is non-injective, scales quadratically as $\mathcal{O}(\sum_{e\in\mathcal{E}}|e|^2)$, and destroys structural information (see Figure 1), thus making computation infeasible on large hypergraphs (Agarwal et al., 2006).



(a) Cellular sheaf on a graph (b) Cellular sheaf on a hypergraph

Figure 3: *Cellular Sheaves on Graphs vs. Hypergraphs.* **(a)** *Graph case:* node stalks $\mathcal{F}(u), \mathcal{F}(v)$ map to edge stalk $\mathcal{F}(e)$ via restriction maps $\mathcal{F}_{u\trianglelefteq e}$ and $\mathcal{F}_{v\trianglelefteq e}$. **(b)** *Hypergraph case:* arbitrary node stalks map directly to a shared hyperedge stalk $\mathcal{F}(e)$.

3 Methodology

3.1 OmniSheaf

A *cellular sheaf* $\mathcal{F}$ on $\mathcal{H}$ assigns a stalk $\mathbb{R}^d$ to every node and every hyperedge and, to every incidence $u \trianglelefteq e$, a linear *restriction map* $\mathcal{F}_{u\trianglelefteq e}\colon \mathcal{F}(u) \to \mathcal{F}(e)$, which is represented by a matrix in $\mathbb{R}^{d\times d}$ (Curry, 2014; Duta et al., 2023). We denote the incidence set as $\mathcal{I} = \{(u, e) : u \trianglelefteq e\}$ and define its size as $K = |\mathcal{I}| = \sum_{e\in\mathcal{E}} \delta_e$.

In hypergraphs with node features, each cell (i.e., a node or hyperedge) does not carry a single stalk vector but rather a stalk-valued *feature matrix*. Specifically, node u holds $\boldsymbol{X}_u \in \mathbb{R}^{d\times f}$ and hyperedge e holds $\boldsymbol{H}_e \in \mathbb{R}^{d\times f}$, where f denotes the feature dimension. Restriction maps operate along the d (stalk) axis, while learned channel maps act on the f axis. For any function $g\colon \mathbb{R}^f \to \mathbb{R}^f$ and matrix $\boldsymbol{A} \in \mathbb{R}^{d\times f}$, as such, we write $g[\boldsymbol{A}]$ to indicate *row-wise* application, meaning $(g[\boldsymbol{A}])_{i,:} = g(\boldsymbol{A}_{i,:})$.

Definition 1 (OmniSheaf)**.** Let $f_{\mathcal{V}\to\mathcal{E}}$ and $f_{\mathcal{E}\to\mathcal{V}}$ be functions that are permutation invariant with respect to their first argument. We define the multisets $V_e = \{\{\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u : u \in e\}\}$ and $E_u = \{\{(\mathcal{F}_{u\trianglelefteq e}^{\top}\boldsymbol{H}_e,\ \mathcal{F}_{u\trianglelefteq e}^{\top}\mathcal{F}_{u\trianglelefteq e}) : e \ni u\}\}$, which leads to the following update rules:

$$\begin{aligned}\boldsymbol{H}_e^{(k+1)} &= f_{\mathcal{V}\to\mathcal{E}}\big(V_e^{(k)}; \boldsymbol{H}_e^{(k)}\big),\\ \boldsymbol{X}_u^{(k+1)} &= f_{\mathcal{E}\to\mathcal{V}}\big(E_u^{(k+1)}; \boldsymbol{X}_u^{(k)}\big).\end{aligned} \tag{1}$$

The first stage of this architecture lifts each incident node feature into the hyperedge stalk via the restriction map $\mathcal{F}_{u\trianglelefteq e}$, performing aggregation within the shared hyperedge space. The second stage co-projects the resulting hyperedge feature back into the node stalk via the adjoint map $\mathcal{F}_{u\trianglelefteq e}^{\top}$, returning to the private opinion space (Hansen & Ghrist, 2021). Crucially, pairing the transported feature $\mathcal{F}_{u\trianglelefteq e}^{\top}\boldsymbol{H}_e$ with the local transport block $\mathcal{F}_{u\trianglelefteq e}^{\top}\mathcal{F}_{u\trianglelefteq e}$ within the multiset E_u incurs in zero additional computational overhead because both depend solely on the incident restriction map. This formulation preserves symmetries (Propositions 2 and 3) over incident hyperedges while allowing the node aggregation function $f_{\mathcal{E}\to\mathcal{V}}$ to natively evaluate the sheaf degree block $\boldsymbol{D}_u = \sum_{e\ni u} \mathcal{F}_{u\trianglelefteq e}^{\top}\mathcal{F}_{u\trianglelefteq e}$. Consequently, OS computes normalized sheaf diffusion entirely locally and spectrum-free, which provides the foundation for subsuming classical sheaf diffusion architectures (Theorems 6, 7 and 11).

3.2 Theoretical Properties of OmniSheaf

We now demonstrate that OS acts as a universal approximator that generalizes several hypergraph and sheaf architectures, while resolving the quadratic scaling bottlenecks of previous sheaf methods. Complete formal proofs are deferred to Appendix A.

Proposition 2 (Permutation Symmetries)**.** *Let $f_{\mathcal{V}\to\mathcal{E}}$ and $f_{\mathcal{E}\to\mathcal{V}}$ be permutation-invariant multiset functions. Then,* OS *updates (Definition 1) are invariant to the local ordering of incident cells within V_e and E_u, and globally permutation-equivariant with respect to joint node and hyperedge reorderings.*

Proposition 3 (Sheaf Gauge Equivariance)**.** *Let $\mathcal{Q} = \{\boldsymbol{Q}_u \in \mathrm{O}(d) : u \in \mathcal{V}\} \cup \{\boldsymbol{Q}_e \in \mathrm{O}(d) : e \in \mathcal{E}\}$ be an orthogonal gauge transformation acting on cell stalks. Under the coordinate changes $\boldsymbol{X}_u \mapsto \boldsymbol{Q}_u\boldsymbol{X}_u$, $\boldsymbol{H}_e \mapsto \boldsymbol{Q}_e\boldsymbol{H}_e$, and $\mathcal{F}_{u\trianglelefteq e} \mapsto \boldsymbol{Q}_e\mathcal{F}_{u\trianglelefteq e}\boldsymbol{Q}_u^\top$, the transformed multiset elements satisfy $\tilde{\mathcal{F}}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u = \boldsymbol{Q}_e(\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u)$ and $\tilde{\mathcal{F}}^\top_{u\trianglelefteq e}\tilde{\boldsymbol{H}}_e = \boldsymbol{Q}_u(\mathcal{F}^\top_{u\trianglelefteq e}\boldsymbol{H}_e)$. Consequently, whenever $f_{\mathcal{V}\to\mathcal{E}}$ and $f_{\mathcal{E}\to\mathcal{V}}$ are stalk-orthogonal equivariant (as satisfied by linear sheaf diffusion), the* OS *forward pass is gauge equivariant:*

$$\boldsymbol{H}_e^{(k+1)} \mapsto \boldsymbol{Q}_e\boldsymbol{H}_e^{(k+1)} \quad \text{and} \quad \boldsymbol{X}_u^{(k+1)} \mapsto \boldsymbol{Q}_u\boldsymbol{X}_u^{(k+1)}. \tag{2}$$

For OmniPhaseSheaf, the result extends to $\mathrm{U}(d)$ gauge transformations in the complex stalk bundle. For non-linear row-wise neural parameterizations, equivariance is preserved over the stalk coordinate permutation subgroup $\mathcal{P}_d \leq \mathrm{O}(d)$ (and signed permutations for odd activations).

Theorem 4 (Time and Space Complexity Reduction)**.** OMNISHEAF *reduces the space and time complexity of sheaf hypergraph message passing from $\mathcal{O}(\sum_{e\in\mathcal{E}}|e|^2)$ to $\mathcal{O}(\sum_{e\in\mathcal{E}}|e|)$.*

Proof sketch. Prior methods expand each hyperedge e into $|e|(|e|-1)/2$ pairwise connections, scaling as $\mathcal{O}(|e|^2)$. In contrast, OS operates over the bipartite incidence graph $\mathcal{I} = \{(u, e) : u \in e\}$. In stage one, each node $u \in e$ sends one projected message $\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u$ to e; in stage two, e sends one message back to each $u \in e$. Each stage performs exactly $|\mathcal{I}| = \sum_{e\in\mathcal{E}}|e|$ local stalk operations, reducing both time and space complexity to $\mathcal{O}(\sum_{e\in\mathcal{E}}|e|)$. □

Theorem 5. ALLSET *(Chien et al., 2022) is a special case of* OMNISHEAF.

Proof sketch. Setting stalk dimension $d = 1$ and trivial restriction maps $\mathcal{F}_{u\trianglelefteq e} = 1 \in \mathbb{R}^{1\times 1}$ collapses the multisets to $V_e = \{\{\boldsymbol{x}_u^\top : u \in e\}\}$ and $E_u = \{\{(\boldsymbol{h}_e^\top, 1) : e \ni u\}\}$. The constant transport block $\mathcal{F}^\top_{u\trianglelefteq e}\mathcal{F}_{u\trianglelefteq e} = 1$ is trivially absorbed by $f_{\mathcal{E}\to\mathcal{V}}$, exactly reproducing the AllSet forward pass. □

Theorem 6. *SheafHyperGNN and SheafHyperGCN (Duta et al., 2023) are special cases of* OMNISHEAF.

Proof sketch. The linear sheaf hypergraph Laplacian diffuses differences $\frac{1}{\delta_e}\sum_{v\in e\setminus\{u\}}(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u - \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v) = \mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u - \frac{1}{\delta_e}\sum_{v\in e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v$. In Stage 1 of OS, $f_{\mathcal{V}\to\mathcal{E}}$ computes the mean-pooled hyperedge signal $\boldsymbol{H}_e = \frac{1}{\delta_e}\sum_{v\in e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v$. In Stage 2, $f_{\mathcal{E}\to\mathcal{V}}$ forms the discrepancy $(\mathcal{F}^\top_{u\trianglelefteq e}\mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u - \mathcal{F}^\top_{u\trianglelefteq e}\boldsymbol{H}_e$ from the local transport pair in E_u and local state $\tilde{\boldsymbol{X}}_u$ without requiring δ_e, while summing transport blocks to form $\boldsymbol{D}_u^{-1/2}$ locally. For SheafHyperGCN, $f_{\mathcal{V}\to\mathcal{E}}$ evaluates pairwise distances over V_e and selects the maximal discrepancy pair. □

Theorem 7. *Sheaf Neural Networks on graphs (Bodnar et al., 2022; Barbero et al., 2022), including Neural Sheaf Diffusion (NSD) and Sheaf Attention Network (SheafAN), are special cases of* OMNISHEAF.

Proof sketch. Representing a graph as a 2-uniform hypergraph ($|e| = 2$ for each edge $e = \{u, v\}$), Stage 1 computes $\boldsymbol{H}_e = \frac{1}{2}(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u + \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v)$. In Stage 2, $f_{\mathcal{E}\to\mathcal{V}}$ forms the local discrepancy $(\mathcal{F}^\top_{u\trianglelefteq e}\mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u - \mathcal{F}^\top_{u\trianglelefteq e}\boldsymbol{H}_e = \frac{1}{2}\mathcal{F}^\top_{u\trianglelefteq e}(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u - \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v)$ and normalizes by the sheaf degree $\boldsymbol{D}_u$, recovering NSD. Symmetric attention $\alpha_{uv} = \alpha_e$ is recovered directly by hyperedge weighting in $f_{\mathcal{V}\to\mathcal{E}}$, while general asymmetric attention is recovered by the directed variant OCS. □

Universal Approximators. To instantiate OS we introduce two concrete realizations: we define OmniDeepSheaf (ODS) and OmniSheafTransformer (OST) architectures by parameterizing $f_{\mathcal{V}\to\mathcal{E}}$ and $f_{\mathcal{E}\to\mathcal{V}}$ with MLPs (Zaheer et al., 2017) and attention mechanisms (Lee et al., 2019) respectively. When parameterized over vectorized stalks $\mathrm{vec}(\boldsymbol{A}) \in \mathbb{R}^{df}$, ODS enjoys universal approximation guarantees, while its row-wise instantiation provides a parameter-efficient $\mathcal{O}(f^2)$ formulation. OST natively processes joint stalk-channel representations via SetTransformer.

Theorem 8. *OmniSheafTransformer, as well as OmniDeepSheaf with vectorized stalks, are universal approximators of continuous* OMNISHEAF *multiset functions.*

Proof sketch. Vectorizing stalk matrices $\mathrm{Vec}(\cdot)$ is a linear isomorphism mapping matrix multisets to vector multisets in $\mathbb{R}^{df}$. By DeepSets universality (Zaheer et al., 2017; Wagstaff et al., 2019), MLPs over vectorized stalks with latent dimension $\geq \max_e |e|$ universally approximate continuous permutation-invariant multiset functions, establishing universality for vectorized ODS. SetTransformer (OST) replaces sum-pooling with multi-head attention (PMA), relaxing the sum-decomposition structure while ensuring universal approximation on $\mathbb{R}^{df}$ (Lee et al., 2019; Chien et al., 2022). □

Corollary 9 (Universality of Directed Variants)**.** *The directed variants OmniCopreSheaf and OmniPhaseSheaf, parameterized via OmniDeepSheaf or OmniSheafTransformer over their respective partitioned or complex stalk multisets, are universal approximators of continuous directionality-aware hypergraph multiset functions.*

Proof sketch. For OCS, the domain is the product space of tail and head multisets $\mathcal{M}(V_e^T) \times \mathcal{M}(V_e^H)$. Applying the Stone-Weierstrass theorem on product compact Hausdorff spaces guarantees that sum-decomposable representations over dual multisets universally approximate joint continuous functions (Zaheer et al., 2017; Keriven & Peyré, 2019; Tabaghi & Wang, 2024), directly inheriting universality from Theorem 8. For OPS, the complex field isomorphism $\mathbb{C}^{d\times f} \cong \mathbb{R}^{2df}$ maps complex phase-modulated stalks to real vector spaces of doubled dimension, preserving the topological conditions of Theorem 8 (Ravanbakhsh et al., 2017). □

3.3 LEARNING THE RESTRICTION MAPS

The restriction maps are predicted per incidence from the two respective cells they connect. Let $\bar{\boldsymbol{x}}_u = \frac{1}{d}\sum_i (\boldsymbol{X}_u)_{i,:} \in \mathbb{R}^f$ represent the stalk-averaged node feature and let $\bar{\boldsymbol{h}}_e$ be the corresponding hyperedge feature. Following standard literature (Bodnar et al., 2022) we parameterize the map as:

$$\mathcal{F}_{u \trianglelefteq e} = \Psi\big(\sigma\big(\Phi(\bar{\boldsymbol{x}}_u \,\big\|\, \bar{\boldsymbol{h}}_e)\big)\big), \tag{3}$$

where Φ acts as a single affine layer, $\sigma \in \{\mathrm{sigmoid}, \tanh, \mathrm{id}\}$, and Ψ is a family-specific reshaping operation mapping into $\mathbb{R}^{d\times d}$ to the diagonal group $\mathcal{D}(d)$, the orthogonal group $O(d)$ and the general linear group $GL(d)$ respectively. Following Wang et al. (2023a), we set $\bar{\boldsymbol{h}}_e = \sum_{u\in e} \phi(\bar{\boldsymbol{x}}_u)$ with ϕ being a learned map, moreover averaging over the d stalk rows before applying Φ keeps the predictor's computational cost completely independent of d on its input side. Finally, a fresh predictor is instantiated per layer, ensuring that the maps are dynamically recomputed at every forward pass and remain untied across depth.

3.4 DIRECTIONALITY-AWARE OMNISHEAF

A significant advantage of OS is its inherent extensibility to directed hypergraphs, where each hyperedge $e \in \mathcal{E}$ is partitioned into a tail set $T_e \subset \mathcal{V}$ (source nodes) and a head set $H_e \subset \mathcal{V}$ (destination nodes). We seamlessly introduce directionality awareness through two distinct paradigms: complex-valued phase shifts and explicit topological tail-head indexing, resulting in two formal variants: OmniPhaseSheaf (OPS) and OmniCopreSheaf (OCS).

OmniPhaseSheaf. To natively capture asymmetric data flows within the continuous dynamics of the sheaf, we lift the stalk spaces and feature representations to the complex domain $\mathbb{C}$ (Trabelsi et al., 2018). Specifically, node features $\boldsymbol{X}_u \in \mathbb{C}^{d\times f}$ and hyperedge features $\boldsymbol{H}_e \in \mathbb{C}^{d\times f}$ are modulated by a complex phase that explicitly encodes the node's role within the directed hyperedge (Fiorini et al., 2026). Let $\theta_{u,e}$ denote the topological phase shift assigned to node u based on its directionality within e ($\theta_{u,e} = 0$ if $u \in H_e$ and $\theta_{u,e} = 2\pi q$ if $u \in T_e$, for a magnetic charge q; see Appendix B.4).

While the base ODS variant directly aggregates these phase-shifted signals, our transformer-augmented variant (OST) parameterizes the restriction maps $\mathcal{F}_{u\trianglelefteq e} \in \mathbb{C}^{d\times d}$ and message passing updates using a different attention mechanism (Yang et al., 2020). I.e., the attention coefficients $\alpha_{u,e}$

are computed via the real-part projection of the phase-rotated query-key dot product:

$$\alpha_{u,e} = \frac{\exp\left(\frac{1}{\sqrt{d}}\mathrm{Re}\left(\langle \mathbf{q}_e, s_{u,e}\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u\rangle\right)\right)}{\sum_{v\in e}\exp\left(\frac{1}{\sqrt{d}}\mathrm{Re}\left(\langle \mathbf{q}_e, s_{v,e}\mathcal{F}_{v\trianglelefteq e}\boldsymbol{X}_v\rangle\right)\right)} \cdot s_{u,e}. \tag{4}$$

Then the restriction maps apply this directional phase shift during the projection to the hyperedge stalk:

$$\boldsymbol{H}_e^{(k+1)} = f_{\mathcal{V}\to\mathcal{E}}\left(\{\{\alpha_{u,e}\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u : u \in T_e \cup H_e\}\}\right). \tag{5}$$

This formulation ensures that the orientation of the hyperedge is preserved, hence allowing the network to continuously distinguish between incoming and outgoing signals without structurally bifurcating the message passing pathways. Before classification, complex feature matrices $\boldsymbol{X}_u \in \mathbb{C}^{d\times f}$ are unwound into real representations $[\mathrm{Re}(\boldsymbol{X}_u) \,\|\, \mathrm{Im}(\boldsymbol{X}_u)] \in \mathbb{R}^{d\times 2f}$.

OmniCopreSheaf. Alternatively, we can encode directionality through explicit architectural bifurcation in tail and head indices, directly generalizing the asymmetric principles of Dir-GNN (Rossi et al., 2024) and DHNN (Fiorini et al., 2024) to hypergraph sheaves. We explicitly partition the incidence multiset V_e into two distinct directional multisets:

$$V_e^T = \left\{\{\mathcal{F}^T_{u\trianglelefteq e}\boldsymbol{X}_u : u \in T_e\}\right\}, \quad V_e^H = \left\{\{\mathcal{F}^H_{v\trianglelefteq e}\boldsymbol{X}_v : v \in H_e\}\right\}, \tag{6}$$

where $\mathcal{F}^T_{u\trianglelefteq e}$ and $\mathcal{F}^H_{v\trianglelefteq e}$ are distinct restriction maps learned specifically for the source and destination incidences. The hyperedge update function $f_{\mathcal{V}\to\mathcal{E}}$ is then formalized to aggregate these multisets independently using separate parametric projections:

$$\boldsymbol{H}_e^{(k+1)} = f_{\mathcal{V}\to\mathcal{E}}\left(V_e^{T,(k)}, V_e^{H,(k)}; \boldsymbol{H}_e^{(k)}\right). \tag{7}$$

A symmetric partitioning is applied to E_u to dynamically distinguish incoming messages from hyperedges where u acts as a head versus where u acts as a tail.

Theorem 10. *Dir-GNN (Rossi et al., 2024) on directed graphs and DHNN (Fiorini et al., 2024) on directed hypergraphs are special cases of* OmniCopreSheaf.

Proof sketch. Setting $d = 1$ and trivial restriction maps $\mathcal{F}^T = \mathcal{F}^H = 1$, the partitioned multisets V_e^T and V_e^H compute independent tail and head hyperedge embeddings. Applying distinct linear transforms for incoming versus outgoing hyperedges reproduces the dual projection architecture of DHNN. When applied to 2-uniform directed graphs ($|T_e| = |H_e| = 1$), this collapses identically to Dir-GNN's in/out directional update. □

Theorem 11. *DSNN (Fiorini et al., 2026) on directed graphs and DSHN (Mule et al., 2026) on directed hypergraphs are special cases of* OmniPhaseSheaf.

Proof sketch. Lifting stalks to $\mathbb{C}^d$ and modulating restriction maps by phase factors $s_{u,e} = e^{i\theta_{u,e}} \in \mathbb{U}(1)$ encodes directional orientation in the complex plane. In Stage 1, $f_{\mathcal{V}\to\mathcal{E}}$ computes the mean-pooled representation $\boldsymbol{H}_e = \frac{1}{\delta_e}\sum_{v\in e} s_{v,e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v$. In Stage 2, because $|s_{u,e}|^2 = 1$, the conjugate transport yields $s^*_{u,e}\mathcal{F}^\dagger_{u\trianglelefteq e}(s_{u,e}\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u - \boldsymbol{H}_e) = (\mathcal{F}^\dagger_{u\trianglelefteq e}\mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u - s^*_{u,e}\mathcal{F}^\dagger_{u\trianglelefteq e}\boldsymbol{H}_e$. Aggregating over incident hyperedges without needing δ_e natively reconstructs the Hermitian sheaf Laplacian degree $\boldsymbol{D}_u$ and the complex diffusion updates of DSNN on graphs and DSHN on hypergraphs. □

4 Experimental Results

We evaluate OS across 25 benchmarks covering real-world directed networks, synthetic directional topologies, molecular reaction networks, and standard undirected hypergraphs. All reported metrics represent the mean and standard deviation over 10 standardized random splits; statistical significance against the best baseline is determined via a two-sided Welch's t-test with Holm-Bonferroni correction ($p < 0.05$). Exhaustive results against 15 baselines across all 12 directed benchmarks and detailed cost tables are deferred to Appendices E.2 and F.

4.1 Real-World Directed and Molecular Benchmarks

As reported in Table 1, our directional variants set the best reported accuracy on seven of the twelve directed benchmarks, with the widest margins on the heterophilic ones; on Chameleon, email-Enron, Wisconsin, Amazon-ratings and directed Cora the strongest baseline retains a lead of at most 2.03 points (Tables 7 and 8). On the large heterophilic Roman-empire benchmark, OCS-T reaches 93.35% accuracy (+4.11% over DSHNLight) while DSHN exhausts GPU memory. On email-EU, OPS-T attains 85.56% (+2.89%), whereas classical hypergraph models fail to exceed 68.91%. Moreover on Telegram and Squirrel, with Chameleon remaining within 0.6 points of DSHN. Beyond node classification, OPS-T and OCS-T achieve 91.11% and 90.40% test F1-scores on the Molecular-1 chemical reaction network (Mule et al., 2026), outperforming the best performing baseline DSHNLight (82.32%) by +8.79% with statistical significance.

Table 1: **Evaluation across real-world directed node and molecular hyperedge benchmarks.** Mean test accuracy or F1-score (%) ± standard deviation over 10 random seeds. Colors mark **first**, **second**, and **third**. † denotes statistical significance over the best baseline ($p < 0.05$). OOM = Out-Of-Memory (> 24 GB). Complete results across all 12 directed datasets are in Tables 7 and 8.

	Directed Node Classification						Hyperedge Classification	
Method	**Roman-empire**	**Telegram**	**email-EU**	**Squirrel**	**Chameleon**	**email-Enron**	**Molecular-1 (F1)**	**Molecular-2 (F1)**
UniGCNII	78.89 ± 0.51	51.73 ± 5.05	44.98 ± 2.69	38.28 ± 2.56	39.85 ± 3.19	47.43 ± 7.47	72.00 ± 0.59	85.61 ± 2.63
AllDeepSets	81.79 ± 0.72	49.19 ± 6.73	37.37 ± 6.29	40.69 ± 1.90	42.97 ± 3.60	37.29 ± 7.90	79.17 ± 0.53	85.78 ± 3.01
AllSetTransformer	83.53 ± 0.64	66.92 ± 4.36	38.26 ± 3.57	40.53 ± 1.33	43.85 ± 5.42	63.78 ± 3.66	79.24 ± 1.08	87.89 ± 2.87
ED-HNN	83.82 ± 0.31	60.38 ± 3.86	68.91 ± 4.00	39.85 ± 1.79	44.67 ± 2.33	51.35 ± 6.04	66.37 ± 2.62	87.05 ± 1.96
GeDi-HNN	83.87 ± 0.63	77.12 ± 4.82	52.31 ± 2.84	43.02 ± 3.00	39.29 ± 2.04	50.54 ± 5.80	80.72 ± 0.78	85.64 ± 2.42
DHGNN	77.58 ± 0.54	79.62 ± 5.78	32.35 ± 2.93	39.85 ± 1.79	44.08 ± 4.11	42.16 ± 8.04	OOM	85.93 ± 3.49
SheafHyperGNN	74.50 ± 0.57	70.00 ± 5.32	52.78 ± 9.13	42.01 ± 1.11	41.06 ± 4.94	63.51 ± 5.95	57.99 ± 2.75	80.25 ± 2.20
DSHN	OOM	**88.65 ± 5.54**	78.62 ± 2.50	43.55 ± 2.87	**47.02 ± 4.35**	**75.68 ± 3.42**	OOM	89.09 ± 3.08
DSHNLight	89.24 ± 0.57	81.15 ± 4.19	**82.67 ± 1.29**	44.09 ± 2.36	**46.50 ± 4.09**	**76.76 ± 2.48**	82.32 ± 0.56	86.52 ± 2.68
Ours (OmniSheaf Family)								
OmniDeepSheaf	85.20 ± 0.78	66.67 ± 4.80	64.10 ± 2.02	42.68 ± 1.64	45.02 ± 2.58	64.86 ± 2.21	79.44 ± 0.46	**90.10 ± 1.66**
OmniSheafTransformer	86.61 ± 0.24	67.95 ± 2.40	**80.43 ± 0.76**	44.60 ± 1.17	45.94 ± 4.05	67.57 ± 6.62	82.40 ± 0.44	**90.44 ± 2.21**
OmniPhaseSheaf	88.00 ± 0.06	87.82 ± 3.95	63.43 ± 7.09	42.23 ± 0.16	44.84 ± 3.50	66.67 ± 4.59	73.66 ± 1.14	87.70 ± 1.94
OmniPhaseSheaf-T	**93.08† ± 0.18**	85.26 ± 1.81	**85.56† ± 0.83**	**45.19 ± 1.99**	45.49 ± 1.24	**74.77 ± 5.55**	**91.11† ± 0.30**	**90.53 ± 1.62**
OmniCopreSheaf	**92.61† ± 0.49**	**88.46 ± 1.57**	71.66 ± 0.99	**45.15 ± 2.11**	**46.46 ± 4.76**	60.36 ± 4.59	**86.66† ± 0.28**	88.59 ± 1.83
OmniCopreSheaf-T	**93.35† ± 0.27**	**89.10 ± 4.80**	76.65 ± 2.25	**45.35 ± 0.72**	45.60 ± 4.83	64.86 ± 7.96	**90.40† ± 0.11**	88.62 ± 2.46

4.2 Synthetic Directional Sensitivity Suite

To explicitly isolate the network's capacity to preserve hyperedge orientation under controlled levels of inter-class connectivity, we benchmark on the synthetic directed suite of Fiorini et al. (2026) in Table 2. As the inter-class directed density increases ($I_o = 10 \rightarrow 50$), undirected models suffer degradation, falling to between 19.84% and 32.48% at $I_o = 50$. In contrast, our directional variants (OPS, OPS-T, and OCS) achieve 100.00% accuracy for $I_o \in \{30, 50\}$ with zero standard deviation, outperforming all prior directional and sheaf models with statistical significance. This confirms that complex phase shifts and copresheaf indexing retain multi-way directional alignment without losing class separation.

Table 2: **Accuracy on synthetic directed hypergraphs (Fiorini et al., 2026).** I_o denotes the number of inter-class directed hyperedges per class pair. Colors mark **first**, **second**, and **third**. † denotes statistical significance ($p < 0.05$). Baseline results are taken from Mule et al. (2026) and follow the same experimental protocol.

Method	$I_o = 10$	$I_o = 30$	$I_o = 50$
UniGCNII	21.44 ± 4.33	21.12 ± 2.95	19.84 ± 2.34
AllDeepSets	44.40 ± 6.81	32.32 ± 4.82	31.70 ± 5.92
AllSetTransformer	21.12 ± 3.79	43.68 ± 8.72	31.84 ± 3.31
ED-HNN	34.00 ± 6.05	18.88 ± 2.56	32.48 ± 6.17
SheafHyperGNN	30.64 ± 5.39	27.28 ± 7.31	26.00 ± 9.59
GeDi-HNN	71.44 ± 3.14	71.84 ± 3.31	78.24 ± 5.64
DHGNN	40.72 ± 4.55	51.68 ± 3.97	35.76 ± 3.70
DHGNN (w/ emb.)	84.48 ± 3.22	85.28 ± 3.32	81.12 ± 3.22
DSHN	94.96 ± 1.75	**97.84 ± 1.86**	95.84 ± 2.17
DSHNLight	95.60 ± 2.15	97.04 ± 2.79	**99.04 ± 0.86**
Ours (OmniSheaf Family)			
OmniPhaseSheaf	**97.33 ± 1.51**	**100.00† ± 0.00**	**100.00† ± 0.00**
OmniPhaseSheaf-T	**98.67† ± 1.00**	**100.00† ± 0.00**	**100.00† ± 0.00**
OmniCopreSheaf	**98.67† ± 0.38**	**100.00† ± 0.00**	**100.00† ± 0.00**
OmniCopreSheaf-T	**98.93† ± 0.38**	**99.47 ± 0.38**	**99.73 ± 0.38**
OmniDeepSheaf	72.80 ± 8.84	54.40 ± 6.53	38.93 ± 7.57
OmniSheafTransformer	88.80 ± 2.99	69.60 ± 13.50	86.67 ± 5.89

4.3 Undirected Hypergraph Benchmarks

Table 4 demonstrates that operating directly over polyadic stalks resolves the lossy projections of graph cliques. On political committee hypergraphs characterized by multi-way overlap, ODS and OST achieve gains over prior sheaf models, reaching 83.10% on Senate (+14.37 over SheafHyperGNN) and 78.43% on House (+4.59), both with statistical significance. On the large-scale Congress network (83k hyperedges), OST attains 95.58%, while on the citation benchmarks it stays within 2.3 points of the best baseline without overtaking it, SheafHyperGNN leading on Cora and Cora-CA and ED-HNN on Pubmed and DBLP-CA.

4.4 Computational Scalability, Latency, and Memory Footprint

Table 3 and Figure 2 show the gains of the linear complexity of Theorem 4 as a function of scale. On Roman-empire (22.7k nodes) the clique expansion issue becomes evident: DSHN and SheafHyperGCN already need 18.82 GB and 16.54 GB at $d = 2$ and run out of memory at $d = 3$, and SheafHyperGNN follows at $d = 4$. ODS reaches $d = 6$ in 13.76 GB, and at $d = 2$ it trains in 256.7 ms against 1816.5 ms for DSHN, a 7.1× speedup. Moreover at $d = 6$ a training step costs ODS 1292 GFLOPs against 19,286 for DSHNLight, a 14.9× reduction, because DSHNLight still pays for $\mathcal{O}(\sum_e |e|^2)$ pairwise blocks while ODS pays for $\mathcal{O}(\sum_e |e|)$ incidences. On Telegram the ordering reverses, with only 245 nodes and 183 hyperedges the clique expansion is small in absolute terms, and the cost is instead dominated by the per-incidence MLP that OS evaluates on every stalk. At $d = 6$ DSHNLight trains in 20.2 ms and 0.24 GB against 61.5 ms and 1.69 GB for ODS. The same holds for arithmetic, where ODS needs 135.6 GFLOPs against 4.7. As such OS is therefore the right choice when the hypergraph is large and clique-based diffusion remains preferable on small dense instances.

Table 3: **Computational cost on an NVIDIA L4 GPU (24 GB)** (width 256, 3 layers, Diag restriction maps). Contrasting a large sparse hypergraph (Roman-empire, 22.7k nodes) with a small dense one (Telegram, 245 nodes, $\bar{\delta}_e = 49.7$). Colors mark **first**, **second**, and **third** lowest resource consumption among stalk-based methods. OOM indicates out-of-memory (> 24 GB).

	Roman-empire (Large Scale: $\lvert\mathcal{V}\rvert = 22,662$)				Telegram (Dense Polyadic: $\bar{\delta}_e = 49.7$)			
Method	**VRAM** ($d = 2$)	**Time** ($d = 2$)	**VRAM** ($d = 6$)	**Time** ($d = 6$)	**VRAM** ($d = 6$)	**Time** ($d = 6$)	**Infer** ($d = 6$)	**GFLOP** ($d = 6$)
SheafHyperGNN	9.28 GB	549.8 ms	OOM	OOM	1.41 GB	57.8 ms	18.2 ms	5.9
SheafHyperGCN	16.54 GB	1004.8 ms	OOM	OOM	0.61 GB	56.7 ms	53.9 ms	0.5
DSHN	18.82 GB	1816.5 ms	OOM	OOM	1.76 GB	37.5 ms	10.1 ms	7.3
DSHNLight	2.99 GB	286.5 ms	7.95 GB	835.5 ms	0.24 GB	20.2 ms	10.7 ms	4.7
Ours (OmniSheaf Family)								
OmniDeepSheaf	5.20 GB	256.7 ms	13.76 GB	600.3 ms	1.69 GB	61.5 ms	20.8 ms	135.6
OmniSheafTransformer	5.34 GB	373.1 ms	14.45 GB	1683.9 ms	2.18 GB	199.9 ms	70.7 ms	1567.0
OmniCopreSheaf	9.46 GB	517.3 ms	21.27 GB	1058.7 ms	2.26 GB	102.9 ms	31.9 ms	144.9
OmniCopreSheaf-T	9.79 GB	639.5 ms	OOM	OOM	2.93 GB	256.4 ms	82.6 ms	1582.7
OmniPhaseSheaf	10.96 GB	970.4 ms	OOM	OOM	3.58 GB	267.9 ms	64.2 ms	267.4
OmniPhaseSheaf-T	16.62 GB	1906.0 ms	OOM	OOM	5.07 GB	622.0 ms	183.0 ms	3130.1

Table 4: **Classification test accuracy (%) on the undirected hypergraph benchmark of Duta et al. (2023).** Mean ± std over 10 random seeds. Colors mark **first**, **second**, and **third** results. † denotes statistical significance ($p < 0.05$).

Method	**Senate**	**House**	**Congress**	**Cora**	**Citeseer**	**Pubmed**	**Cora-CA**	**DBLP-CA**
HCHA	48.62 ± 4.41	61.36 ± 2.53	90.43 ± 1.20	79.14 ± 1.02	72.42 ± 1.42	86.41 ± 0.36	82.55 ± 0.97	90.92 ± 0.22
HNHN	50.93 ± 6.33	67.80 ± 2.59	53.35 ± 1.45	76.36 ± 1.92	72.64 ± 1.57	86.90 ± 0.30	77.19 ± 1.49	86.78 ± 0.29
UniGCNII	49.30 ± 4.25	67.25 ± 2.57	94.81 ± 0.81	78.81 ± 1.05	73.05 ± 2.21	88.25 ± 0.33	83.60 ± 1.14	91.69 ± 0.19
ED-HNN	64.79 ± 5.14	72.45 ± 2.28	95.00 ± 0.99	80.31 ± 1.35	73.70 ± 1.38	89.03 ± 0.53	83.97 ± 1.55	91.90 ± 0.19
AllDeepSets	48.17 ± 5.67	67.82 ± 2.40	91.80 ± 1.53	76.88 ± 1.80	70.83 ± 1.63	88.75 ± 0.33	81.97 ± 1.50	91.27 ± 0.27
AllSetTransformer	51.83 ± 5.22	69.33 ± 2.20	92.16 ± 1.05	78.58 ± 1.47	73.08 ± 1.20	88.72 ± 0.37	83.63 ± 1.47	91.53 ± 0.23
SheafHyperGCN	66.33 ± 4.58	72.66 ± 2.26	90.37 ± 1.52	80.06 ± 1.12	73.27 ± 0.50	87.09 ± 0.71	83.26 ± 1.20	90.83 ± 0.23
SheafHyperGNN	68.73 ± 4.68	73.84 ± 2.30	91.81 ± 1.60	81.30 ± 1.70	74.71 ± 1.23	87.68 ± 0.60	85.52 ± 1.28	91.59 ± 0.24
Ours (OmniSheaf Family)								
OmniDeepSheaf	83.10† ± 1.99	78.33† ± 1.26	94.73 ± 0.58	78.73 ± 1.15	74.72 ± 0.92	87.86 ± 0.13	83.26 ± 0.62	91.64 ± 0.16
OmniSheafTransformer	81.69† ± 2.30	78.43† ± 1.17	95.58 ± 1.16	79.76 ± 1.81	74.52 ± 0.65	87.82 ± 0.35	83.21 ± 1.00	91.70 ± 0.21

5 Conclusions

We presented OS, a new way of doing cellular sheaf learning on hypergraphs without clique expansions. By parameterizing incidence message passing via multiset functions over stalks, OS reduces complexity from $\mathcal{O}(\sum |e|^2)$ to $\mathcal{O}(\sum |e|)$. We proved that its instantiations (ODS, OST) are universal approximators that subsume sheaf diffusion, as well as directed architectures via phase modulation (OPS) and asymmetric copresheaves (OCS). Across 25 benchmarks, OS sets state-of-the-art results on directed, heterophilic, and molecular tasks, unlocking higher stalk dimensions ($d = 6$) with up to 7.1× faster training and 3.6× less memory. However, while effective, restriction maps currently assume static incidences; extending stalk multisets to continuous-time dynamic hypergraphs (Bamberger et al., 2025) and simplicial complexes (Bodnar et al., 2021) represents a promising direction.

ACKNOWLEDGEMENTS

This work was supported in part by NSF under Grants CNS-2312875 and OAC-2530896, DARPA under Cooperative Agreement D25AC00374-00, ONR under Award N00014-23-1-2221, and AFOSR under Award FA9550-23-1-0261.

REFERENCES

Sameer Agarwal, Kristin Branson, and Serge Belongie. Higher order learning with graphs. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pp. 17–24, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933832. doi: 10.1145/1143844.1143847. URL https://doi.org/10.1145/1143844.1143847.

Song Bai, Feihu Zhang, and Philip H.S. Torr. Hypergraph convolution and hypergraph attention. *Pattern Recognition*, 110:107637, 2021. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2020.107637. URL https://www.sciencedirect.com/science/article/pii/S0031320320304404.

Jacob Bamberger, Federico Barbero, Xiaowen Dong, and Michael M. Bronstein. Bundle neural network for message diffusion on graphs. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=scI9307PLG.

Federico Barbero, Cristian Bodnar, Haitz Sáez de Ocáriz Borde, and Pietro Lió. Sheaf attention networks. In *NeurIPS 2022 Workshop on Symmetry and Geometry in Neural Representations*, 2022. URL https://openreview.net/forum?id=LIDvgVjpkZr.

Cristian Bodnar, Fabrizio Frasca, Yuguang Wang, Nina Otter, Guido F Montufar, Pietro Lió, and Michael Bronstein. Weisfeiler and lehman go topological: Message passing simplicial networks. 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. 1026–1037. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/bodnar21a.html.

Cristian Bodnar, Francesco Di Giovanni, Benjamin Paul Chamberlain, Pietro Liò, and Michael M. Bronstein. Neural sheaf diffusion: A topological perspective on heterophily and oversmoothing in GNNs. In *Advances in Neural Information Processing Systems*, 2022. URL https://papers.neurips.cc/paper_files/paper/2022/file/75c45fca2aa416ada062b26cc4fb7641-Paper-Conference.pdf.

Guanzi Chen, Jiying Zhang, Xi Xiao, and Yang Li. Preventing over-smoothing for hypergraph neural networks, 2022. URL https://arxiv.org/abs/2203.17159.

Eli Chien, Chao Pan, Jianhao Peng, and Olgica Milenkovic. You are AllSet: A multiset function framework for hypergraph neural networks. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=hpBTIv2uy_E.

Justin Michael Curry. *Sheaves, Cosheaves and Applications*. PhD thesis, University of Pennsylvania, 2014. URL https://www2.math.upenn.edu/grad/dissertations/CurryThesis.pdf.

Yihe Dong, Will Sawin, and Yoshua Bengio. HNHN: Hypergraph networks with hyperedge neurons. In *ICML Graph Representation Learning and Beyond (GRL+) Workshop*, 2020. URL https://grlplus.github.io/papers/40.pdf.

Iulia Duta, Giulia Cassarà, Fabrizio Silvestri, and Pietro Liò. Sheaf hypergraph networks. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=NvcVXzJvhX.

Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. Hypergraph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3558–3565, 2019. doi: 10.1609/aaai.v33i01.33013558. URL https://doi.org/10.1609/aaai.v33i01.33013558.

Stefano Fiorini, Stefano Coniglio, Michele Ciavotta, and Alessio Del Bue. Let there be direction in hypergraph neural networks. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=h48Ri6pmvi.

Stefano Fiorini, Hakan Aktas, Iulia Duta, Pietro Morerio, Alessio Del Bue, Pietro Lió, and Stefano Coniglio. Sheaves reloaded: A direction awakening. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=iDiiETH7Qv.

M. Gori, G. Monfardini, and F. Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pp. 729–734 vol. 2, 2005. doi: 10.1109/IJCNN.2005.1555942. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1555942.

Tingting Guo, Yining Zhang, Yanfang Xue, Lishan Qiao, and Dinggang Shen. Brain function network: Higher order vs. more discrimination. *Frontiers in Neuroscience*, Volume 15 - 2021, 2021. ISSN 1662-453X. doi: 10.3389/fnins.2021.696639. URL https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2021.696639.

Jakob Hansen and Thomas Gebhart. Sheaf neural networks. In *NeurIPS Workshop on TDA & Beyond*, 2020. URL https://openreview.net/forum?id=GgcgIJsT8HD.

Jakob Hansen and Robert Ghrist. Opinion dynamics on discourse sheaves. *SIAM Journal on Applied Mathematics*, 81(5):2033–2060, 2021. doi: 10.1137/20M1341088. URL https://epubs.siam.org/doi/10.1137/20M1341088.

Jing Huang and Jie Yang. UniGNN: a unified framework for graph and hypergraph neural networks. In Zhi-Hua Zhou (ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 2563–2569. International Joint Conferences on Artificial Intelligence Organization, 8 2021. doi: 10.24963/ijcai.2021/353. URL https://doi.org/10.24963/ijcai.2021/353. Main Track.

Jianwen Jiang, Yuxuan Wei, Yifan Feng, Jingxuan Cao, and Yue Gao. Dynamic hypergraph neural networks. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pp. 2635–2641. International Joint Conferences on Artificial Intelligence Organization, 7 2019. doi: 10.24963/ijcai.2019/366. URL https://doi.org/10.24963/ijcai.2019/366.

Nicolas Keriven and Gabriel Peyré. Universal invariant and equivariant graph neural networks. In *Advances in Neural Information Processing Systems*, Red Hook, NY, USA, 2019. Curran Associates Inc. doi: 10.5555/3454287.3454924. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/ea9268cb43f55d1d12380fb6ea5bf572-Paper.pdf.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6980.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJU4ayYgl.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set Transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning*, pp. 3744–3753. PMLR, 2019. URL https://proceedings.mlr.press/v97/lee19d/lee19d.pdf.

Shengjie Liu, Jia Guo, and Chenyang Yang. Multidimensional graph neural networks for wireless communications. *Trans. Wireless. Comm.*, 23(4):3057–3073, April 2024. ISSN 1536-1276. doi: 10.1109/TWC.2023.3305124. URL https://doi.org/10.1109/TWC.2023.3305124.

Alexander Mathiasen, Frederik Hvilshøj, Jakob Rødsgaard Jørgensen, Anshul Nasery, and Davide Mottin. What if neural networks had SVDs? In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546. doi: 10.5555/3495724.3497270. URL https://papers.neurips.cc/paper_files/paper/2020/file/d61e4bbd6393c9111e6526ea173a7c8b-Paper.pdf.

Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: higher-order graph neural networks. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, AAAI'19. AAAI Press, 2019. ISBN 978-1-57735-809-1. doi: 10.1609/aaai.v33i01.33014602. URL https://doi.org/10.1609/aaai.v33i01.33014602.

Emanuele Mule, Stefano Fiorini, Antonio Purificato, Federico Siciliano, Stefano Coniglio, and Fabrizio Silvestri. Directional sheaf hypergraph networks: Unifying learning on directed and undirected hypergraphs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=d1hQkcDy7c.

Konstantin Prokopchik, Austin R Benson, and Francesco Tudisco. Nonlinear feature diffusion on hypergraphs. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17945–17958. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/prokopchik22a.html.

Siamak Ravanbakhsh, Jeff Schneider, and Barnabás Póczos. Equivariance through parameter-sharing. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2892–2901. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/ravanbakhsh17a.html.

Emanuele Rossi, Bertrand Charpentier, Francesco Di Giovanni, Fabrizio Frasca, Stephan Günnemann, and Michael M. Bronstein. Edge directionality improves learning on heterophilic graphs. In Soledad Villar and Benjamin Chamberlain (eds.), *Proceedings of the Second Learning on Graphs Conference*, volume 231 of *Proceedings of Machine Learning Research*, pp. 25:1–25:27. PMLR, 27–30 Nov 2024. URL https://proceedings.mlr.press/v231/rossi24a.html.

T. Konstantin Rusch, Michael M. Bronstein, and Siddhartha Mishra. A survey on oversmoothing in graph neural networks, 2023. URL https://arxiv.org/abs/2303.10993.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009. doi: 10.1109/TNN.2008.2005605. URL https://dl.acm.org/doi/10.1109/TNN.2008.2005605.

Puoya Tabaghi and Yusu Wang. Universal representation of permutation-invariant functions on vectors and tensors. In Claire Vernade and Daniel Hsu (eds.), *Proceedings of The 35th International Conference on Algorithmic Learning Theory*, volume 237 of *Proceedings of Machine Learning Research*, pp. 1134–1187. PMLR, 25–28 Feb 2024. URL https://proceedings.mlr.press/v237/tabaghi24a.html.

Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, Joao Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J Pal. Deep complex networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1T2hmZAb.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Edward Wagstaff, Fabian Fuchs, Martin Engelcke, Ingmar Posner, and Michael A. Osborne. On the limitations of representing functions on sets. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of

Proceedings of Machine Learning Research, pp. 6487–6494. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/wagstaff19a.html.

Peihao Wang, Shenghao Yang, Yunyu Liu, Zhangyang Wang, and Pan Li. Equivariant hypergraph diffusion neural operators. In *The Eleventh International Conference on Learning Representations*, 2023a. URL https://openreview.net/forum?id=RiTjKoscnNd.

Yuxin Wang, Quan Gan, Xipeng Qiu, Xuanjing Huang, and David Wipf. From hypergraph energy functions to hypergraph neural networks. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 35605–35623. PMLR, 23–29 Jul 2023b. URL https://proceedings.mlr.press/v202/wang23d.html.

Naganand Yadati, Madhav Nimishakavi, Prateek Yadav, Vikram Nitin, Anand Louis, and Partha Talukdar. HyperGCN: A new method for training graph convolutional networks on hypergraphs. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/1efa39bcaec6f3900149160693694536-Paper.pdf.

Chaoqi Yang, Ruijie Wang, Shuochao Yao, and Tarek Abdelzaher. Semi-supervised hypergraph node classification on hypergraph line expansion. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, pp. 2352–2361, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392365. doi: 10.1145/3511808.3557447. URL https://doi.org/10.1145/3511808.3557447.

Muqiao Yang, Martin Q. Ma, Dongyu Li, Yao-Hung Hubert Tsai, and Ruslan Salakhutdinov. Complex transformer: A framework for modeling complex-valued sequence. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4232–4236, 2020. doi: 10.1109/ICASSP40776.2020.9054008. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9054008.

Olga Zaghen, Antonio Longa, Steve Azzolin, Lev Telyatnikov, Andrea Passerini, and Pietro Lió. Sheaf diffusion goes nonlinear: Enhancing GNNs with adaptive sheaf laplacians. In *ICML 2024 Workshop on Geometry-grounded Representation Learning and Generative Modeling*, 2024. URL https://openreview.net/forum?id=MGQtGV5gPO.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/f22e4747da1aa27e363d86d40ff442fe-Paper.pdf.

Hongliang Zhang, Lingyang Song, and Zhu Han. Radio resource allocation for device-to-device underlay communication using hypergraph theory. *IEEE Transactions on Wireless Communications*, 15(7):4852–4861, 2016. doi: 10.1109/TWC.2016.2547862. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7442907.

OmniSheaf: Unifying Sheaf Hypergraph Networks via Efficient Universal Approximators

Supplementary Material

CONTENTS

A Proofs **15**

A.1 Proofs of Symmetries and Gauge Equivariance 15

A.2 Proof of Complexity 16

A.3 Proofs of Generalization over Undirected Models 17

A.4 Proofs of Universal Approximators 19

A.5 Proofs of Generalization over Directed Models 20

B Architectural and Implementation Details **22**

B.1 OmniSheaf Message Passing Pipeline 22

B.2 Universal Approximator Instantiations 23

B.3 Restriction Map Builders and Parameterization Families 23

B.4 Directional OmniSheaf Variants 24

B.5 Numerical Reduction Validation 25

B.6 Hyperedge Classification Readout 25

C Experimental Setup and Hardware Infrastructure **25**

C.1 Hardware Infrastructure and Compute Environment 25

C.2 Dataset Descriptions and Preprocessing Protocols 25

D Hyperparameter Search and Tuning Details **27**

D.1 Hyperparameter Search Space 27

D.2 Budget Allocation and Memory Capping Rules 27

D.3 Two-Stage Selection and 10-Split Confirmation Protocol 27

D.4 Statistical Significance Testing 28

E Extended Experimental Results and Cost Profiling **28**

E.1 Hyperparameter Sensitivity and Qualitative Findings 28

E.2 Computational Efficiency, Peak Memory, and Scaling 28

E.3 GFLOPs and Arithmetic Complexity Analysis 28

F Full Results **30**

G Hyperparameter Ablations **63**

A PROOFS

A.1 PROOFS OF SYMMETRIES AND GAUGE EQUIVARIANCE

Proposition 2 (Permutation Symmetries). *Let $f_{\mathcal{V}\to\mathcal{E}}$ and $f_{\mathcal{E}\to\mathcal{V}}$ be permutation-invariant multiset functions. Then,* OS *updates (Definition 1) are invariant to the local ordering of incident cells within V_e and E_u, and globally permutation-equivariant with respect to joint node and hyperedge reorderings.*

Proof. **(Local Invariance).** By Definition 1, $V_e = \{\{\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u : u \in e\}\}$ is a multiset whose elements are unordered. For any bijection $\pi_e \colon e \to e$, the underlying multiset satisfies $\pi_e(V_e) = V_e$. Since $f_{\mathcal{V}\to\mathcal{E}}$ is permutation-invariant with respect to its first argument, we have:

$$f_{\mathcal{V}\to\mathcal{E}}\big(\pi_e(V_e); \boldsymbol{H}_e\big) = f_{\mathcal{V}\to\mathcal{E}}\big(V_e; \boldsymbol{H}_e\big) = \boldsymbol{H}_e^{(k+1)}. \tag{8}$$

Identical reasoning applies to $E_u = \big\{\!\big\{(\mathcal{F}_{u\trianglelefteq e}^\top \boldsymbol{H}_e, \mathcal{F}_{u\trianglelefteq e}^\top \mathcal{F}_{u\trianglelefteq e}) : e \ni u\big\}\!\big\}$ under any local reordering of incident hyperedges $\pi_u \colon \{e : e \ni u\} \to \{e : e \ni u\}$, ensuring local permutation invariance.

(Global Equivariance). Let $\boldsymbol{P}_{\mathcal{V}} \in \{0,1\}^{|\mathcal{V}|\times|\mathcal{V}|}$ and $\boldsymbol{P}_{\mathcal{E}} \in \{0,1\}^{|\mathcal{E}|\times|\mathcal{E}|}$ be arbitrary global permutation matrices defining bijections $\sigma_{\mathcal{V}} \colon \mathcal{V} \to \mathcal{V}$ and $\sigma_{\mathcal{E}} \colon \mathcal{E} \to \mathcal{E}$. Under relabeling, let the permuted cell features be $\tilde{\boldsymbol{X}}_{\sigma_{\mathcal{V}}(u)} = \boldsymbol{X}_u$ and $\tilde{\boldsymbol{H}}_{\sigma_{\mathcal{E}}(e)} = \boldsymbol{H}_e$, with transformed restriction maps $\tilde{\mathcal{F}}_{\sigma_{\mathcal{V}}(u)\trianglelefteq\sigma_{\mathcal{E}}(e)} = \mathcal{F}_{u\trianglelefteq e}$. The incident multiset for hyperedge $\sigma_{\mathcal{E}}(e)$ evaluates to:

$$\begin{aligned}\tilde{V}_{\sigma_{\mathcal{E}}(e)} &= \Big\{\!\Big\{\tilde{\mathcal{F}}_{v'\trianglelefteq\sigma_{\mathcal{E}}(e)}\tilde{\boldsymbol{X}}_{v'} : v' \in \sigma_{\mathcal{E}}(e)\Big\}\!\Big\} = \Big\{\!\Big\{\tilde{\mathcal{F}}_{\sigma_{\mathcal{V}}(u)\trianglelefteq\sigma_{\mathcal{E}}(e)}\tilde{\boldsymbol{X}}_{\sigma_{\mathcal{V}}(u)} : u \in e\Big\}\!\Big\} \\ &= \{\{\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u : u \in e\}\} = V_e.\end{aligned} \tag{9}$$

By local invariance of $f_{\mathcal{V}\to\mathcal{E}}$, the output satisfies $\tilde{\boldsymbol{H}}_{\sigma_{\mathcal{E}}(e)}^{(k+1)} = f_{\mathcal{V}\to\mathcal{E}}(\tilde{V}_{\sigma_{\mathcal{E}}(e)}; \tilde{\boldsymbol{H}}_{\sigma_{\mathcal{E}}(e)}) = f_{\mathcal{V}\to\mathcal{E}}(V_e; \boldsymbol{H}_e) = \boldsymbol{H}_e^{(k+1)}$, meaning $\tilde{\boldsymbol{H}}^{(k+1)} = \boldsymbol{P}_{\mathcal{E}}\boldsymbol{H}^{(k+1)}$. By symmetry, the node update satisfies $\tilde{\boldsymbol{X}}_{\sigma_{\mathcal{V}}(u)}^{(k+1)} = \boldsymbol{X}_u^{(k+1)}$, which yields $\tilde{\boldsymbol{X}}^{(k+1)} = \boldsymbol{P}_{\mathcal{V}}\boldsymbol{X}^{(k+1)}$. Hence, OS is globally permutation equivariant. □

Proposition 3 (Sheaf Gauge Equivariance). *Let $\mathcal{Q} = \{\boldsymbol{Q}_u \in \mathrm{O}(d) : u \in \mathcal{V}\} \cup \{\boldsymbol{Q}_e \in \mathrm{O}(d) : e \in \mathcal{E}\}$ be an orthogonal gauge transformation acting on cell stalks. Under the coordinate changes $\boldsymbol{X}_u \mapsto \boldsymbol{Q}_u\boldsymbol{X}_u$, $\boldsymbol{H}_e \mapsto \boldsymbol{Q}_e\boldsymbol{H}_e$, and $\mathcal{F}_{u\trianglelefteq e} \mapsto \boldsymbol{Q}_e\mathcal{F}_{u\trianglelefteq e}\boldsymbol{Q}_u^\top$, the transformed multiset elements satisfy $\tilde{\mathcal{F}}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u = \boldsymbol{Q}_e(\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u)$ and $\tilde{\mathcal{F}}_{u\trianglelefteq e}^\top\tilde{\boldsymbol{H}}_e = \boldsymbol{Q}_u(\mathcal{F}_{u\trianglelefteq e}^\top\boldsymbol{H}_e)$. Consequently, whenever $f_{\mathcal{V}\to\mathcal{E}}$ and $f_{\mathcal{E}\to\mathcal{V}}$ are stalk-orthogonal equivariant (as satisfied by linear sheaf diffusion), the* OS *forward pass is gauge equivariant:*

$$\boldsymbol{H}_e^{(k+1)} \mapsto \boldsymbol{Q}_e\boldsymbol{H}_e^{(k+1)} \quad \textit{and} \quad \boldsymbol{X}_u^{(k+1)} \mapsto \boldsymbol{Q}_u\boldsymbol{X}_u^{(k+1)}. \tag{2}$$

For OmniPhaseSheaf, the result extends to $\mathrm{U}(d)$ gauge transformations in the complex stalk bundle. For non-linear row-wise neural parameterizations, equivariance is preserved over the stalk coordinate permutation subgroup $\mathcal{P}_d \leq \mathrm{O}(d)$ (and signed permutations for odd activations).

Proof. Let $\boldsymbol{Q}_u \in \mathrm{O}(d)$ and $\boldsymbol{Q}_e \in \mathrm{O}(d)$ be orthogonal matrices representing local changes of basis in stalks $\mathcal{F}(u)$ and $\mathcal{F}(e)$, satisfying $\boldsymbol{Q}_u^\top\boldsymbol{Q}_u = \boldsymbol{I}_d$ and $\boldsymbol{Q}_e^\top\boldsymbol{Q}_e = \boldsymbol{I}_d$. The transformed cell states and restriction maps are given by:

$$\tilde{\boldsymbol{X}}_u = \boldsymbol{Q}_u\boldsymbol{X}_u, \quad \tilde{\boldsymbol{H}}_e = \boldsymbol{Q}_e\boldsymbol{H}_e, \quad \tilde{\mathcal{F}}_{u\trianglelefteq e} = \boldsymbol{Q}_e\mathcal{F}_{u\trianglelefteq e}\boldsymbol{Q}_u^\top. \tag{10}$$

For stage one, each transported representation in multiset V_e transforms as:

$$\tilde{\mathcal{F}}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u = (\boldsymbol{Q}_e\mathcal{F}_{u\trianglelefteq e}\boldsymbol{Q}_u^\top)(\boldsymbol{Q}_u\boldsymbol{X}_u) = \boldsymbol{Q}_e\mathcal{F}_{u\trianglelefteq e}(\boldsymbol{Q}_u^\top\boldsymbol{Q}_u)\boldsymbol{X}_u = \boldsymbol{Q}_e(\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u). \tag{11}$$

Hence, the transformed incident multiset is $\tilde{V}_e = \{\{\boldsymbol{Q}_e(\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u) : u \in e\}\} = \boldsymbol{Q}_e V_e$.

For stage two, the transported features and local transport blocks in E_u transform as:

$$\tilde{\mathcal{F}}_{u\trianglelefteq e}^\top\tilde{\boldsymbol{H}}_e = (\boldsymbol{Q}_e\mathcal{F}_{u\trianglelefteq e}\boldsymbol{Q}_u^\top)^\top(\boldsymbol{Q}_e\boldsymbol{H}_e) = \boldsymbol{Q}_u\mathcal{F}_{u\trianglelefteq e}^\top(\boldsymbol{Q}_e^\top\boldsymbol{Q}_e)\boldsymbol{H}_e = \boldsymbol{Q}_u(\mathcal{F}_{u\trianglelefteq e}^\top\boldsymbol{H}_e), \tag{12}$$

$$\tilde{\mathcal{F}}_{u\trianglelefteq e}^\top\tilde{\mathcal{F}}_{u\trianglelefteq e} = (\boldsymbol{Q}_u\mathcal{F}_{u\trianglelefteq e}^\top\boldsymbol{Q}_e^\top)(\boldsymbol{Q}_e\mathcal{F}_{u\trianglelefteq e}\boldsymbol{Q}_u^\top) = \boldsymbol{Q}_u(\mathcal{F}_{u\trianglelefteq e}^\top\mathcal{F}_{u\trianglelefteq e})\boldsymbol{Q}_u^\top. \tag{13}$$

Summing the transport blocks yields the transformed sheaf degree block:

$$\tilde{\boldsymbol{D}}_u = \sum_{e \ni u} \tilde{\mathcal{F}}_{u \trianglelefteq e}^\top \tilde{\mathcal{F}}_{u \trianglelefteq e} = \boldsymbol{Q}_u \left(\sum_{e \ni u} \mathcal{F}_{u \trianglelefteq e}^\top \mathcal{F}_{u \trianglelefteq e} \right) \boldsymbol{Q}_u^\top = \boldsymbol{Q}_u \boldsymbol{D}_u \boldsymbol{Q}_u^\top. \tag{14}$$

Assuming non-degeneracy $\lambda_{\min}(\boldsymbol{D}_u) > 0$ (or with standard self-loop regularization $\boldsymbol{D}_u + \epsilon \boldsymbol{I}_d$), the inverse square root satisfies $\tilde{\boldsymbol{D}}_u^{-1/2} = \boldsymbol{Q}_u \boldsymbol{D}_u^{-1/2} \boldsymbol{Q}_u^\top$.

1. Orthogonal Gauge Equivariance of Linear Sheaf Message Passing. Whenever the multiset functions $f_{\mathcal{V} \to \mathcal{E}}$ and $f_{\mathcal{E} \to \mathcal{V}}$ are stalk-orthogonal equivariant (as holds for multiset summation and the linear sheaf diffusion operators subsumed in Theorems 6, 7 and 11), the orthogonal operators factor out linearly:

$$\tilde{\boldsymbol{H}}_e^{(k+1)} = \sum_{u \in e} \tilde{\mathcal{F}}_{u \trianglelefteq e} \tilde{\boldsymbol{X}}_u = \boldsymbol{Q}_e \sum_{u \in e} (\mathcal{F}_{u \trianglelefteq e} \boldsymbol{X}_u) = \boldsymbol{Q}_e \boldsymbol{H}_e^{(k+1)}. \tag{15}$$

Similarly, for normalized diffusion in stage two:

$$\tilde{\boldsymbol{D}}_u^{-1/2} \sum_{e \ni u} \tilde{\mathcal{F}}_{u \trianglelefteq e}^\top \tilde{\boldsymbol{H}}_e^{(k+1)} = (\boldsymbol{Q}_u \boldsymbol{D}_u^{-1/2} \boldsymbol{Q}_u^\top) \boldsymbol{Q}_u \sum_{e \ni u} (\mathcal{F}_{u \trianglelefteq e}^\top \boldsymbol{H}_e^{(k+1)}) \tag{16}$$

$$= \boldsymbol{Q}_u \left(\boldsymbol{D}_u^{-1/2} \sum_{e \ni u} \mathcal{F}_{u \trianglelefteq e}^\top \boldsymbol{H}_e^{(k+1)} \right), \tag{17}$$

$$\tilde{\boldsymbol{D}}_u^{1/2} \tilde{\boldsymbol{X}}_u^{(k)} = (\boldsymbol{Q}_u \boldsymbol{D}_u^{1/2} \boldsymbol{Q}_u^\top)(\boldsymbol{Q}_u \boldsymbol{X}_u^{(k)}) = \boldsymbol{Q}_u (\boldsymbol{D}_u^{1/2} \boldsymbol{X}_u^{(k)}). \tag{18}$$

Thus, $\tilde{\boldsymbol{X}}_u^{(k+1)} = \boldsymbol{Q}_u \boldsymbol{X}_u^{(k+1)}$, proving that linear sheaf hypergraph diffusion is $\mathrm{O}(d)$-gauge equivariant. For OmniPhaseSheaf, the identical algebra holds under the unitary gauge group $\mathrm{U}(d)$ by replacing transposes with Hermitian conjugate transposes $\dagger$, since $\boldsymbol{Q}_u^\dagger \boldsymbol{Q}_u = \boldsymbol{I}_d$ and $\boldsymbol{Q}_e^\dagger \boldsymbol{Q}_e = \boldsymbol{I}_d$.

2. Gauge Equivariance of Row-Wise Neural Parameterizations. When $f_{\mathcal{V} \to \mathcal{E}}$ and $f_{\mathcal{E} \to \mathcal{V}}$ are parameterized by row-wise multi-layer perceptrons $\phi[\boldsymbol{A}]$ with coordinate-wise non-linearities (e.g., ReLU), a general orthogonal matrix $\boldsymbol{Q} \in \mathrm{O}(d)$ mixes stalk coordinates, which does not commute with non-linear activation functions ($\phi(\sum_j Q_{ij} \boldsymbol{A}_{j,:}) \neq \sum_j Q_{ij} \phi(\boldsymbol{A}_{j,:})$). However, let $\boldsymbol{Q} = \boldsymbol{P} \in \mathcal{P}_d \leq \mathrm{O}(d)$ be a coordinate permutation matrix representing a relabeling of stalk basis axes. Then $(\boldsymbol{P}\boldsymbol{A})_{i,:} = \boldsymbol{A}_{\pi(i),:}$, and row-wise evaluation satisfies:

$$(\phi[\boldsymbol{P}\boldsymbol{A}])_{i,:} = \phi(\boldsymbol{A}_{\pi(i),:}) = (\boldsymbol{P}\phi[\boldsymbol{A}])_{i,:}. \tag{19}$$

Furthermore, for odd activations without bias ($\phi(-x) = -\phi(x)$), row-wise transformations additionally commute with sign flips, establishing equivariance under the signed permutation (hyperoctahedral) group $B_d = \{\pm 1\}^d \rtimes \mathcal{P}_d \leq \mathrm{O}(d)$. Consequently, row-wise neural parameterizations preserve gauge equivariance with respect to stalk coordinate permutations $\mathcal{P}_d$ (and signed permutations for odd activations). □

A.2 Proof of Complexity

Theorem 4 (Time and Space Complexity Reduction)**.** OmniSheaf *reduces the space and time complexity of sheaf hypergraph message passing from* $\mathcal{O}(\sum_{e \in \mathcal{E}} |e|^2)$ *to* $\mathcal{O}(\sum_{e \in \mathcal{E}} |e|)$.

Proof. Previous sheaf hypergraph methods, such as SheafHyperGNN (Duta et al., 2023) and DSHN (Mule et al., 2026), rely on the graph clique expansion of the hypergraph. For each hyperedge $e \in \mathcal{E}$, the clique expansion creates pairwise connections between all constituent nodes $u, v \in e$. This requires storing and computing messages for $|e|(|e|-1)/2$ edges per hyperedge. Thus, the time and memory complexity for message passing is strictly bounded by $\mathcal{O}(\sum_{e \in \mathcal{E}} |e|^2)$.

In contrast, OS leverages multiset functions $f_{\mathcal{V} \to \mathcal{E}}$ and $f_{\mathcal{E} \to \mathcal{V}}$ operating directly over the bipartite incidence graph. During the first stage, each node $u \in e$ sends exactly one projected message to the hyperedge e. In the second stage, the hyperedge e sends exactly one message back to each $u \in e$. The total number of message passing operations per hyperedge scales linearly with the size of the hyperedge, requiring $\mathcal{O}(|e|)$ operations. Consequently, summing over all hyperedges, the overall space and time complexity is reduced to $\mathcal{O}(\sum_{e \in \mathcal{E}} |e|)$, entirely bypassing the quadratic bottleneck of the clique expansion. □

A.3 Proofs of Generalization over Undirected Models

Theorem 5. AllSet *(Chien et al., 2022) is a special case of* OmniSheaf.

Proof. Let $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ be a hypergraph with node features $\boldsymbol{x}_u \in \mathbb{R}^f$. The AllSet update rule is defined as:

$$\boldsymbol{h}_e^{(k+1)} = f_{\mathcal{V}\to\mathcal{E}}\left(\left\{\!\left\{\boldsymbol{x}_u^{(k)} : u \in e\right\}\!\right\}\right),$$
$$\boldsymbol{x}_u^{(k+1)} = f_{\mathcal{E}\to\mathcal{V}}\left(\left\{\!\left\{\boldsymbol{h}_e^{(k+1)} : e \ni u\right\}\!\right\}\right).$$

We define a trivial cellular sheaf $\mathcal{F}$ over $\mathcal{H}$ where the stalk dimension is $d = 1$. Consequently, the feature matrices $\boldsymbol{X}_u \in \mathbb{R}^{1\times f}$ and $\boldsymbol{H}_e \in \mathbb{R}^{1\times f}$ are equivalent to the row vectors $\boldsymbol{x}_u^\top$ and $\boldsymbol{h}_e^\top$. By setting the restriction maps $\mathcal{F}_{u\trianglelefteq e} = 1 \in \mathbb{R}^{1\times 1}$ for all incidences $u \trianglelefteq e$, the OS multisets evaluate to:

$$V_e = \left\{\!\left\{1 \cdot \boldsymbol{x}_u^\top : u \in e\right\}\!\right\} = \left\{\!\left\{\boldsymbol{x}_u^\top : u \in e\right\}\!\right\},$$
$$E_u = \left\{\!\left\{(1 \cdot \boldsymbol{h}_e^\top, 1 \cdot 1) : e \ni u\right\}\!\right\} = \left\{\!\left\{(\boldsymbol{h}_e^\top, 1) : e \ni u\right\}\!\right\}.$$

Since the transport block $\mathcal{F}_{u\trianglelefteq e}^\top \mathcal{F}_{u\trianglelefteq e} = 1$ is a constant, it conveys no additional information and is trivially absorbed by the permutation-invariant function $f_{\mathcal{E}\to\mathcal{V}}$. Thus, the OS updates mathematically recover the AllSet forward pass, demonstrating that the AllSet hypothesis class is contained within OS. □

Theorem 6. *SheafHyperGNN and SheafHyperGCN (Duta et al., 2023) are special cases of* OmniSheaf.

Proof. We construct an OS layer that computes the exact operations of a SheafHyperGNN layer. Let $\boldsymbol{X} \in \mathbb{R}^{|\mathcal{V}|\times(df)}$ be the initial node features, which are reshaped into $\boldsymbol{X}_u \in \mathbb{R}^{d\times f}$. Let $\tilde{\boldsymbol{X}}_u^{(k)} = \boldsymbol{D}_u^{-1/2}\boldsymbol{X}_u^{(k)}\boldsymbol{W}_2$ denote the normalized, linearly transformed features, where $\boldsymbol{D}_u = \sum_{e\ni u} \mathcal{F}_{u\trianglelefteq e}^\top \mathcal{F}_{u\trianglelefteq e}$ is the node degree under the sheaf Laplacian, assumed non-degenerate ($\lambda_{\min}(\boldsymbol{D}_u) > 0$) or regularized via $\boldsymbol{D}_u + \epsilon \boldsymbol{I}_d$. The SheafHyperGNN update using the linear hypergraph sheaf Laplacian is given by:

$$\boldsymbol{X}_u^{(k+1)} = \sigma\left(\tilde{\boldsymbol{X}}_u^{(k)} - \boldsymbol{D}_u^{-1/2}\sum_{e\ni u}\frac{1}{\delta_e}\mathcal{F}_{u\trianglelefteq e}^\top\left(\sum_{v\in e\setminus\{u\}}\left(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}\right)\right)\right).$$

Notice that since the term for $v = u$ vanishes ($\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} = \boldsymbol{0}$), the normalized discrepancy sum over $v \in e \setminus \{u\}$ is identically equal to the sum over all nodes in e:

$$\frac{1}{\delta_e}\sum_{v\in e\setminus\{u\}}\left(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}\right) = \frac{1}{\delta_e}\sum_{v\in e}\left(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}\right)$$
$$= \mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - \frac{1}{\delta_e}\sum_{v\in e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}.$$

In Stage 1 ($V \to E$) of OS, the hyperedge multiset function computes the mean-pooled hyperedge representation over $V_e^{(k)} = \left\{\!\left\{\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)} : v \in e\right\}\!\right\}$:

$$\boldsymbol{H}_e^{(k+1)} = f_{\mathcal{V}\to\mathcal{E}}\left(V_e^{(k)}\right) = \frac{1}{\delta_e}\sum_{v\in e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)},$$

which is a standard permutation-invariant mean aggregator over $V_e^{(k)}$ (matching `aggregate = mean` in our implementation), completely independent of the receiver node u and satisfying Definition 1.

In Stage 2 ($E \to V$), node u receives the multiset $E_u^{(k+1)} = \left\{\!\left\{(\mathcal{F}_{u\trianglelefteq e}^\top \boldsymbol{H}_e^{(k+1)}, \mathcal{F}_{u\trianglelefteq e}^\top \mathcal{F}_{u\trianglelefteq e}) : e \ni u\right\}\!\right\}$ alongside its own state $\tilde{\boldsymbol{X}}_u^{(k)}$. For each incident hyperedge $e \ni u$, the local co-projected discrepancy evaluates to:

$$\mathcal{F}_{u\trianglelefteq e}^\top\left(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - \boldsymbol{H}_e^{(k+1)}\right) = (\mathcal{F}_{u\trianglelefteq e}^\top \mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{u\trianglelefteq e}^\top \boldsymbol{H}_e^{(k+1)}.$$

Because $E_u^{(k+1)}$ contains the transport blocks $\mathcal{F}_{u\trianglelefteq e}^\top \mathcal{F}_{u\trianglelefteq e}$, $f_{\mathcal{E}\to\mathcal{V}}$ natively evaluates the sheaf degree block $\boldsymbol{D}_u = \sum_{e\ni u} \mathcal{F}_{u\trianglelefteq e}^\top \mathcal{F}_{u\trianglelefteq e}$. Crucially, Stage 2 does not require access to the hyperedge cardinality δ_e, as the normalizer was already absorbed during mean pooling in Stage 1. Summing across incident hyperedges, the node aggregation function evaluates:

$$\begin{aligned}\boldsymbol{X}_u^{(k+1)} &= f_{\mathcal{E}\to\mathcal{V}}\left(E_u^{(k+1)}; \tilde{\boldsymbol{X}}_u^{(k)}\right)\\ &= \sigma\left(\tilde{\boldsymbol{X}}_u^{(k)} - \boldsymbol{D}_u^{-1/2}\sum_{e\ni u}\left[(\mathcal{F}_{u\trianglelefteq e}^\top \mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{u\trianglelefteq e}^\top \boldsymbol{H}_e^{(k+1)}\right]\right)\\ &= \sigma\left(\tilde{\boldsymbol{X}}_u^{(k)} - \boldsymbol{D}_u^{-1/2}\left(\boldsymbol{D}_u\tilde{\boldsymbol{X}}_u^{(k)} - \sum_{e\ni u}\mathcal{F}_{u\trianglelefteq e}^\top \boldsymbol{H}_e^{(k+1)}\right)\right),\end{aligned}$$

which exactly reproduces the SheafHyperGNN forward pass while performing only $|\mathcal{I}| = \sum_e |e|$ local operations.

For SheafHyperGCN, the non-linear Laplacian evaluates pairwise distances $\arg\max_{u,v\in e} \|\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u - \mathcal{F}_{v\trianglelefteq e}\boldsymbol{X}_v\|$. Since multiset V_e carries the transported features of all nodes in e, $f_{\mathcal{V}\to\mathcal{E}}$ computes pairwise distances across V_e and extracts the maximal discrepancy pair, recovering SheafHyperGCN. □

Theorem 7. *Sheaf Neural Networks on graphs (Bodnar et al., 2022; Barbero et al., 2022), including NSD and SheafAN, are special cases of* OMNISHEAF.

Proof. A standard graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}_\mathcal{G})$ is isomorphic to a 2-uniform hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ where each hyperedge $e = \{u, v\} \in \mathcal{E}$ corresponds to an edge in $\mathcal{E}_\mathcal{G}$ with uniform cardinality $|e| = 2$ and uniform degree $\delta_e = 2$. Under a cellular sheaf $\mathcal{F}$ on $\mathcal{G}$, stalks $\mathcal{F}(u), \mathcal{F}(v), \mathcal{F}(e) \cong \mathbb{R}^d$ are assigned to vertices and edges, with linear restriction maps $\mathcal{F}_{u\trianglelefteq e}, \mathcal{F}_{v\trianglelefteq e} \in \mathbb{R}^{d\times d}$. Let $\boldsymbol{X}_u \in \mathbb{R}^{d\times f}$ denote the stalk-valued node features.

1. Neural Sheaf Diffusion (NSD). In Neural Sheaf Diffusion (Bodnar et al., 2022) the normalized sheaf Laplacian is defined as $\boldsymbol{\Delta}_\mathcal{F} = \boldsymbol{D}^{-1/2}\boldsymbol{L}_\mathcal{F}\boldsymbol{D}^{-1/2}$, where $\boldsymbol{L}_\mathcal{F} = \boldsymbol{\delta}^\top\boldsymbol{\delta}$ has block diagonals $\boldsymbol{D}_u = (\boldsymbol{L}_\mathcal{F})_{uu} = \sum_{e\ni u}\mathcal{F}_{u\trianglelefteq e}^\top\mathcal{F}_{u\trianglelefteq e}$ (assumed non-degenerate) and off-diagonal blocks $(\boldsymbol{L}_\mathcal{F})_{uv} = -\mathcal{F}_{u\trianglelefteq e}^\top\mathcal{F}_{v\trianglelefteq e}$ for $e = \{u, v\}$. Let $\tilde{\boldsymbol{X}}_u^{(k)} = \boldsymbol{D}_u^{-1/2}(\boldsymbol{W}_1\boldsymbol{X}_u^{(k)}\boldsymbol{W}_2)$ represent the linearly transformed, normalized features. The discrete diffusion update at node u is:

$$\boldsymbol{X}_u^{(k+1)} = \sigma\left(\tilde{\boldsymbol{X}}_u^{(k)} - \boldsymbol{D}_u^{-1/2}\sum_{e=\{u,v\}\ni u}\mathcal{F}_{u\trianglelefteq e}^\top\left(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}\right)\right).$$

Under OS, for each 2-node hyperedge $e = \{u, v\}$, the multiset of incident transported stalks is $V_e^{(k)} = \left\{\!\left\{\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)}, \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}\right\}\!\right\}$. In Stage 1 ($V \to E$), the hyperedge function computes the mean-pooled representation:

$$\boldsymbol{H}_e^{(k+1)} = f_{\mathcal{V}\to\mathcal{E}}\left(V_e^{(k)}\right) = \frac{1}{2}\left(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} + \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}\right).$$

In Stage 2 ($E \to V$), node u receives $E_u^{(k+1)} = \left\{\!\left\{(\mathcal{F}_{u\trianglelefteq e}^\top\boldsymbol{H}_e^{(k+1)}, \mathcal{F}_{u\trianglelefteq e}^\top\mathcal{F}_{u\trianglelefteq e}) : e \ni u\right\}\!\right\}$. Evaluating the local discrepancy:

$$(\mathcal{F}_{u\trianglelefteq e}^\top\mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{u\trianglelefteq e}^\top\boldsymbol{H}_e^{(k+1)} = \frac{1}{2}\mathcal{F}_{u\trianglelefteq e}^\top\left(\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}\right).$$

Because E_u explicitly carries the transport blocks $\mathcal{F}_{u\trianglelefteq e}^\top\mathcal{F}_{u\trianglelefteq e}$, $f_{\mathcal{E}\to\mathcal{V}}$ forms the sheaf node normalizer $\boldsymbol{D}_u = \sum_{e\ni u}\mathcal{F}_{u\trianglelefteq e}^\top\mathcal{F}_{u\trianglelefteq e}$. Defining the node aggregation function as:

$$\boldsymbol{X}_u^{(k+1)} = f_{\mathcal{E}\to\mathcal{V}}\left(E_u^{(k+1)}; \tilde{\boldsymbol{X}}_u^{(k)}\right) = \sigma\left(\tilde{\boldsymbol{X}}_u^{(k)} - \boldsymbol{D}_u^{-1/2}\sum_{e\ni u} 2\left[(\mathcal{F}_{u\trianglelefteq e}^\top\mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u^{(k)} - \mathcal{F}_{u\trianglelefteq e}^\top\boldsymbol{H}_e^{(k+1)}\right]\right),$$

we exactly recover the discrete NSD and SNN diffusion equations.

2. Sheaf Attention Networks (SheafAN). A Sheaf Attention Network (Barbero et al., 2022) layer computes:

$$\boldsymbol{X}_u^{(k+1)} = \sigma\left(\sum_{e=\{u,v\}\ni u} \alpha_{uv}\mathcal{F}_{u\trianglelefteq e}^{\top}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}\right),$$

where α_{uv} denotes attention coefficients over incident edges. When attention coefficients are symmetric across each edge ($\alpha_{uv} = \alpha_{vu} = \alpha_e$, standard in undirected graphs and hypergraphs), this is directly recovered in base OS by defining $f_{\mathcal{V}\to\mathcal{E}}(V_e^{(k)}) = \alpha_e \sum_{v\in e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}$ and $f_{\mathcal{E}\to\mathcal{V}}(E_u^{(k+1)}; \tilde{\boldsymbol{X}}_u^{(k)}) = \sigma\left(\sum_{e\ni u}\mathcal{F}_{u\trianglelefteq e}^{\top}\boldsymbol{H}_e^{(k+1)}\right)$ (or subtracting $\tilde{\boldsymbol{X}}_u^{(k)}$ inside σ for Res-SheafAN). When attention coefficients are asymmetric ($\alpha_{uv} \neq \alpha_{vu}$), the interaction is inherently directed and is subsumed by OCS: the undirected edge $\{u, v\}$ is represented by the two oriented hyperedges $(T = \{u\}, H = \{v\})$ and $(T = \{v\}, H = \{u\})$, whose role-specific restriction maps $\mathcal{F}^T, \mathcal{F}^H$ and separate aggregation weights $\boldsymbol{W}_{\text{in}}, \boldsymbol{W}_{\text{out}}$ realize independent coefficients in the two directions. Note that OPS alone does not suffice here: a $\mathbb{U}(1)$ phase factor satisfies $|\alpha_{uv}| = |\alpha_{vu}|$, so it can encode a direction-dependent phase but not a direction-dependent magnitude.

3. Sheaf Message Passing Neural Networks (Sheaf-MPNN). More generally, any graph sheaf message-passing layer $\boldsymbol{X}_u^{(k+1)} = \phi_u(\boldsymbol{X}_u^{(k)}, \bigoplus_{e=\{u,v\}\ni u}\mathcal{F}_{u\trianglelefteq e}^{\top}\psi_e(\mathcal{F}_{v\trianglelefteq e}\boldsymbol{X}_v^{(k)}, \mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u^{(k)}))$ is recovered by setting $f_{\mathcal{V}\to\mathcal{E}} = \psi_e$ and $f_{\mathcal{E}\to\mathcal{V}} = \phi_u$.

Therefore, all standard graph Sheaf Neural Networks are exact special cases of OS on 2-uniform hypergraphs. Moreover, by Theorem 8, OST and vectorized ODS universally approximate these multiset transformations, confirming that our universal approximators generalize graph SNNs. □

A.4 Proofs of Universal Approximators

Theorem 8. *OmniSheafTransformer, as well as OmniDeepSheaf with vectorized stalks, are universal approximators of continuous* OmniSheaf *multiset functions.*

Proof. Following Tabaghi & Wang (2024), OS updates operate on multisets of stalk matrices in $\mathbb{R}^{d\times f}$. Since matrix vectorization $\text{Vec}\colon \mathbb{R}^{d\times f} \to \mathbb{R}^{df}$ is a linear isomorphism, any multiset of matrix-valued stalks injectively maps to a multiset of vectors in $\mathbb{R}^{df}$.

1. Universal Approximation of OmniSheafTransformer. In OST, each transported stalk representation $\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u \in \mathbb{R}^{d\times f}$ is flattened into a token in $\mathbb{R}^{df}$. By Chien et al. (2022, Proposition 4.1) and Lee et al. (2019), SetTransformer layers employing multi-head attention (PMA and MAB) are universal approximators of continuous permutation-invariant multiset functions over compact subsets of $\mathbb{R}^{df}$. Because both stage-one ($V \to E$) and stage-two ($E \to V$) updates in OS are continuous permutation-invariant multiset functions, OST unconditionally universally approximates the OS forward pass.

2. Universal Approximation of Vectorized OmniDeepSheaf. When ODS parameterizes multiset functions via unconstrained multi-layer perceptrons acting on vectorized stalks $\text{vec}(\boldsymbol{A}) \in \mathbb{R}^{df}$, the classical DeepSets universality theorem (Zaheer et al., 2017) applies directly: sum-decomposable representations $\rho(\sum\phi(\text{vec}(\boldsymbol{A})))$ can approximate any continuous permutation-invariant function on vector multisets arbitrarily well in the supremum norm. As established by Wagstaff et al. (2019), universal approximation of continuous functions on multisets of cardinality at most $N_{\max} = \max_e |e|$ requires that the latent dimension of the inner encoder ϕ satisfies $\dim(\phi) \geq N_{\max}$. Under this standard latent dimension condition, vectorized ODS is a universal approximator of OS. OST replaces sum-pooling with multi-head attention (PMA), which relaxes the sum-decomposition structure underlying the bound of Wagstaff et al. (2019); we state universality for OST under the same latent-dimension condition and leave a tighter characterization to future work.

3. Row-Wise Parameter-Efficient Instantiation. In practical implementations where stalk dimension d is large, ODS can alternatively be instantiated using row-wise MLPs operating along the feature axis f, shared across the d stalk coordinates (see Appendix B.2). This reduces the parameter footprint from $\mathcal{O}(d^2f^2)$ to $\mathcal{O}(f^2)$, while cross-stalk interactions are governed linearly by the learnable restriction maps $\mathcal{F}_{u\trianglelefteq e} \in \mathbb{R}^{d\times d}$. While row-wise MLPs without restriction maps are restricted to

row-separable functionals, coupling them with adaptive restriction maps provides a highly expressive, parameter-efficient hypothesis class that recovers classical sheaf diffusion (Theorems 6 and 7). □

Corollary 9 (Universality of Directed Variants)**.** *The directed variants OmniCopreSheaf and OmniPhaseSheaf, parameterized via OmniDeepSheaf or OmniSheafTransformer over their respective partitioned or complex stalk multisets, are universal approximators of continuous directionality-aware hypergraph multiset functions.*

Proof. We consider both directional formulations separately:

Case 1: OmniCopreSheaf. In OCS, the hyperedge multiset function takes as input a pair of multisets $(V_e^T, V_e^H) \in \mathcal{M}(\mathbb{R}^{d\times f}) \times \mathcal{M}(\mathbb{R}^{d\times f})$, where $\mathcal{M}(\mathcal{X})$ denotes the space of multisets over a compact domain $\mathcal{X} \subset \mathbb{R}^{d\times f}$. A function $F(V_e^T, V_e^H)$ is continuous and invariant under separate permutations of the elements within V_e^T and within V_e^H. By the universal approximation theorem for multiple multiset inputs (Zaheer et al., 2017; Tabaghi & Wang, 2024), any continuous invariant function on a product of compact multiset domains $\mathcal{M}(\mathcal{X}) \times \mathcal{M}(\mathcal{X})$ can be approximated arbitrarily well in the supremum norm by functions of the form:

$$F(V_e^T, V_e^H) = \rho\left(\sum_{\boldsymbol{x}\in V_e^T} \phi_T(\boldsymbol{x}) + \sum_{\boldsymbol{y}\in V_e^H} \phi_H(\boldsymbol{y})\right), \tag{20}$$

where ϕ_T, ϕ_H, and ρ are continuous mappings parameterized by multi-layer perceptrons (MLPs). Since SetTransformer layers preserve the capacity of sum-decomposable representations while parameterizing multi-head attention over both partitions (Lee et al., 2019; Chien et al., 2022), both OCS is a universal approximator on (V_e^T, V_e^H). A symmetric argument applies to the node-level update on $(E_u^{\text{in}}, E_u^{\text{out}})$.

Case 2: OmniPhaseSheaf. In OPS, cell features and restriction maps reside in the complex vector spaces $\mathbb{C}^{d\times f}$ and $\mathbb{C}^{d\times d}$, modulated by a continuous phase factor $s_{u,e} = e^{i\theta_{u,e}} \in \mathrm{U}(1)$. There exists an isometric isomorphism between $\mathbb{C}^k$ and $\mathbb{R}^{2k}$ given by the canonical decomposition $\boldsymbol{z} \mapsto (\mathrm{Re}(\boldsymbol{z}), \mathrm{Im}(\boldsymbol{z}))$. Under this homeomorphism, any compact subset $\mathcal{K}_{\mathbb{C}} \subset \mathbb{C}^{d\times f}$ is homeomorphic to a compact subset $\mathcal{K}_{\mathbb{R}} \subset \mathbb{R}^{2df}$. Because the complex phase modulation $\alpha_{u,e}\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u$ is a continuous operation over the compact group $\mathrm{U}(1) \times \mathcal{K}_{\mathbb{C}}$, the multiset of complex stalk features maps homeomorphically to a multiset of real vectors in $\mathbb{R}^{2df}$ (Zaheer et al., 2017; Keriven & Peyré, 2019; Ravanbakhsh et al., 2017; Tabaghi & Wang, 2024). At readout, complex representations are unwound into real representations $[\mathrm{Re}(\boldsymbol{X}_u) \,\|\, \mathrm{Im}(\boldsymbol{X}_u)] \in \mathbb{R}^{d\times 2f}$ followed by real decoding layers, ensuring that the network is not restricted by complex analyticity and recovers dense approximation over continuous real-valued functions. Applying Theorem 8 directly to the real-isomorphic stalk representations establishes that OPS universally approximates any continuous permutation-invariant mapping over phase-modulated stalks. □

A.5 Proofs of Generalization over Directed Models

Theorem 10. *Dir-GNN (Rossi et al., 2024) on directed graphs and DHNN (Fiorini et al., 2024) on directed hypergraphs are special cases of* OmniCopreSheaf.

Proof. Dir-GNN (Rossi et al., 2024) and DHNN (Fiorini et al., 2024) preserve directionality by explicitly segregating incoming and outgoing structural relations. Let $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ be a directed hypergraph where each hyperedge e is partitioned into a source/tail set $T_e \subset \mathcal{V}$ and a destination/head set $H_e \subset \mathcal{V}$. OmniCopreSheaf natively accommodates this by partitioning the incidence multiset into $V_e^T = \{\{\mathcal{F}_{u\trianglelefteq e}^T \boldsymbol{X}_u : u \in T_e\}\}$ and $V_e^H = \{\{\mathcal{F}_{v\trianglelefteq e}^H \boldsymbol{X}_v : v \in H_e\}\}$ with role-specific restriction maps $\mathcal{F}^T, \mathcal{F}^H$.

1. Recovery of DHNN (Fiorini et al., 2024). For general hyperedges with arbitrary $|T_e| \geq 1$ and $|H_e| \geq 1$, we set $d = 1$ (trivial stalks $\mathbb{R}$) and configure restriction maps to identity scalars $\mathcal{F}_{u\trianglelefteq e}^T = \mathcal{F}_{v\trianglelefteq e}^H = 1$. The hyperedge function aggregates the multisets into separate head and tail representations:

$$\boldsymbol{h}_e^T = f_{\mathcal{V}\to\mathcal{E}}^T(V_e^T) = \sum_{u\in T_e} \boldsymbol{x}_u, \quad \boldsymbol{h}_e^H = f_{\mathcal{V}\to\mathcal{E}}^H(V_e^H) = \sum_{v\in H_e} \boldsymbol{x}_v.$$

In the second stage, node u aggregates hyperedges where it acts as a tail versus a head using separate parametric transformations $\boldsymbol{W}_{\text{tail}}$ and $\boldsymbol{W}_{\text{head}}$:

$$\boldsymbol{x}_u^{(k+1)} = f_{\mathcal{E}\to\mathcal{V}}(E_u^{(k+1)}; \boldsymbol{x}_u^{(k)}) = \sigma\left(\boldsymbol{W}_{\text{self}}\boldsymbol{x}_u^{(k)} + \boldsymbol{W}_{\text{tail}}\sum_{e:u\in T_e}\boldsymbol{h}_e^H + \boldsymbol{W}_{\text{head}}\sum_{e:u\in H_e}\boldsymbol{h}_e^T\right),$$

which analytically recovers the dual head-tail projection update of DHNN.

2. Recovery of Dir-GNN (Rossi et al., 2024). When $\mathcal{H}$ is 2-uniform with $|T_e| = 1$ and $|H_e| = 1$ (representing a standard directed graph where each edge is an ordered pair $u \to v$), $V_e^T = \{\{\boldsymbol{x}_u\}\}$ and $V_e^H = \{\{\boldsymbol{x}_v\}\}$. The above update collapses exactly to:

$$\boldsymbol{x}_u^{(k+1)} = \sigma\left(\boldsymbol{W}_{\text{self}}\boldsymbol{x}_u^{(k)} + \boldsymbol{W}_{\text{out}}\sum_{v\in\mathcal{N}_{\text{out}}(u)}\boldsymbol{x}_v + \boldsymbol{W}_{\text{in}}\sum_{v\in\mathcal{N}_{\text{in}}(u)}\boldsymbol{x}_v\right),$$

which is identical to the defining Dir-GNN update equation (with degree normalizations $1/|\mathcal{N}(u)|$ natively absorbed into $f_{\mathcal{E}\to\mathcal{V}}$ via mean pooling). Thus, both Dir-GNN and DHNN are exact special cases of OCS. □

Theorem 11. *DSNN (Fiorini et al., 2026) on directed graphs and DSHN (Mule et al., 2026) on directed hypergraphs are special cases of* OMNIPHASESHEAF.

Proof. Both DSNN (Fiorini et al., 2026) on directed graphs and DSHN (Mule et al., 2026) on directed hypergraphs encode edge orientations via a complex phase shift $s_{u,e} = e^{i\theta_{u,e}} \in \mathbb{U}(1)$ and define diffusion using a complex-valued, Hermitian sheaf Laplacian $\boldsymbol{L}_{\mathcal{F}} = \boldsymbol{\delta}^\dagger\boldsymbol{\delta}$.

1. Recovery of DSHN on Directed Hypergraphs (Mule et al., 2026). DSHN extends the complex sheaf Laplacian to general directed hypergraphs $e = (T_e, H_e)$ by defining diffusion through pairwise differences across incident nodes weighted by $1/\delta_e$:

$$(\boldsymbol{L}_{\mathcal{F}}\tilde{\boldsymbol{X}})_u = \sum_{e\ni u}\frac{1}{\delta_e}s_{u,e}^*\mathcal{F}_{u\trianglelefteq e}^\dagger\left(\sum_{v\in e\setminus\{u\}}\left(s_{u,e}\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u - s_{v,e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v\right)\right).$$

Since the difference vanishes for $v = u$, the normalized inner discrepancy sum over $v \in e \setminus \{u\}$ equals the normalized full sum over $v \in e$:

$$\frac{1}{\delta_e}\sum_{v\in e}\left(s_{u,e}\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u - s_{v,e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v\right) = s_{u,e}\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u - \frac{1}{\delta_e}\sum_{v\in e}s_{v,e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v.$$

In Stage 1 ($V \to E$) of OPS, the hyperedge multiset function computes the mean-pooled representation over $V_e^{(k)} = \left\{\!\left\{s_{v,e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)} : v \in e\right\}\!\right\}$:

$$\boldsymbol{H}_e^{(k+1)} = f_{\mathcal{V}\to\mathcal{E}}\left(V_e^{(k)}\right) = \frac{1}{\delta_e}\sum_{v\in e}s_{v,e}\mathcal{F}_{v\trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}.$$

In Stage 2 ($E \to V$), node u receives $E_u^{(k+1)} = \left\{\!\left\{(s_{u,e}^*\mathcal{F}_{u\trianglelefteq e}^\dagger\boldsymbol{H}_e^{(k+1)}, \mathcal{F}_{u\trianglelefteq e}^\dagger\mathcal{F}_{u\trianglelefteq e}) : e \ni u\right\}\!\right\}$ and its own state $\tilde{\boldsymbol{X}}_u^{(k)}$. Crucially, because $|s_{u,e}|^2 = 1$, the conjugate transport of the local discrepancy evaluates to:

$$\begin{aligned}s_{u,e}^*\mathcal{F}_{u\trianglelefteq e}^\dagger\left(s_{u,e}\mathcal{F}_{u\trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - \boldsymbol{H}_e^{(k+1)}\right) &= |s_{u,e}|^2(\mathcal{F}_{u\trianglelefteq e}^\dagger\mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u^{(k)} - s_{u,e}^*\mathcal{F}_{u\trianglelefteq e}^\dagger\boldsymbol{H}_e^{(k+1)}\\ &= (\mathcal{F}_{u\trianglelefteq e}^\dagger\mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u^{(k)} - s_{u,e}^*\mathcal{F}_{u\trianglelefteq e}^\dagger\boldsymbol{H}_e^{(k+1)}.\end{aligned}$$

Summing across incident hyperedges without needing access to hyperedge size δ_e:

$$\sum_{e\ni u}\left[(\mathcal{F}_{u\trianglelefteq e}^\dagger\mathcal{F}_{u\trianglelefteq e})\tilde{\boldsymbol{X}}_u^{(k)} - s_{u,e}^*\mathcal{F}_{u\trianglelefteq e}^\dagger\boldsymbol{H}_e^{(k+1)}\right] = \boldsymbol{D}_u\tilde{\boldsymbol{X}}_u^{(k)} - \sum_{e\ni u}s_{u,e}^*\mathcal{F}_{u\trianglelefteq e}^\dagger\boldsymbol{H}_e^{(k+1)},$$

where $\boldsymbol{D}_u = \sum_{e \ni u} \mathcal{F}^{\dagger}_{u \trianglelefteq e} \mathcal{F}_{u \trianglelefteq e}$ is the Hermitian sheaf degree block, which is natively computed by $f_{\mathcal{E}\to\mathcal{V}}$ from the blocks $\mathcal{F}^{\dagger}_{u \trianglelefteq e} \mathcal{F}_{u \trianglelefteq e}$ present in $E_u^{(k+1)}$. Evaluating:

$$\boldsymbol{X}_u^{(k+1)} = f_{\mathcal{E}\to\mathcal{V}}\left(E_u^{(k+1)}; \tilde{\boldsymbol{X}}_u^{(k)}\right) = \sigma\left(\tilde{\boldsymbol{X}}_u^{(k)} - \boldsymbol{D}_u^{-1/2}\left(\boldsymbol{D}_u \tilde{\boldsymbol{X}}_u^{(k)} - \sum_{e \ni u} s^*_{u,e} \mathcal{F}^{\dagger}_{u \trianglelefteq e} \boldsymbol{H}_e^{(k+1)}\right)\right)$$

exactly yields the normalized DSHN diffusion update on directed hypergraphs.

2. Recovery of DSNN on Directed Graphs (Fiorini et al., 2026). When $\mathcal{H}$ is a 2-uniform directed graph where each edge $e = (u \to v)$ has $T_e = \{u\}$, $H_e = \{v\}$, and uniform degree $\delta_e = 2$, Stage 1 computes the mean $\boldsymbol{H}_e^{(k+1)} = \frac{1}{2}(s_{u,e}\mathcal{F}_{u \trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} + s_{v,e}\mathcal{F}_{v \trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)})$. The local discrepancy at node u evaluates:

$$(\mathcal{F}^{\dagger}_{u \trianglelefteq e}\mathcal{F}_{u \trianglelefteq e})\tilde{\boldsymbol{X}}_u^{(k)} - s^*_{u,e}\mathcal{F}^{\dagger}_{u \trianglelefteq e}\boldsymbol{H}_e^{(k+1)} = \frac{1}{2} s^*_{u,e}\mathcal{F}^{\dagger}_{u \trianglelefteq e}\left(s_{u,e}\mathcal{F}_{u \trianglelefteq e}\tilde{\boldsymbol{X}}_u^{(k)} - s_{v,e}\mathcal{F}_{v \trianglelefteq e}\tilde{\boldsymbol{X}}_v^{(k)}\right),$$

which identically reproduces the complex Hermitian DSNN diffusion equation on directed graphs (with the factor $1/2$ absorbed into the diffusion step size).

Therefore, both DSNN and DSHN are exact special cases of OPS. □

B Architectural and Implementation Details

In this section, we provide the full architectural equations, layer designs, and computational mechanics of OS and its universal approximators.

B.1 OmniSheaf Message Passing Pipeline

Let $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ be a hypergraph with incidence set $\mathcal{I} = \{(u, e) : u \in e\}$ of cardinality $K = |\mathcal{I}| = \sum_{e\in\mathcal{E}} |e|$. Each node $u \in \mathcal{V}$ is associated with a stalk-valued feature matrix $\boldsymbol{X}_u \in \mathbb{R}^{d\times f}$, and each hyperedge $e \in \mathcal{E}$ holds an aggregated stalk matrix $\boldsymbol{H}_e \in \mathbb{R}^{d\times f}$, where d denotes the stalk dimension and f represents the hidden channel capacity.

A single convolutional layer of OS updates cell features through a two-stage multiset message-passing scheme:

$$\textbf{Stage 1 } (V \to E)\textbf{:} \quad \boldsymbol{H}_e^{(k+1)} = f_{\mathcal{V}\to\mathcal{E}}\left(\left\{\!\!\left\{\mathcal{F}^{(k)}_{u \trianglelefteq e}\boldsymbol{X}_u^{(k)} : u \in e\right\}\!\!\right\}; \boldsymbol{H}_e^{(k)}\right), \tag{21}$$

$$\textbf{Stage 2 } (E \to V)\textbf{:} \quad \boldsymbol{X}_u^{(k+1)} = f_{\mathcal{E}\to\mathcal{V}}\left(\left\{\!\!\left\{\left((\mathcal{F}^{(k)}_{u \trianglelefteq e})^{\top}\boldsymbol{H}_e^{(k+1)}, (\mathcal{F}^{(k)}_{u \trianglelefteq e})^{\top}\mathcal{F}^{(k)}_{u \trianglelefteq e}\right) : e \ni u\right\}\!\!\right\}; \boldsymbol{X}_u^{(k)}\right). \tag{22}$$

Row-Wise Channel Transforms. To prevent dimensional explosion and isolate geometric alignment from feature extraction, linear transformations and activation functions in the encoder, decoder, and update blocks operate *row-wise* along the feature axis f, shared across the d stalk coordinates. Formally, for an MLP $\phi\colon \mathbb{R}^{f_{\text{in}}} \to \mathbb{R}^{f_{\text{out}}}$ and stalk matrix $\boldsymbol{A} \in \mathbb{R}^{d\times f_{\text{in}}}$, the row-wise mapping $\phi[\boldsymbol{A}] \in \mathbb{R}^{d\times f_{\text{out}}}$ satisfies:

$$(\phi[\boldsymbol{A}])_{i,:} = \phi(\boldsymbol{A}_{i,:}), \quad \forall i \in \{1, \dots, d\}. \tag{23}$$

Spatial coordinate mixing across stalk dimensions is governed exclusively by the restriction matrices $\mathcal{F}_{u \trianglelefteq e} \in \mathbb{R}^{d\times d}$. This parameterization ensures:

1. **Parameter Economy:** The parameter footprint of internal MLPs scales as $\mathcal{O}(f^2)$ rather than $\mathcal{O}(d^2 f^2)$, saving orders of magnitude in trainable weights at higher stalk dimensions ($d \in \{4, 5, 6\}$).
2. **Coordinate Gauge Preservation:** Permutations of basis axes in stalk coordinates $\mathcal{P}_d \leq \mathrm{O}(d)$ commute directly through the row-wise encoders and decoders (extending to signed permutations B_d for odd activations), preserving coordinate gauge equivariance (Proposition 3), while linear sheaf message passing preserves the full $\mathrm{O}(d)$ / $\mathrm{U}(d)$ gauge group.

B.2 Universal Approximator Instantiations

We realize OS through foundational parameterizations of multiset functions:

1. OmniDeepSheaf (ODS): Depending on computational constraints and expressivity requirements, ODS supports two distinct instantiations:

- **Vectorized OmniDeepSheaf (Full Capacity):** Both $f_{\mathcal{V}\to\mathcal{E}}$ and $f_{\mathcal{E}\to\mathcal{V}}$ are parameterized via DeepSets (Zaheer et al., 2017) operating directly over vectorized stalk representations $\mathrm{vec}(\boldsymbol{A}) \in \mathbb{R}^{df}$:

$$\boldsymbol{H}_e^{(k+1)} = \mathrm{mat}_{d,f}\left(\psi_{\mathcal{E}}\left(\sum_{u\in e}\phi_{\mathcal{V}}\left(\mathrm{vec}(\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u^{(k)})\right)\right)\right), \tag{24}$$

$$\boldsymbol{X}_u^{(k+1)} = \mathrm{mat}_{d,f}\left(\psi_{\mathcal{V}}\left(\sum_{e\ni u}\phi_{\mathcal{E}}\left(\mathrm{vec}(\mathcal{F}_{u\trianglelefteq e}^{\top}\boldsymbol{H}_e^{(k+1)})\right) + \boldsymbol{W}_{\mathrm{self}}\mathrm{vec}(\boldsymbol{X}_u^{(k)})\right)\right), \tag{25}$$

  where $\phi_{\mathcal{V}}, \phi_{\mathcal{E}}, \psi_{\mathcal{V}}, \psi_{\mathcal{E}}$ are unconstrained MLPs on $\mathbb{R}^{df}$, and $\mathrm{mat}_{d,f}$ reshapes vectors back into $\mathbb{R}^{d\times f}$. By Zaheer et al. (2017), this variant is a provable universal approximator of continuous OS multiset functions (Theorem 8).
- **Row-Wise OmniDeepSheaf (Parameter Efficient):** To avoid $\mathcal{O}(d^2f^2)$ parameter scaling at higher stalk dimensions, MLPs are applied row-wise along the feature channel axis f and shared across the d stalk coordinates:

$$\boldsymbol{H}_e^{(k+1)} = \psi_{\mathcal{E}}\left[\sum_{u\in e}\phi_{\mathcal{V}}\left[\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u^{(k)}\right]\right], \tag{26}$$

$$\boldsymbol{X}_u^{(k+1)} = \psi_{\mathcal{V}}\left[\sum_{e\ni u}\phi_{\mathcal{E}}\left[\mathcal{F}_{u\trianglelefteq e}^{\top}\boldsymbol{H}_e^{(k+1)}\right] + \boldsymbol{W}_{\mathrm{self}}\boldsymbol{X}_u^{(k)}\right], \tag{27}$$

  where $\phi_{\mathcal{V}}, \phi_{\mathcal{E}}, \psi_{\mathcal{V}}, \psi_{\mathcal{E}}$ require only $\mathcal{O}(f^2)$ parameters, while coordinate mixing across stalk dimensions is mediated by the learnable restriction maps $\mathcal{F}_{u\trianglelefteq e} \in \mathbb{R}^{d\times d}$.

2. OmniSheafTransformer (OST): To capture higher-order contextual correlations among elements within hyperedges and incident star-neighborhoods, we replace sum-pooling with the Pooling by Multi-head Attention (PMA) mechanism of SetTransformer (Lee et al., 2019; Chien et al., 2022). Let $\mathrm{MAB}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V})$ denote a Multi-head Attention Block. For n_h attention heads, the hyperedge representation is computed via learned seed vectors $\boldsymbol{S} \in \mathbb{R}^{k\times(df)}$:

$$\mathrm{PMA}_k(\boldsymbol{Z}) = \mathrm{MAB}(\boldsymbol{S}, \mathrm{rFF}(\boldsymbol{Z}), \mathrm{rFF}(\boldsymbol{Z})), \tag{28}$$

where $\boldsymbol{Z} \in \mathbb{R}^{|e|\times(df)}$ is formed by stacking the flattened stalk representations $(\mathrm{vec}(\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u))_{u\in e}$, and rFF denotes a feedforward network. Because SetTransformer operates directly on the full df-dimensional tokens, OST unconditionally guarantees universal approximation of OS (Theorem 8).

B.3 Restriction Map Builders and Parameterization Families

The restriction map $\mathcal{F}_{u\trianglelefteq e}$ models the alignment between the private coordinate space of node u and the common semantic coordinate space of hyperedge e. Following Bodnar et al. (2022) and Duta et al. (2023), maps are predicted dynamically per layer from the cell features:

$$\mathcal{F}_{u\trianglelefteq e} = \Psi\left(\sigma_{\mathcal{F}}\left(\Phi\left(\bar{\boldsymbol{x}}_u \,\middle\|\, \bar{\boldsymbol{h}}_e\right)\right)\right), \tag{29}$$

where $\bar{\boldsymbol{x}}_u = \frac{1}{d}\sum_{i=1}^{d}(\boldsymbol{X}_u)_{i,:} \in \mathbb{R}^f$ is the stalk-averaged node embedding, and $\bar{\boldsymbol{h}}_e = \sum_{v\in e}\phi_{\mathrm{edge}}(\bar{\boldsymbol{x}}_v) \in \mathbb{R}^f$ is a pooled hyperedge context embedding. $\Phi\colon \mathbb{R}^{2f} \to \mathbb{R}^p$ is a linear projection layer, $\sigma_{\mathcal{F}} \in \{\mathrm{id}, \tanh, \mathrm{sigmoid}\}$ is a non-linear activation, and Ψ reshapes and constrains the output into a specific matrix manifold:

- **Diagonal Sheaves ($\mathcal{D}(d)$):** Ψ maps $p = d$ predicted values directly to a diagonal matrix:

$$\mathcal{F}_{u\trianglelefteq e} = \mathrm{diag}(v_1, \ldots, v_d) \in \mathbb{R}^{d\times d}. \tag{30}$$

  This requires only $\mathcal{O}(d)$ parameters and permits exact degree inversions $\boldsymbol{D}_u^{-1}$ via elementwise operations.

- **General Linear Sheaves ($\mathrm{GL}(d)$):** Ψ reshapes $p = d^2$ values into an unconstrained matrix in $\mathbb{R}^{d\times d}$.
- **Orthogonal Sheaves ($\mathrm{O}(d)$):** To preserve feature norms and prevent exponential explosion or collapse over deep layers, Ψ parameterizes orthogonal matrices via a product of d Householder reflections (Mathiasen et al., 2020; Bodnar et al., 2022). Let $\boldsymbol{v}_k \in \mathbb{R}^d$ be predicted vectors ($k \in \{1, \dots, d\}$). The restriction map is formed as:

$$\mathcal{F}_{u\trianglelefteq e} = \prod_{k=1}^{d} \left(\boldsymbol{I}_d - 2\frac{\boldsymbol{v}_k \boldsymbol{v}_k^\top}{\|\boldsymbol{v}_k\|_2^2} \right) \in \mathrm{O}(d). \tag{31}$$

  In our implementation, this is evaluated via a custom C++/CUDA just-in-time (JIT) extension (`torch-householder`), ensuring high GPU throughput during backpropagation.

B.4 Directional OmniSheaf Variants

In a directed hypergraph, each hyperedge e is an ordered pair of disjoint subsets $e = (T_e, H_e)$, where $T_e \subset \mathcal{V}$ represents the tail (source nodes) and $H_e \subset \mathcal{V}$ represents the head (destination nodes).

1. OmniPhaseSheaf (OPS): Following Fiorini et al. (2026) and Mule et al. (2026), we map directional orientation into a continuous $\mathbb{U}(1)$ magnetic flux. Node and hyperedge features are lifted to the complex field $\mathbb{C}^{d\times f}$. For a magnetic charge parameter $q \in [0, 0.25]$, we define the directional phase shift $s_{u,e} \in \mathbb{C}$ as:

$$s_{u,e} = \begin{cases} 1 + 0i = e^{i\cdot 0}, & \text{if } u \in H_e \text{ (head)}, \\ e^{i2\pi q} = \cos(2\pi q) + i\sin(2\pi q), & \text{if } u \in T_e \text{ (tail)}. \end{cases} \tag{32}$$

The directed restriction map becomes $\mathcal{F}^{\mathrm{dir}}_{u\trianglelefteq e} = s_{u,e}\mathcal{F}_{u\trianglelefteq e}$. Because $|s_{u,e}| = 1$, the inner transport block satisfies:

$$(\mathcal{F}^{\mathrm{dir}}_{u\trianglelefteq e})^\dagger \mathcal{F}^{\mathrm{dir}}_{u\trianglelefteq e} = (s^*_{u,e}\mathcal{F}^\top_{u\trianglelefteq e})(s_{u,e}\mathcal{F}_{u\trianglelefteq e}) = |s_{u,e}|^2 \mathcal{F}^\top_{u\trianglelefteq e}\mathcal{F}_{u\trianglelefteq e} = \mathcal{F}^\top_{u\trianglelefteq e}\mathcal{F}_{u\trianglelefteq e}, \tag{33}$$

where $\dagger$ denotes the conjugate transpose. For OPS-T, the attention coefficients in `ComplexSheafPMA` are computed via the real-part projection of the phase-rotated query-key dot product:

$$\alpha_{u,e} = \frac{\exp\left(\frac{1}{\sqrt{d}}\mathrm{Re}\left(\langle \mathbf{q}_e, s_{u,e}\mathcal{F}_{u\trianglelefteq e}\boldsymbol{X}_u\rangle\right)\right)}{\sum_{v\in e}\exp\left(\frac{1}{\sqrt{d}}\mathrm{Re}\left(\langle \mathbf{q}_e, s_{v,e}\mathcal{F}_{v\trianglelefteq e}\boldsymbol{X}_v\rangle\right)\right)} \cdot s_{u,e}. \tag{34}$$

Before classification, complex feature matrices $\boldsymbol{X}_u \in \mathbb{C}^{d\times f}$ are unwound into real representations $[\mathrm{Re}(\boldsymbol{X}_u) \,\|\, \mathrm{Im}(\boldsymbol{X}_u)] \in \mathbb{R}^{d\times 2f}$.

2. OmniCopreSheaf (OCS): Generalizing Dir-GNN (Rossi et al., 2024) and DHNN (Fiorini et al., 2024), OCS topologically bifurcates message passing into dual copresheaf pathways with independent parameters:

$$V_e^T = \left\{\left\{\mathcal{F}^T_{u\trianglelefteq e}\boldsymbol{X}_u : u \in T_e\right\}\right\}, \quad V_e^H = \left\{\left\{\mathcal{F}^H_{v\trianglelefteq e}\boldsymbol{X}_v : v \in H_e\right\}\right\}, \tag{35}$$

where $\mathcal{F}^T$ and $\mathcal{F}^H$ are parameterized by distinct MLPs. The hyperedge aggregation function combines both multisets via:

$$\boldsymbol{H}_e^{(k+1)} = \psi_{\mathcal{E}}\left[\phi^T\left[\sum_{u\in T_e}\mathcal{F}^T_{u\trianglelefteq e}\boldsymbol{X}_u\right] \Big\| \phi^H\left[\sum_{v\in H_e}\mathcal{F}^H_{v\trianglelefteq e}\boldsymbol{X}_v\right]\right]. \tag{36}$$

In stage two, incident hyperedges are partitioned into incoming hyperedges $\mathcal{E}_{\mathrm{in}}(u) = \{e \in \mathcal{E} : u \in H_e\}$ and outgoing hyperedges $\mathcal{E}_{\mathrm{out}}(u) = \{e \in \mathcal{E} : u \in T_e\}$, updating node features via:

$$\boldsymbol{X}_u^{(k+1)} = \psi_{\mathcal{V}}\left[\boldsymbol{W}_{\mathrm{self}}\boldsymbol{X}_u^{(k)} + \boldsymbol{W}_{\mathrm{in}}\sum_{e\in\mathcal{E}_{\mathrm{in}}(u)}(\mathcal{F}^H_{u\trianglelefteq e})^\top \boldsymbol{H}_e^{(k+1)} + \boldsymbol{W}_{\mathrm{out}}\sum_{e\in\mathcal{E}_{\mathrm{out}}(u)}(\mathcal{F}^T_{u\trianglelefteq e})^\top \boldsymbol{H}_e^{(k+1)}\right]. \tag{37}$$

B.5 Numerical Reduction Validation

To verify our theoretical subsumption theorems (Theorems 5 to 7, 10 and 11) empirically, the learned non-linear multiset functions are bypassed and replaced with exact degree-normalized linear operators:

- `DSHN`: Stage 1 scales by the inverse hyperedge degree $\delta_e^{-1} = |e|^{-1}$, and Stage 2 multiplies by the block degree inverse $\boldsymbol{D}_u^{-1} = (\sum_{e \ni u} \mathcal{F}_{u \trianglelefteq e}^{\dagger} \mathcal{F}_{u \trianglelefteq e})^{-1}$. For Diag and Ortho families, this block inverse is diagonal and evaluated elementwise, reproducing DSHN Eq. (2) exactly.
- `Dir-GNN`: Linear pooling over tail and head partitions reproduces Dir-GNN.

These modes are verified via unit tests against dense spectral reference implementations to machine precision ($< 10^{-6}$ Frobenius error).

B.6 Hyperedge Classification Readout

For hyperedge classification on chemical reaction networks (Mule et al., 2026), hyperedge representations $\boldsymbol{X}_{\text{edge}} \in \mathbb{R}^{|\mathcal{E}| \times f}$ are formed by sum-aggregating the final layer's node stalk embeddings over the binary incidence matrix $\boldsymbol{B} \in \{0,1\}^{|\mathcal{V}| \times |\mathcal{E}|}$:

$$\boldsymbol{X}_{\text{edge}} = \boldsymbol{B}^{\top} \left(\frac{1}{d} \sum_{i=1}^{d} (\boldsymbol{X}^{(L)})_{i,:} \right) \in \mathbb{R}^{|\mathcal{E}| \times f}. \tag{38}$$

The matrix $\boldsymbol{X}_{\text{edge}}$ is then passed through a 2-layer classification MLP to output logits over reaction classes.

C Experimental Setup and Hardware Infrastructure

C.1 Hardware Infrastructure and Compute Environment

All empirical benchmarks, sweeps, and cost profilings were conducted on NVIDIA L4 Cloud Instances (24 GB GDDR6) on AWS EC2 `g6.xlarge` instance.

Software Stack: Ubuntu 22.04 LTS, Python 3.13, PyTorch 2.8 with CUDA 12.8 runtime, PyTorch Geometric (PyG) 2.5, `torch-scatter` 2.1, and `torch-householder` (compiled via MSVC/GCC). All sweep pipelines and trial tracking were orchestrated using Weights & Biases (`WandB`).

C.2 Dataset Descriptions and Preprocessing Protocols

We evaluate on 25 benchmark datasets spanning directed node classification, synthetic benchmarks, molecular hyperedge classification, and undirected citation/committee networks. Comprehensive statistics are summarized in Table 5.

1. Directed Real-World Benchmarks. Constructed following Mule et al. (2026) and Fiorini et al. (2024) by transforming directed graph benchmarks into forward directed hypergraphs. For each node v, a directed hyperedge e_v is created with a singleton tail $T(e_v) = \{v\}$ and a head set $H(e_v) = \mathcal{N}_{\text{out}}(v)$ comprising all nodes in its out-neighborhood. Where node attributes are missing (EMAIL-EU, EMAIL-ENRON), we impute local structural degree features. Data splits follows Mule et al. (2026).

2. Synthetic Directional Benchmarks. Synthesized according to Fiorini et al. (2026) to evaluate directional sensitivity. Each graph consists of $|\mathcal{V}| = 500$ vertices partitioned into $c = 5$ classes. Each class contains $I_i = 30$ undirected intra-class hyperedges. For each ordered pair of distinct classes (C_i, C_j) with $i < j$, we introduce $I_o \in \{10, 30, 50\}$ directed inter-class hyperedges with tail in C_i and head in C_j.

3. Molecular Reaction Networks. Derived from USPTO chemical reaction datasets (Mule et al., 2026), where reactants and reagents form the tail set T_e, products form the head set H_e, and the task is to predict the chemical reaction type (binary F1 score).

Table 5: Dataset statistics across all 25 evaluated benchmarks. $|\mathcal{V}|$: number of vertices; $|\mathcal{E}|$: number of hyperedges; $|\mathcal{I}|$: number of incidences ($\sum_e |e|$); f: input feature dimension; C: number of target classes; $\bar{\delta}_e$: average hyperedge cardinality; $\bar{d}_v$: average vertex degree; CE Hom.: clique-expansion homophily ratio (Wang et al., 2023a).

Dataset	**Type**	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{E}\rvert$	$\lvert\mathcal{I}\rvert$	f	C	$\bar{\delta}_e$	CE Hom.
Real-World Directed Hypergraphs (Mule et al., 2026)								
Roman-empire	Directed Node	22,662	22,662	61,867	300	18	2.73	0.2363
Squirrel	Directed Node	2,223	2,060	49,048	2,089	5	23.81	0.2448
email-EU	Directed Node	986	787	34,124	1*	10	43.36	0.2608
Telegram	Directed Node	245	183	9,095	1	4	49.70	0.2854
Chameleon	Directed Node	890	797	9,651	2,325	5	12.11	0.3221
email-Enron	Directed Node	143	139	2,721	1*	7	19.58	0.3251
Cornell	Directed Node	183	96	391	1,703	5	4.07	0.4200
Wisconsin	Directed Node	251	170	670	1,703	5	3.94	0.4398
Amazon-ratings	Directed Node	24,492	24,456	137,687	300	5	5.63	0.4460
Texas	Directed Node	183	110	419	1,703	5	3.81	0.5049
Citeseer (dir)	Directed Node	3,312	1,951	6,535	3,703	6	3.35	0.7947
Cora (dir)	Directed Node	2,708	1,565	6,995	1,433	7	4.47	0.8035
Synthetic Directional Hypergraphs (Fiorini et al., 2026)								
Synth ($I_o = 10$)	Directed Node	500	250	2,262	1*	5	9.05	0.6233
Synth ($I_o = 30$)	Directed Node	500	450	4,855	1*	5	10.79	0.5020
Synth ($I_o = 50$)	Directed Node	500	650	7,559	1*	5	11.63	0.4528
Molecular Reaction Networks (Mule et al., 2026)								
Molecular-1	Hyperedge Class	4,125	1,375	11,250	1,024	2	8.18	–
Molecular-2	Hyperedge Class	4,125	1,375	11,250	1,024	2	8.18	–
Undirected Hypergraphs (Duta et al., 2023)								
Cora (und)	Undirected Node	2,708	1,579	4,287	1,433	7	2.71	0.8120
Citeseer (und)	Undirected Node	3,327	1,079	3,142	3,703	6	2.91	0.7480
Pubmed (und)	Undirected Node	19,717	7,963	29,865	500	3	3.75	0.8250
Cora-CA	Undirected Node	2,708	1,072	4,287	1,433	7	4.00	0.8310
DBLP-CA	Undirected Node	41,302	22,363	140,863	1,425	6	6.30	0.8870
Senate	Undirected Node	282	315	1,524	2	2	4.84	0.5210
House	Undirected Node	1,290	340	12,854	2	2	37.81	0.5430
Congress	Undirected Node	1,718	83,105	734,018	2	2	8.83	0.8720

*Indicates structural degree features used in the absence of raw node attributes.

4. Undirected Benchmarks (Duta et al. Construction). Evaluated on the exact benchmark suite of Duta et al. (2023), including co-citation networks (CORA, CITESEER, PUBMED), co-authorship networks (CORA-CA, DBLP-CA), and political committee bipartite hypergraphs (SENATE, HOUSE, CONGRESS). In strict accordance with the baseline protocol of Duta et al. (2023) and Chien et al. (2022), *self-loop hyperedges* are prepended to every node, ensuring exact topological parity with published baselines.

D HYPERPARAMETER SEARCH AND TUNING DETAILS

D.1 HYPERPARAMETER SEARCH SPACE

To ensure fair comparison, our search spaces match the freedom granted to baseline models in Duta et al. (2023) and Mule et al. (2026). We optimize parameters using random search via `WandB` sweeps with Hyperband pruning ($\text{min_iter} = 1, \eta = 3$). The exact search grids are provided in Table 6.

Table 6: Hyperparameter search grids across all OS variants.

Hyperparameter	**Search Grid / Values**
Stalk dimension (d)	$\{1, 2, 3, 4, 5, 6\}$ (for Ortho: $\{2, 3, 4, 5, 6\}$)
Hidden feature width (w)	$\{32, 64, 128, 256, 512\}$
Convolutional layers (L_{conv})	$\{1, 2, 3, 4, 5\}$
Internal MLP layers (L_{MLP})	$\{1, 2\}$
Attention heads (n_h)	$\{1, 4, 8\}$ (Transformer variants only)
Restriction map family ($\mathcal{F}$)	$\{\text{Diag}, \text{Ortho}, \text{General}\}$
Restriction map activation ($\sigma_{\mathcal{F}}$)	$\{\text{none}, \tanh, \text{sigmoid}\}$
Restriction map dropout	$\{\text{true}, \text{false}\}$
Magnetic charge (q)	$\{0.0, 0.05, 0.10, 0.15, 0.20, 0.25\}$ (OPS only)
Learning rate (lr)	$\{0.02, 0.01, 0.005, 0.001\}$
Weight decay (wd)	$\{0.0, 5 \times 10^{-5}, 5 \times 10^{-4}\}$
Feature dropout (p)	$\{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$
Residual skip connections	$\{\text{true}, \text{false}\}$
Input normalization	$\{\text{true}, \text{false}\}$
Fixed Hyperparameters (All Runs)	
Optimizer	Adam (Kingma & Ba, 2015)
Max Epochs / Early Stopping Patience	500 epochs / 200 epochs
Classifier Depth / Aggregation	2 layers / `mean`
Layer Normalization	LayerNorm (`ln`)

D.2 BUDGET ALLOCATION AND MEMORY CAPPING RULES

We run 150 randomly chosen trials via `WandB` for each dataset.

Capacity Constraints on Memory Outliers. For ROMAN-EMPIRE, AMAZON-RATINGS, and DBLP-CA, the search space is capped at $d \leq 4$, $w \leq 128$, and $L \leq 3$ to prevent GPU memory thrashing during random sampling.

CONGRESS represents the extreme memory outlier of the entire project: containing 83,105 hyperedges over just 1,718 nodes, adding self-loops yields 734,018 incidences (over $5\times$ DBLP-CA). Peak training memory scales at ~ 0.034 GB per unit of $(d \times w \times L)$, meaning an unconstrained trial at $(6 \times 256 \times 3)$ would demand ~ 48 GB. Because node features are a 2-dimensional one-hot label, the graph is not capacity-limited; we thus cap $w \in \{32, 64\}$ and $L \in \{1, 2\}$, while preserving the full stalk dimension range $d \in \{1, \ldots, 6\}$.

D.3 TWO-STAGE SELECTION AND 10-SPLIT CONFIRMATION PROTOCOL

To eliminate selection bias, we enforce the same protocol of Mule et al. (2026).

D.4 Statistical Significance Testing

To verify whether performance gains are statistically significant, we execute a two-sided Welch's t-test comparing our 10-run test accuracy vectors against the best-performing baseline per dataset. We apply the Holm-Bonferroni correction to control the family-wise error rate (FWER) at $\alpha = 0.05$ across the family of hypotheses. Significant improvements are designated by † in all tables.

E Extended Experimental Results and Cost Profiling

E.1 Hyperparameter Sensitivity and Qualitative Findings

Inspecting the optimal configurations in Table 12 reveals several key insights:

1. **Magnetic Charge and Homophily:** On Cora, the most homophilic citation network in the suite, the optimal magnetic charge collapses to $q = 0.0$ for both phase variants. In contrast, on directed heterophilic networks (Roman-empire, Telegram, Squirrel) and synthetic directional graphs, optimal performance requires non-zero magnetic flux ($q \in [0.05, 0.25]$), indicating that continuous complex phase modulation captures asymmetric data flow.
2. **Stalk Dimension Expressivity:** While DSHN and SheafHyperGCN are capped at $d \leq 2$ on the largest hypergraph in our study, OS usually prefers higher dimensions: $d \in \{3, 4, 5, 6\}$ was selected in 79.7% of the winning configurations.
3. **Orthogonal Manifolds:** The orthogonal restriction map family ($\mathrm{O}(d)$) was selected in 53.4% of best configurations, indicating that norm-preserving linear transport prevents vanishing and exploding gradients across bipartite incidences.

E.2 Computational Efficiency, Peak Memory, and Scaling

Tables 19, 28, 37 and 46 provide a rigorous benchmark of peak memory and latency measured on an NVIDIA L4 GPU.

- **Elimination of OOM Failures:** On Roman-empire the clique-based baselines exhaust a 24 GB card at $d = 3$ (SheafHyperGCN, DSHN) and at $d = 4$ (SheafHyperGNN). ODS and OST run to $d = 6$ in 13.76 GB and 14.45 GB. DSHNLight, which replaces the dense clique with a sparser approximation, also reaches $d = 6$ and does so in 7.95 GB: it is more memory-frugal than OS, and we note this rather than restricting the comparison to the methods we beat.
- **Training Speedups:** At $d = 2$ on Roman-empire, ODS takes 256.7 ms per step against 1816.5 ms for DSHN ($7.1\times$) and 1004.8 ms for SheafHyperGCN ($3.9\times$). The largest gap in the grid is at $d = 1$ with General restriction maps, where DSHN needs 3554.1 ms against 123.6 ms for ODS ($28.7\times$).
- **Where the Advantage Disappears:** On Telegram (245 nodes) most clique baseline are cheaper than OS at every d on time, memory and FLOPs. The quadratic term in $\mathcal{O}(\sum_e |e|^2)$ is negligible at that size, so the per-incidence MLPs dominate the budget.

E.3 GFLOPs and Arithmetic Complexity Analysis

While wall-clock latency (ms/step) and peak allocated VRAM (GB) provide practical measures of execution cost on a specific accelerator, runtime timings inherently depend on GPU architecture, memory bandwidth, caching hierarchies, and driver versions. To provide a rigorous, *machine-independent* quantification of computational cost, we benchmark the exact floating-point operations (FLOPs) required by OS and baseline models.

E.3.1 FLOP Profiling Methodology

FLOP profiling is executed utilizing PyTorch's native operator profiling mode, `torch.utils.flop_counter.FlopCounterMode`. Because floating-point operations are deterministic for static computational graphs, a single forward and backward evaluation per configuration provides exact arithmetic measurements.

We measure and report two distinct metrics:

1. **Forward GFLOPs (GFLOP$_{\text{fwd}}$):** Profiled in evaluation mode (`model.eval()`) within a `torch.no_grad()` context, capturing the arithmetic operations required for a single inference pass.
2. **Training GFLOPs (GFLOP$_{\text{train}}$):** Profiled in training mode (`model.train()`), capturing the composite forward pass, cross-entropy/negative log-likelihood loss evaluation, and exact autograd backward pass ($\nabla_\theta \mathcal{L}$) via `.backward()`. This measures the true floating-point workload of an optimization step.

Scope and Semantics of the Profiler. `FlopCounterMode` tracks tensor-level floating-point matrix multiplications (`matmul`, `addmm`, `bmm`), general linear projections, attention dot-products, and Householder reflector products ($\mathrm{O}(d)$). In accordance with standard arithmetic complexity theory, memory-bound indexing operations (e.g., `scatter_add`, `gather`, and index selection over incidences) do not perform floating-point multiplications and are thus assigned zero arithmetic FLOPs. Consequently, FLOP counts reflect the pure *arithmetic intensity* of the neural operators, isolating mathematical complexity from memory-bandwidth bottlenecks (such as cuSPARSE cache thrashing).

E.3.2 Theoretical vs. Empirical Scaling: Clique Explosion vs. OmniSheaf

As established in Theorem 4, clique-based hypergraph sheaf architectures (Duta et al., 2023; Mule et al., 2026) expand every hyperedge e into $S_2 = \sum_{e\in\mathcal{E}} |e|^2$ pairwise block interactions. In terms of arithmetic operations, assembling, normalizing, and diffusing over this quadratic graph clique requires $\mathcal{O}(m\bar{e}^2 d + nc^2)$ FLOPs for diagonal restriction maps and $\mathcal{O}(m\bar{e}^2 d^3 + nc^2)$ FLOPs for general linear maps (Mule et al., 2026), where $m = |\mathcal{E}|$, $\bar{e}$ is the average hyperedge cardinality, and $c = df$.

In contrast, OS operates exclusively over the bipartite incidence graph $\mathcal{I} = \{(u, e) : u \in e\}$, restricting message passing to $K = |\mathcal{I}| = \sum_{e\in\mathcal{E}} |e|$ linear transformations. In ODS, the arithmetic cost of transporting stalks via $\mathcal{F}_{u\trianglelefteq e}$ and row-wise MLPs scales as:

$$\text{FLOPs(ODS)} = \mathcal{O}\left(|\mathcal{I}| \cdot (d^2 + df) + |\mathcal{V}| \cdot f^2\right), \tag{39}$$

which is *linear* in the number of incidences $|\mathcal{I}|$ and linear in the channel dimension f.

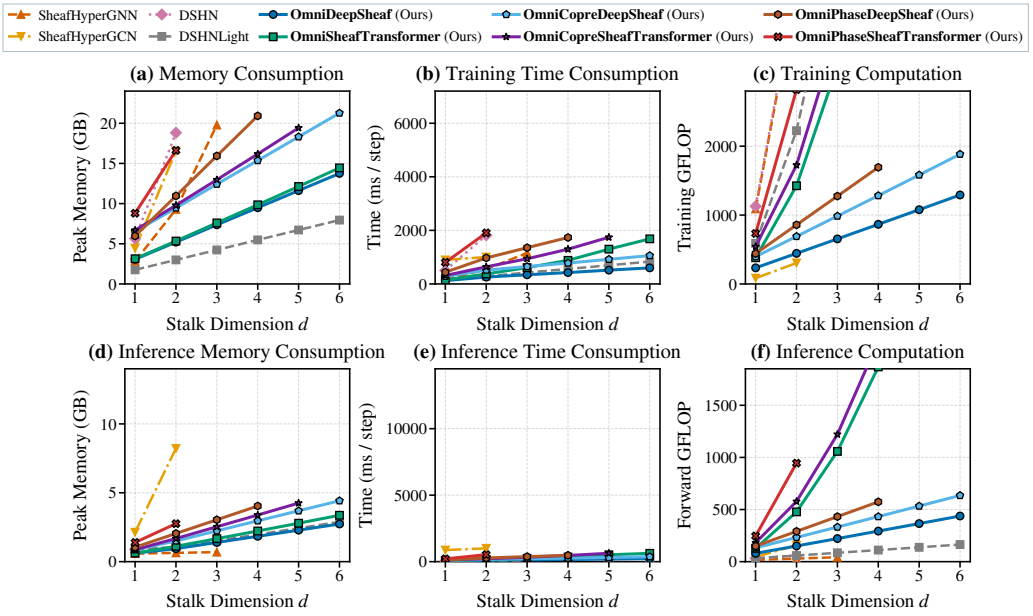


Figure 4: *Scaling across stalk dimensions* $d \in \{1, \ldots, 6\}$ *on Roman-empire.* Top row training, bottom row inference; left to right: peak memory (GB), step time (ms), and GFLOPs. OS variants scale linearly in d; the clique-based baselines grow quadratically and terminate where they exhaust the 24 GB card.

E.3.3 Empirical Findings across Benchmarks

The complete empirical GFLOP counts across all three restriction-map families (Diag, General, and Ortho) are compiled in Tables 49, 52, 55, 58, 61 and 64 (Diag), Tables 50, 53, 56, 59, 62 and 65 (General), and Tables 51, 54, 57, 60, 63 and 66 (Ortho). Analysis of these results confirms three primary properties:

1. **Strict Linearity in Stalk Dimension d:** On Roman-empire (width 256, 3 layers, Diag maps, Table 64), the training GFLOPs of ODS progress as:

$$234 \to 444 \to 654 \to 866 \to 1078 \to 1292 \text{ GFLOPs},$$

   exhibiting a constant step of $\Delta \approx 210$ GFLOPs per stalk dimension. In contrast, clique-based SheafHyperGNN exhibits steep superlinear growth ($1090 \to 4274 \to 9562$ GFLOPs) before exhausting GPU memory at $d = 4$.

2. **Massive FLOP Reductions at High Stalk Dimensions:** At $d = 6$ on Roman-empire, the lightest competitive sheaf baseline (DSHNLight) requires $19,286$ training GFLOPs. ODS executes the identical optimization step in only $1,292$ GFLOPs, representing an immediate $14.9\times$ **reduction in arithmetic operations**. Against full SheafHyperGNN at $d = 2$, ODS reduces training operations from $4,274$ GFLOPs to 444 GFLOPs ($9.6\times$ **reduction**).

3. **Dense Polyadic Efficiency on High-Degree Hyperedges:** On Telegram ($\bar{\delta}_e = 49.70$, Table 52), hyperedges contain up to dozens of nodes. While pairwise clique expansion squares this cardinality ($|e|^2 \approx 2500$ pairwise edges per hyperedge), OS routes messages directly through the hyperedge stalk, keeping the forward pass under 45.2 GFLOPs at $d = 6$.

These profiling results confirm that OS resolves both the arithmetic complexity bottleneck and the physical memory footprint of sheaf hypergraph learning, transforming sheaf methods into computationally scalable tools for large-scale polyadic data.

F Full Results

Table 7: Node classification performance on the directed benchmark of Mule et al. (2026). Mean accuracy ± standard deviation over 10 random seeds. Colors mark the first, second, and third result per column. † denotes statistical significance over the best baseline ($p < 0.05$, Welch's t-test with Holm-Bonferroni correction).

Method	Roman-empire	Squirrel	email-EU	Telegram	Chameleon	email-Enron
HGNN	38.44 ± 0.44	35.47 ± 1.44	48.91 ± 3.11	51.73 ± 3.38	39.98 ± 2.28	52.85 ± 7.27
HNHN	46.07 ± 1.22	35.62 ± 1.30	29.68 ± 1.68	38.22 ± 6.95	35.81 ± 3.23	18.64 ± 6.90
UniGCNII	78.89 ± 0.51	38.28 ± 2.56	44.98 ± 2.69	51.73 ± 5.05	39.85 ± 3.19	47.43 ± 7.47
LEGCN	65.60 ± 0.41	39.18 ± 1.54	32.91 ± 1.83	45.38 ± 4.23	39.29 ± 2.04	37.03 ± 7.16
HyperND	68.31 ± 0.69	40.13 ± 1.85	32.79 ± 2.90	44.62 ± 5.49	44.95 ± 3.20	38.11 ± 7.69
AllDeepSets	81.79 ± 0.72	40.69 ± 1.90	37.37 ± 6.29	49.19 ± 6.73	42.97 ± 3.60	37.29 ± 7.90
AllSetTransformer	83.53 ± 0.64	40.53 ± 1.33	38.26 ± 3.57	66.92 ± 4.36	43.85 ± 5.42	63.78 ± 3.66
ED-HNN	83.82 ± 0.31	39.85 ± 1.79	68.91 ± 4.00	60.38 ± 3.86	44.67 ± 2.33	51.35 ± 6.04
SheafHyperGNN	74.50 ± 0.57	42.01 ± 1.11	52.78 ± 9.13	70.00 ± 5.32	41.06 ± 4.94	63.51 ± 5.95
PhenomNN	71.22 ± 0.45	39.45 ± 2.19	37.69 ± 4.40	47.69 ± 6.59	43.62 ± 4.29	47.02 ± 6.75
GeDi-HNN	83.87 ± 0.63	43.02 ± 3.00	52.31 ± 2.84	77.12 ± 4.82	39.29 ± 2.04	50.54 ± 5.80
DHGNN (Jiang et al., 2019)	77.58 ± 0.54	39.85 ± 1.79	32.35 ± 2.93	79.62 ± 5.78	44.08 ± 4.11	42.16 ± 8.04
DHGNN (w/ emb.)	22.50 ± 0.81	40.33 ± 1.42	55.10 ± 3.48	80.58 ± 3.89	40.85 ± 2.76	58.38 ± 7.57
DSHN	OOM	43.55 ± 2.87	78.62 ± 2.50	**88.65 ± 5.54**	**47.02 ± 4.35**	**75.68 ± 3.42**
DSHNLight	89.24 ± 0.57	44.09 ± 2.36	**82.67 ± 1.29**	81.15 ± 4.19	**46.50 ± 4.09**	**76.76 ± 2.48**
Ours						
OmniPhaseSheaf	88.00 ± 0.06	42.23 ± 0.16	63.43 ± 7.09	87.82 ± 3.95	44.84 ± 3.50	66.67 ± 4.59
OmniPhaseSheaf-T	**$93.08^{\dagger} \pm 0.18$**	**45.19 ± 1.99**	**$85.56^{\dagger} \pm 0.83$**	85.26 ± 1.81	45.49 ± 1.24	**74.77 ± 5.55**
OmniCopreSheaf	**$92.61^{\dagger} \pm 0.49$**	**45.15 ± 2.11**	71.66 ± 0.99	**88.46 ± 1.57**	**46.46 ± 4.76**	60.36 ± 4.59
OmniCopreSheaf-T	**$93.35^{\dagger} \pm 0.27$**	**45.35 ± 0.72**	76.65 ± 2.25	**89.10 ± 4.80**	45.60 ± 4.83	64.86 ± 7.96
OmniDeepSheaf	85.20 ± 0.78	42.68 ± 1.64	64.10 ± 2.02	66.67 ± 4.80	45.02 ± 2.58	64.86 ± 2.21
OmniSheafTransformer	86.61 ± 0.24	44.60 ± 1.17	**80.43 ± 0.76**	67.95 ± 2.40	45.94 ± 4.05	67.57 ± 6.62

Table 8: Node classification performance on the directed benchmark of Mule et al. (2026), continued. Mean accuracy ± standard deviation over 10 random seeds. Colors mark the first, second, and third result per column.

Method	Cornell	Wisconsin	Amazon-ratings	Texas	Citeseer	Cora
HGNN	43.51 ± 6.44	51.56 ± 6.68	46.20 ± 0.45	52.77 ± 7.48	76.02 ± 0.81	87.25 ± 1.01
HNHN	43.51 ± 6.09	49.60 ± 4.96	42.29 ± 0.34	58.11 ± 3.87	71.24 ± 0.66	78.16 ± 0.98
UniGCNII	73.24 ± 5.19	86.86 ± 4.30	49.12 ± 0.46	81.35 ± 5.33	77.30 ± 1.15	87.53 ± 1.06
LEGCN	75.14 ± 5.51	84.71 ± 4.00	47.02 ± 0.59	81.35 ± 4.26	72.62 ± 1.09	74.96 ± 0.94
HyperND	75.14 ± 5.38	86.67 ± 5.02	47.33 ± 0.51	**83.51 ± 5.19**	75.21 ± 1.37	78.48 ± 1.02
AllDeepSets	77.83 ± 3.78	**87.84 ± 3.69**	51.91 ± 0.68	82.76 ± 5.74	75.78 ± 0.94	86.86 ± 0.85
AllSetTransformer	75.94 ± 2.97	86.27 ± 3.92	**52.28 ± 0.67**	82.76 ± 5.07	75.61 ± 1.44	86.73 ± 1.13
ED-HNN	76.49 ± 4.53	85.09 ± 4.89	51.58 ± 0.53	80.00 ± 5.05	74.95 ± 1.27	86.94 ± 1.25
SheafHyperGNN	74.59 ± 4.39	85.29 ± 4.74	48.90 ± 0.59	80.00 ± 2.48	77.21 ± 1.44	87.15 ± 0.64
PhenomNN	72.16 ± 4.19	80.58 ± 6.10	48.81 ± 0.37	81.49 ± 4.95	77.21 ± 1.32	**88.12 ± 0.86**
GeDi-HNN	78.37 ± 3.19	87.45 ± 3.41	49.30 ± 0.52	82.55 ± 4.64	75.94 ± 0.95	85.16 ± 0.94
DHGNN	77.30 ± 4.05	87.45 ± 3.84	**52.48 ± 0.50**	83.24 ± 5.64	74.67 ± 1.21	83.16 ± 1.33
DHGNN (w/ emb.)	51.08 ± 4.43	59.80 ± 5.63	**53.64 ± 0.52**	63.51 ± 9.84	56.78 ± 1.32	73.12 ± 1.04
DSHN	79.19 ± 4.37	**88.63 ± 3.49**	OOM	**83.78 ± 5.13**	77.39 ± 1.04	**87.84 ± 0.90**
DSHNLight	79.19 ± 3.20	87.25 ± 4.90	50.94 ± 0.68	82.43 ± 5.44	**77.45 ± 0.74**	**88.02 ± 1.11**
Ours						
OmniPhaseSheaf	**80.18 ± 1.27**	**87.58 ± 4.03**	49.31 ± 0.42	**84.68 ± 5.10**	**77.54 ± 0.69**	85.82 ± 1.71
OmniPhaseSheaf-T	77.48 ± 1.27	**87.58 ± 4.03**	49.90 ± 0.83	82.88 ± 1.27	77.42 ± 0.89	86.56 ± 1.46
OmniCopreSheaf	**81.08 ± 6.62**	86.27 ± 2.77	46.59 ± 1.21	**84.68 ± 3.37**	76.97 ± 1.67	86.56 ± 1.07
OmniCopreSheaf-T	**80.18 ± 4.59**	**87.58 ± 5.15**	48.63 ± 0.90	**84.68 ± 3.37**	76.97 ± 0.77	86.71 ± 0.63
OmniDeepSheaf	**80.18 ± 2.55**	86.93 ± 2.45	51.61 ± 0.73	**84.68 ± 5.10**	**77.62 ± 1.38**	86.95 ± 0.57
OmniSheafTransformer	**79.28 ± 1.27**	86.27 ± 4.80	48.59 ± 0.81	**83.78 ± 4.41**	77.38 ± 0.50	87.00 ± 0.43

Table 9: Accuracy on the synthetic directed hypergraphs of Fiorini et al. (2026), where I_o is the number of inter-class directed hyperedges per class pair. Mean accuracy ± standard deviation over 10 random seeds. Colors mark the first, second, and third result per column. † denotes statistical significance over the best baseline ($p < 0.05$, Welch's t-test with Holm-Bonferroni correction).

Method	$I_o = 10$	$I_o = 30$	$I_o = 50$
HGNN	47.12 ± 5.37	43.44 ± 6.63	37.76 ± 7.72
HNHN	20.40 ± 2.93	28.88 ± 9.45	19.76 ± 3.85
UniGCNII	21.44 ± 4.33	21.12 ± 2.95	19.84 ± 2.34
LEGCN	17.60 ± 2.43	20.72 ± 3.48	19.60 ± 2.82
HyperND	20.40 ± 2.93	21.12 ± 3.20	20.64 ± 1.92
AllDeepSets	44.40 ± 6.81	32.32 ± 4.82	31.70 ± 5.92
AllSetTransformer	21.12 ± 3.79	43.68 ± 8.72	31.84 ± 3.31
ED-HNN	34.00 ± 6.05	18.88 ± 2.56	32.48 ± 6.17
SheafHyperGNN	30.64 ± 5.39	27.28 ± 7.31	26.00 ± 9.59
PhenomNN	22.24 ± 4.73	22.08 ± 4.20	18.72 ± 3.22
GeDi-HNN	71.44 ± 3.14	71.84 ± 3.31	78.24 ± 5.64
DHGNN	40.72 ± 4.55	51.68 ± 3.97	35.76 ± 3.70
DHGNN (w/ emb.)	84.48 ± 3.22	85.28 ± 3.32	81.12 ± 3.22
DSHN	94.96 ± 1.75	**97.84 ± 1.86**	95.84 ± 2.17
DSHNLight	95.60 ± 2.15	97.04 ± 2.79	**99.04 ± 0.86**
Ours			
OmniPhaseSheaf	**97.33 ± 1.51**	**100.00† ± 0.00**	**100.00† ± 0.00**
OmniPhaseSheaf-T	**98.67† ± 1.00**	**100.00† ± 0.00**	**100.00† ± 0.00**
OmniCopreSheaf	**98.67† ± 0.38**	**100.00† ± 0.00**	**100.00† ± 0.00**
OmniCopreSheaf-T	**98.93† ± 0.38**	**99.47 ± 0.38**	**99.73 ± 0.38**
OmniDeepSheaf	72.80 ± 8.84	54.40 ± 6.53	38.93 ± 7.57
OmniSheafTransformer	88.80 ± 2.99	69.60 ± 13.50	86.67 ± 5.89

Table 10: Hyperedge classification performance on the molecular reaction datasets of Mule et al. (2026). Mean test F1-score ± standard deviation over 10 random seeds. Colors mark the first, second, and third result per column. † denotes statistical significance over the best baseline ($p < 0.05$, Welch's t-test with Holm-Bonferroni correction).

Method	Molecular-1	Molecular-2
HGNN	69.38 ± 0.48	81.40 ± 2.68
HNHN	32.27 ± 1.30	45.69 ± 7.48
UniGCNII	72.00 ± 0.59	85.61 ± 2.63
LEGCN	OOM	84.75 ± 2.68
HyperND	44.16 ± 1.27	82.86 ± 3.17
AllDeepSets	79.17 ± 0.53	85.78 ± 3.01
AllSetTransformer	79.24 ± 1.08	87.89 ± 2.87
ED-HNN	66.37 ± 2.62	87.05 ± 1.96
SheafHyperGNN	57.99 ± 2.75	80.25 ± 2.20
PhenomNN	47.71 ± 2.90	86.27 ± 2.40
GeDi-HNN	80.72 ± 0.78	85.64 ± 2.42
DHGNN	OOM	85.93 ± 3.49
DHGNN (w/ emb.)	–	–
DSHN	OOM	89.09 ± 3.08
DSHNLight	82.32 ± 0.56	86.52 ± 2.68
Ours		
OmniPhaseSheaf	73.66 ± 1.14	87.70 ± 1.94
OmniPhaseSheaf-T	$\mathbf{91.11^{\dagger} \pm 0.30}$	$\mathbf{90.53 \pm 1.62}$
OmniCopreSheaf	$\mathbf{86.66^{\dagger} \pm 0.28}$	88.59 ± 1.83
OmniCopreSheaf-T	$\mathbf{90.40^{\dagger} \pm 0.11}$	88.62 ± 2.46
OmniDeepSheaf	79.44 ± 0.46	$\mathbf{90.10 \pm 1.66}$
OmniSheafTransformer	82.40 ± 0.44	$\mathbf{90.44 \pm 2.21}$

Table 11: Classification accuracy on the undirected hypergraph benchmark of Duta et al. (2023). Mean accuracy ± standard deviation over 10 random seeds. Colors mark the first, second, and third result per column. † denotes statistical significance over the best baseline ($p < 0.05$, Welch's t-test with Holm-Bonferroni correction).

Method	Cora	Citeseer	Pubmed	Cora-CA	DBLP-CA	Senate	House	Congress
HCHA	79.14 ± 1.02	72.42 ± 1.42	86.41 ± 0.36	82.55 ± 0.97	90.92 ± 0.22	48.62 ± 4.41	61.36 ± 2.53	90.43 ± 1.20
HNHN	76.36 ± 1.92	72.64 ± 1.57	86.90 ± 0.30	77.19 ± 1.49	86.78 ± 0.29	50.93 ± 6.33	67.80 ± 2.59	53.35 ± 1.45
AllDeepSets	76.88 ± 1.80	70.83 ± 1.63	$\mathbf{88.75 \pm 0.33}$	81.97 ± 1.50	91.27 ± 0.27	48.17 ± 5.67	67.82 ± 2.40	91.80 ± 1.53
AllSetTransformer	78.58 ± 1.47	73.08 ± 1.20	$\mathbf{88.72 \pm 0.37}$	$\mathbf{83.63 \pm 1.47}$	91.53 ± 0.23	51.83 ± 5.22	69.33 ± 2.20	92.16 ± 1.05
UniGCNII	78.81 ± 1.05	73.05 ± 2.21	88.25 ± 0.33	83.60 ± 1.14	$\mathbf{91.69 \pm 0.19}$	49.30 ± 4.25	67.25 ± 2.57	$\mathbf{94.81 \pm 0.81}$
HyperND	79.20 ± 1.14	72.62 ± 1.49	86.68 ± 0.43	80.62 ± 1.32	90.35 ± 0.26	52.82 ± 3.20	51.70 ± 3.37	74.63 ± 3.62
ED-HNN	$\mathbf{80.31 \pm 1.35}$	73.70 ± 1.38	$\mathbf{89.03 \pm 0.53}$	$\mathbf{83.97 \pm 1.55}$	$\mathbf{91.90 \pm 0.19}$	64.79 ± 5.14	72.45 ± 2.28	$\mathbf{95.00 \pm 0.99}$
HyperGCN	78.36 ± 2.01	71.01 ± 2.21	80.81 ± 12.40	79.50 ± 2.11	89.42 ± 0.16	51.13 ± 4.15	69.29 ± 2.05	89.67 ± 1.22
SheafHyperGCN	$\mathbf{80.06 \pm 1.12}$	73.27 ± 0.50	87.09 ± 0.71	83.26 ± 1.20	90.83 ± 0.23	66.33 ± 4.58	72.66 ± 2.26	90.37 ± 1.52
HyperGNN	79.39 ± 1.36	72.45 ± 1.16	86.44 ± 0.44	82.64 ± 1.65	91.03 ± 0.20	48.59 ± 4.52	61.39 ± 2.96	91.26 ± 1.15
SheafHyperGNN	$\mathbf{81.30 \pm 1.70}$	$\mathbf{74.71 \pm 1.23}$	87.68 ± 0.60	$\mathbf{85.52 \pm 1.28}$	91.59 ± 0.24	$\mathbf{68.73 \pm 4.68}$	$\mathbf{73.84 \pm 2.30}$	91.81 ± 1.60
Ours								
OmniDeepSheaf	78.73 ± 1.15	$\mathbf{74.72 \pm 0.92}$	87.86 ± 0.13	83.26 ± 0.62	91.64 ± 0.16	$\mathbf{83.10^{\dagger} \pm 1.99}$	$\mathbf{78.33^{\dagger} \pm 1.26}$	94.73 ± 0.58
OmniSheafTransformer	79.76 ± 1.81	$\mathbf{74.52 \pm 0.65}$	87.82 ± 0.35	83.21 ± 1.00	$\mathbf{91.70 \pm 0.21}$	$\mathbf{81.69^{\dagger} \pm 2.30}$	$\mathbf{78.43^{\dagger} \pm 1.17}$	$\mathbf{95.58 \pm 1.16}$

Table 12: Hyperparameters of every reported OmniSheaf result. $\mathcal{F}$ is the restriction-map family (Diag, General or Ortho); d the stalk dimension; w the hidden width `MLP_hidden`; L_{conv} the number of V→E→V layers; L_{MLP} the depth of the per-stalk encoder/decoder; C_h the classifier width; n_h the number of PMA heads (if available); IN input normalization; p feature dropout; $p_{\mathcal{F}}$ dropout on the restriction maps; $\sigma_{\mathcal{F}}$ the activation applied to them; Res the residual connection; q the magnetic charge (phase variants only).

Dataset	**Model**	$\mathcal{F}$	d	w	L_{conv}	L_{MLP}	C_h	H	IN	p	$p_{\mathcal{F}}$	$\sigma_{\mathcal{F}}$	Res	lr	wd	q
Directed node classification																
Roman-empire	OmniPhaseSheaf	General	4	64	3	1	256	–	✓	0.1	–	none	✓	0.005	0	0.15
Roman-empire	OmniPhaseSheaf-T	Ortho	3	64	2	1	128	4	✓	0.1	✓	none	–	0.02	0.0005	0.15
Roman-empire	OmniCopreSheaf	General	2	64	3	1	128	–	✓	0.1	–	none	✓	0.001	5e-05	–
Roman-empire	OmniCopreSheaf-T	Ortho	4	64	2	2	64	8	–	0.1	✓	none	✓	0.001	0.0005	–
Roman-empire	OmniDeepSheaf	Ortho	2	128	3	2	256	–	–	0.1	–	none	✓	0.001	0	–
Roman-empire	OmniSheafTransformer	General	4	64	3	2	64	4	–	0.5	–	tanh	✓	0.02	5e-05	–
Squirrel	OmniPhaseSheaf	General	6	32	5	1	64	–	✓	0.5	–	sigmoid	✓	0.001	0	0.25
Squirrel	OmniPhaseSheaf-T	Diag	2	64	2	1	32	8	✓	0.6	–	none	✓	0.02	0.0005	0.05
Squirrel	OmniCopreSheaf	Diag	2	512	4	1	64	–	✓	0.2	✓	tanh	✓	0.02	0.0005	–
Squirrel	OmniCopreSheaf-T	General	1	64	1	2	64	1	✓	0.4	✓	tanh	✓	0.005	5e-05	–
Squirrel	OmniDeepSheaf	General	6	32	3	1	32	–	–	0.6	✓	tanh	✓	0.001	0	–
Squirrel	OmniSheafTransformer	General	2	128	3	1	128	4	✓	0.5	–	none	✓	0.005	0.0005	–
email-EU	OmniPhaseSheaf	Diag	5	256	2	1	64	–	✓	0.2	–	sigmoid	–	0.001	0	0.25
email-EU	OmniPhaseSheaf-T	Diag	3	32	2	1	128	4	–	0.2	–	tanh	–	0.005	5e-05	0.1
email-EU	OmniCopreSheaf	Ortho	5	64	2	1	128	–	✓	0	✓	sigmoid	✓	0.001	0	–
email-EU	OmniCopreSheaf-T	Ortho	6	32	2	2	128	4	✓	0	✓	none	–	0.001	0	–
email-EU	OmniDeepSheaf	Ortho	3	256	2	2	32	–	✓	0.1	–	none	✓	0.001	0	–
email-EU	OmniSheafTransformer	Diag	2	128	2	2	256	8	✓	0.2	–	none	–	0.001	5e-05	–
Telegram	OmniPhaseSheaf	Ortho	5	32	4	1	64	–	–	0	–	sigmoid	–	0.001	0.0005	0.2
Telegram	OmniPhaseSheaf-T	Ortho	3	64	2	2	64	8	✓	0.6	–	tanh	✓	0.005	0.0005	0.15
Telegram	OmniCopreSheaf	Ortho	6	128	2	1	32	–	–	0.6	–	sigmoid	–	0.001	0.0005	–
Telegram	OmniCopreSheaf-T	Ortho	4	256	3	1	128	8	✓	0.7	–	none	–	0.001	0	–
Telegram	OmniDeepSheaf	Ortho	5	256	4	1	256	–	–	0	✓	none	–	0.001	0	–
Telegram	OmniSheafTransformer	Ortho	3	32	3	1	32	4	✓	0.2	–	sigmoid	✓	0.001	5e-05	–
Chameleon	OmniPhaseSheaf	General	6	32	3	2	32	–	–	0.7	–	sigmoid	✓	0.01	5e-05	0.05
Chameleon	OmniPhaseSheaf-T	Ortho	2	64	2	1	32	8	–	0.4	✓	sigmoid	✓	0.01	0.0005	0.1
Chameleon	OmniCopreSheaf	Ortho	5	512	1	1	64	–	✓	0.6	✓	sigmoid	✓	0.02	0.0005	–
Chameleon	OmniCopreSheaf-T	Ortho	4	128	2	2	32	8	✓	0.5	–	tanh	✓	0.01	0.0005	–
Chameleon	OmniDeepSheaf	General	6	32	1	2	32	–	✓	0.3	–	none	✓	0.01	0	–
Chameleon	OmniSheafTransformer	Ortho	4	64	3	2	32	1	–	0.5	–	none	✓	0.02	0	–
email-Enron	OmniPhaseSheaf	Ortho	3	64	5	1	64	–	–	0.4	–	sigmoid	–	0.001	0	0.1
email-Enron	OmniPhaseSheaf-T	Ortho	2	32	3	2	256	4	–	0.3	–	none	–	0.001	5e-05	0.1
email-Enron	OmniCopreSheaf	Ortho	5	32	2	1	256	–	✓	0.2	–	tanh	–	0.001	0	–
email-Enron	OmniCopreSheaf-T	Ortho	4	32	3	1	32	8	✓	0	–	none	✓	0.001	0	–
email-Enron	OmniDeepSheaf	Ortho	3	64	4	1	128	–	✓	0	–	none	✓	0.001	0.0005	–
email-Enron	OmniSheafTransformer	Ortho	5	64	2	1	64	8	–	0.2	✓	none	✓	0.001	0.0005	–
Cornell	OmniPhaseSheaf	Diag	4	32	3	1	256	–	✓	0.5	✓	tanh	✓	0.005	5e-05	0.25
Cornell	OmniPhaseSheaf-T	Diag	5	32	1	1	64	8	✓	0.2	–	sigmoid	✓	0.01	0.0005	0.15
Cornell	OmniCopreSheaf	General	1	128	1	2	256	–	–	0.6	✓	none	✓	0.02	0.0005	–
Cornell	OmniCopreSheaf-T	General	4	256	2	2	64	4	✓	0.3	✓	tanh	✓	0.01	0.0005	–
Cornell	OmniDeepSheaf	Ortho	5	64	2	1	128	–	–	0.6	✓	sigmoid	✓	0.02	0.0005	–
Cornell	OmniSheafTransformer	Ortho	5	256	3	2	128	4	–	0.3	✓	sigmoid	✓	0.005	5e-05	–
Wisconsin	OmniPhaseSheaf	Ortho	3	512	2	1	64	–	✓	0.3	–	tanh	✓	0.005	0.0005	0.1
Wisconsin	OmniPhaseSheaf-T	General	2	32	2	2	64	4	–	0.5	✓	none	✓	0.005	0	0.25
Wisconsin	OmniCopreSheaf	General	6	512	3	1	32	–	–	0.6	–	sigmoid	✓	0.001	0.0005	–
Wisconsin	OmniCopreSheaf-T	General	5	256	3	2	32	8	–	0.4	–	sigmoid	✓	0.01	0.0005	–
Wisconsin	OmniDeepSheaf	Ortho	6	128	1	2	32	–	–	0.5	–	tanh	✓	0.005	0.0005	–
Wisconsin	OmniSheafTransformer	General	6	256	1	2	128	4	✓	0.6	✓	sigmoid	✓	0.01	0.0005	–
Amazon-ratings	OmniPhaseSheaf	Diag	3	64	3	2	64	–	–	0.1	✓	sigmoid	✓	0.005	0	0.15
Amazon-ratings	OmniPhaseSheaf-T	Diag	3	64	3	1	128	8	–	0.3	✓	tanh	✓	0.02	5e-05	0.15
Amazon-ratings	OmniCopreSheaf	Diag	4	128	3	1	128	–	–	0.3	✓	tanh	✓	0.001	5e-05	–
Amazon-ratings	OmniCopreSheaf-T	General	4	128	3	2	64	4	✓	0.3	–	tanh	✓	0.001	5e-05	–
Amazon-ratings	OmniDeepSheaf	Ortho	3	128	2	1	256	–	–	0.1	–	sigmoid	✓	0.001	5e-05	–
Amazon-ratings	OmniSheafTransformer	Ortho	2	128	2	2	256	8	–	0.4	–	sigmoid	✓	0.001	0	–
Texas	OmniPhaseSheaf	Ortho	2	256	1	2	128	–	–	0.1	✓	none	✓	0.02	0.0005	0.1
Texas	OmniPhaseSheaf-T	General	6	32	1	2	64	1	✓	0.3	–	tanh	✓	0.01	5e-05	0.25
Texas	OmniCopreSheaf	General	2	32	2	2	32	–	–	0.1	–	sigmoid	✓	0.02	0.0005	–
Texas	OmniCopreSheaf-T	Ortho	5	128	1	1	64	1	–	0.7	–	sigmoid	✓	0.02	0.0005	–
Texas	OmniDeepSheaf	General	5	32	3	2	128	–	–	0.1	–	sigmoid	✓	0.02	0.0005	–
Texas	OmniSheafTransformer	General	3	64	3	1	128	1	✓	0.4	–	sigmoid	✓	0.01	0.0005	–
Citeseer	OmniPhaseSheaf	Ortho	5	512	4	1	64	–	✓	0.5	–	sigmoid	✓	0.001	0	0.2
Citeseer	OmniPhaseSheaf-T	Ortho	3	256	1	1	256	8	–	0.3	✓	tanh	✓	0.001	0	0.1
Citeseer	OmniCopreSheaf	General	6	128	4	1	32	–	–	0.6	–	tanh	✓	0.001	0.0005	–
Citeseer	OmniCopreSheaf-T	Ortho	6	64	1	1	128	1	✓	0.3	✓	sigmoid	✓	0.001	0	–
Citeseer	OmniDeepSheaf	Ortho	4	512	1	1	64	–	✓	0.6	✓	sigmoid	✓	0.001	0	–
Citeseer	OmniSheafTransformer	General	1	32	2	2	128	4	✓	0.2	✓	sigmoid	✓	0.01	0.0005	–
Cora	OmniPhaseSheaf	Diag	4	32	5	1	128	–	✓	0.5	–	tanh	✓	0.01	0.0005	0
Cora	OmniPhaseSheaf-T	General	1	128	1	2	64	1	✓	0.3	✓	sigmoid	–	0.001	5e-05	0
Cora	OmniCopreSheaf	Diag	2	512	2	1	64	–	✓	0.1	–	sigmoid	–	0.001	0.0005	–

continues on the next page

Table 12, continued

Dataset	Model	$\mathcal{F}$	d	w	L_{conv}	L_{MLP}	C_h	H	IN	p	$p_{\mathcal{F}}$	$\sigma_{\mathcal{F}}$	Res	lr	wd	q
Cora	OmniCopreSheaf-T	Ortho	2	128	1	1	64	8	–	0.5	✓	tanh	–	0.001	5e-05	–
Cora	OmniDeepSheaf	Ortho	3	128	3	1	256	–	✓	0.3	✓	sigmoid	✓	0.005	0.0005	–
Cora	OmniSheafTransformer	General	4	32	3	2	256	8	✓	0.4	✓	tanh	✓	0.001	0.0005	–
Synthetic directed hypergraphs																
$I_o = 10$	OmniPhaseSheaf	Ortho	5	128	3	1	32	–	✓	0.4	–	none	–	0.001	5e-05	0.2
$I_o = 10$	OmniPhaseSheaf-T	Ortho	3	32	3	1	32	8	–	0.1	✓	sigmoid	✓	0.001	0	0.2
$I_o = 10$	OmniCopreSheaf	Ortho	2	256	5	1	64	–	✓	0.3	–	sigmoid	✓	0.001	5e-05	–
$I_o = 10$	OmniCopreSheaf-T	Ortho	6	64	3	2	256	8	✓	0.4	–	none	✓	0.001	0	–
$I_o = 10$	OmniDeepSheaf	Ortho	4	64	4	1	64	–	✓	0	–	sigmoid	✓	0.001	0.0005	–
$I_o = 10$	OmniSheafTransformer	Ortho	5	32	2	1	32	4	–	0.3	–	none	–	0.001	0	–
$I_o = 30$	OmniPhaseSheaf	Ortho	3	256	3	2	32	–	–	0.1	✓	none	✓	0.001	5e-05	0.25
$I_o = 30$	OmniPhaseSheaf-T	Ortho	6	128	3	1	64	8	✓	0.5	–	sigmoid	–	0.001	0	0.05
$I_o = 30$	OmniCopreSheaf	Ortho	6	128	3	1	256	–	✓	0.4	–	tanh	–	0.02	5e-05	–
$I_o = 30$	OmniCopreSheaf-T	Ortho	2	64	3	2	128	8	✓	0.3	✓	tanh	✓	0.005	0.0005	–
$I_o = 30$	OmniDeepSheaf	Diag	3	64	5	2	128	–	–	0	–	tanh	✓	0.001	0	–
$I_o = 30$	OmniSheafTransformer	Ortho	2	256	3	1	128	8	–	0	✓	sigmoid	✓	0.001	0	–
$I_o = 50$	OmniPhaseSheaf	Ortho	3	512	3	1	64	–	–	0.2	–	none	–	0.005	5e-05	0.25
$I_o = 50$	OmniPhaseSheaf-T	Ortho	5	64	3	1	256	8	–	0.1	–	sigmoid	✓	0.001	0.0005	0.1
$I_o = 50$	OmniCopreSheaf	General	5	32	2	1	32	–	✓	0.1	✓	none	–	0.01	0.0005	–
$I_o = 50$	OmniCopreSheaf-T	Ortho	2	32	3	1	256	8	–	0.3	✓	sigmoid	✓	0.02	5e-05	–
$I_o = 50$	OmniDeepSheaf	Ortho	4	128	5	2	32	–	✓	0.8	–	sigmoid	–	0.001	5e-05	–
$I_o = 50$	OmniSheafTransformer	General	6	32	3	1	128	8	–	0.1	–	tanh	✓	0.001	0	–
Molecular hyperedge classification (F1)																
Molecular-1	OmniPhaseSheaf	Diag	4	64	1	2	32	–	✓	0	✓	sigmoid	✓	0.005	5e-05	0.1
Molecular-1	OmniPhaseSheaf-T	Diag	3	64	2	2	128	1	✓	0.2	✓	tanh	–	0.001	5e-05	0.25
Molecular-1	OmniCopreSheaf	Ortho	4	128	1	2	32	–	–	0.2	–	tanh	✓	0.001	5e-05	–
Molecular-1	OmniCopreSheaf-T	Ortho	6	64	1	1	128	1	–	0.6	✓	none	–	0.001	5e-05	–
Molecular-1	OmniDeepSheaf	Ortho	5	64	1	1	64	–	✓	0.2	–	none	✓	0.01	0	–
Molecular-1	OmniSheafTransformer	Diag	3	64	2	1	256	8	–	0.1	–	none	✓	0.005	5e-05	–
Molecular-2	OmniPhaseSheaf	Diag	3	512	1	1	64	–	–	0.3	✓	tanh	–	0.001	5e-05	0.25
Molecular-2	OmniPhaseSheaf-T	Ortho	4	32	3	2	32	4	–	0.5	–	sigmoid	–	0.001	5e-05	0.05
Molecular-2	OmniCopreSheaf	Diag	6	64	1	1	128	–	✓	0	✓	none	–	0.001	0	–
Molecular-2	OmniCopreSheaf-T	Diag	5	64	1	1	256	4	✓	0.6	✓	none	–	0.001	0.0005	–
Molecular-2	OmniDeepSheaf	Ortho	6	256	1	1	256	–	✓	0.2	–	tanh	–	0.001	0.0005	–
Molecular-2	OmniSheafTransformer	Ortho	5	128	3	1	256	4	–	0.3	✓	sigmoid	✓	0.001	5e-05	–
Undirected benchmark (Duta construction)																
Cora	OmniDeepSheaf	General	3	512	3	1	256	–	✓	0	✓	sigmoid	✓	0.001	0	–
Cora	OmniSheafTransformer	Diag	4	256	1	2	64	4	–	0.8	–	sigmoid	✓	0.001	0.0005	–
Citeseer	OmniDeepSheaf	Ortho	5	128	1	2	128	–	–	0.7	–	tanh	✓	0.001	5e-05	–
Citeseer	OmniSheafTransformer	Ortho	4	32	1	2	32	4	–	0.5	✓	sigmoid	–	0.001	0	–
Pubmed	OmniDeepSheaf	Diag	5	256	1	1	256	–	–	0.2	–	tanh	–	0.02	0.0005	–
Pubmed	OmniSheafTransformer	Diag	5	32	3	1	32	4	–	0.3	–	sigmoid	✓	0.02	0.0005	–
Cora-CA	OmniDeepSheaf	Diag	5	128	1	1	64	–	✓	0.2	–	sigmoid	✓	0.001	0	–
Cora-CA	OmniSheafTransformer	General	1	256	2	2	256	1	–	0.1	–	sigmoid	✓	0.001	5e-05	–
DBLP-CA	OmniDeepSheaf	Ortho	4	128	1	1	256	–	✓	0.1	✓	tanh	–	0.001	5e-05	–
DBLP-CA	OmniSheafTransformer	Ortho	2	128	2	1	32	1	–	0.4	–	tanh	✓	0.001	5e-05	–
Senate	OmniDeepSheaf	Ortho	4	32	1	1	128	–	✓	0	–	none	–	0.005	5e-05	–
Senate	OmniSheafTransformer	General	3	256	1	1	32	8	–	0	–	tanh	–	0.01	0	–
House	OmniDeepSheaf	Ortho	6	256	1	1	256	–	–	0.9	–	tanh	✓	0.005	0	–
House	OmniSheafTransformer	General	4	64	2	2	128	8	✓	0.2	–	sigmoid	✓	0.001	5e-05	–
Congress	OmniDeepSheaf	Diag	3	64	2	1	32	–	✓	0.2	–	none	✓	0.005	0.0005	–
Congress	OmniSheafTransformer	General	4	32	2	1	128	8	✓	0.5	✓	tanh	✓	0.005	0	–

TRAINING TIME PER STEP

Table 13: Training time per step (ms) on CORA (Diag restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			7.218 ± 0.03 (no stalk)			
AllSetTransformer			11.8 ± 0.18 (no stalk)			
Ours						
AllDeepSets			7.218 ± 0.03 (no stalk)			
AllSetTransformer			11.8 ± 0.18 (no stalk)			
SheafHyperGNN	**36.23 ± 0.55**	**45.26 ± 0.16**	**59.46 ± 0.27**	**79.95 ± 1.1**	**101.4 ± 1.3**	**130.2 ± 0.29**
SheafHyperGCN	70.63 ± 0.12	76.33 ± 0.82	78.64 ± 0.29	**81.68 ± 0.14**	**84.89 ± 0.48**	**90.0 ± 0.09**
OmniDeepSheaf	**11.8 ± 0.03**	**22.18 ± 0.28**	**30.35 ± 0.33**	**39.2 ± 0.38**	**49.1 ± 1.2**	**59.37 ± 1.5**
OmniSheafTransformer	**14.96 ± 0.16**	**35.41 ± 0.32**	**64.33 ± 1.8**	101.8 ± 1.1	144.6 ± 1.2	190.1 ± 1.5

Table 14: Training time per step (ms) on CORA (General restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			7.218 ± 0.03 (no stalk)			
AllSetTransformer			11.8 ± 0.18 (no stalk)			
Ours						
AllDeepSets			7.218 ± 0.03 (no stalk)			
AllSetTransformer			11.8 ± 0.18 (no stalk)			
SheafHyperGNN	**37.47 ± 0.27**	**75.73 ± 0.53**	243.2 ± 0.42	675.6 ± 1.1	OOM	OOM
SheafHyperGCN	71.35 ± 0.07	80.43 ± 0.22	**87.64 ± 0.25**	**96.04 ± 0.28**	**107 ± 0.57**	**119 ± 0.15**
OmniDeepSheaf	**11.7 ± 0.10**	**22.11 ± 0.13**	**30.4 ± 0.12**	**39.26 ± 0.21**	**49.47 ± 0.94**	**59.81 ± 2.0**
OmniSheafTransformer	**14.75 ± 0.02**	**35.66 ± 0.08**	**64.93 ± 2.1**	**102.1 ± 1.3**	**145.2 ± 2.3**	**189.5 ± 1.8**

Table 15: Training time per step (ms) on CORA (Ortho restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			7.218 ± 0.03 (no stalk)			
AllSetTransformer			11.8 ± 0.18 (no stalk)			
Ours						
AllDeepSets			7.218 ± 0.03 (no stalk)			
AllSetTransformer			11.8 ± 0.18 (no stalk)			
SheafHyperGNN	–	**38.16 ± 0.09**	**57.43 ± 0.23**	**95.08 ± 0.96**	159.9 ± 0.13	249.1 ± 0.18
SheafHyperGCN	–	77.0 ± 0.24	79.68 ± 0.26	**82.77 ± 0.31**	**86.9 ± 0.27**	**92.29 ± 0.26**
OmniDeepSheaf	–	**23.64 ± 0.28**	**32.57 ± 0.21**	**42.16 ± 0.34**	**53.06 ± 1.4**	**64.5 ± 1.3**
OmniSheafTransformer	–	**36.68 ± 0.20**	**66.16 ± 1.6**	104 ± 0.14	**147.6 ± 0.77**	**194.4 ± 0.85**

Table 16: Training time per step (ms) on TELEGRAM (Diag restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			6.923 ± 0.12 (no stalk)			
AllSetTransformer			11.27 ± 0.04 (no stalk)			
SheafHyperGNN	34.8 ± 0.30	39.96 ± 0.02	44.77 ± 0.33	48.52 ± 0.16	52.82 ± 0.26	57.81 ± 0.90
SheafHyperGCN	29.76 ± 0.05	49.89 ± 0.11	51.85 ± 0.09	53.76 ± 0.12	55.51 ± 0.11	**56.65 ± 0.08**
DSHN	26.51 ± 0.11	**28.75 ± 0.24**	**30.53 ± 0.06**	**32.45 ± 0.21**	**34.89 ± 0.20**	**37.52 ± 0.22**
DSHNLight	**17.98 ± 0.08**	**18.39 ± 0.15**	**18.86 ± 0.18**	**19.42 ± 0.31**	**19.92 ± 0.20**	**20.18 ± 0.13**
Ours						
OmniDeepSheaf	**12.36 ± 0.09**	**23.84 ± 0.31**	**33.18 ± 0.13**	**42.57 ± 0.16**	**52.12 ± 0.88**	61.46 ± 0.56
OmniSheafTransformer	**15.31 ± 0.11**	39.02 ± 0.52	68.12 ± 2.3	101 ± 1.8	153.9 ± 1.3	199.9 ± 0.09
OmniPhaseSheaf	28.32 ± 0.15	90.82 ± 1.1	143.7 ± 0.49	184.2 ± 0.48	226.2 ± 0.17	267.9 ± 0.33
OmniPhaseSheaf-T	57.55 ± 1.2	149.4 ± 1.0	249.8 ± 0.36	351.4 ± 1.3	483.4 ± 3.1	622 ± 2.7
OmniCopreSheaf	30.5 ± 0.25	49.25 ± 0.56	63.7 ± 0.82	75.66 ± 1.3	89.36 ± 1.6	102.9 ± 0.11
OmniCopreSheaf-T	34.94 ± 0.12	66.14 ± 0.57	102 ± 1.2	140.7 ± 1.4	199.8 ± 0.38	256.4 ± 1.1

Table 17: Training time per step (ms) on TELEGRAM (General restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			6.923 ± 0.12 (no stalk)			
AllSetTransformer			11.27 ± 0.04 (no stalk)			
SheafHyperGNN	35.03 ± 0.04	88.31 ± 1.1	388.6 ± 0.30	1165 ± 2.0	OOM	OOM
SheafHyperGCN	30.05 ± 0.16	50.75 ± 0.12	**52.71 ± 0.23**	**54.57 ± 0.13**	**56.43 ± 0.14**	**58.0 ± 0.01**
DSHN	65.45 ± 0.22	72.99 ± 0.35	110 ± 0.43	218.6 ± 1.0	433.2 ± 0.33	756.5 ± 0.52
DSHNLight	60.03 ± 0.11	62.24 ± 0.41	66.54 ± 0.27	79.14 ± 0.20	112.6 ± 0.46	173.2 ± 0.15
Ours						
OmniDeepSheaf	**12.16 ± 0.03**	**23.65 ± 0.05**	**33.15 ± 0.04**	**42.64 ± 0.23**	**52.04 ± 0.85**	**61.75 ± 1.3**
OmniSheafTransformer	**15.05 ± 0.12**	**38.65 ± 0.26**	68.19 ± 1.2	102.1 ± 0.60	155.2 ± 1.9	202.9 ± 2.7
OmniPhaseSheaf	**28.29 ± 0.08**	90.79 ± 1.1	144.4 ± 0.79	185.2 ± 0.07	227.5 ± 0.57	269.7 ± 0.88
OmniPhaseSheaf-T	52.76 ± 0.70	151.5 ± 1.8	255 ± 1.5	360 ± 0.80	460.1 ± 53	625.3 ± 69
OmniCopreSheaf	30.17 ± 0.44	**48.79 ± 0.62**	**63.04 ± 1.2**	**76.13 ± 1.1**	**89.88 ± 0.75**	**104 ± 0.43**
OmniCopreSheaf-T	34.79 ± 0.07	67.11 ± 0.75	103.5 ± 0.56	143.2 ± 2.4	206.9 ± 3.6	262.6 ± 1.0

Table 18: Training time per step (ms) on TELEGRAM (Ortho restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			6.923 ± 0.12 (no stalk)			
AllSetTransformer			11.27 ± 0.04 (no stalk)			
SheafHyperGNN	–	42.73 ± 0.50	109.9 ± 0.53	308.9 ± 0.56	738.9 ± 1.5	1400 ± 1.9
SheafHyperGCN	–	50.68 ± 0.05	**52.28 ± 0.06**	**54.3 ± 0.06**	**56.84 ± 0.58**	**57.84 ± 0.06**
DSHN	–	**32.33 ± 0.13**	67.64 ± 0.49	174.8 ± 0.26	388.4 ± 0.09	712 ± 0.66
DSHNLight	–	**20.15 ± 0.17**	**23.92 ± 0.10**	**35.03 ± 0.17**	**66.86 ± 0.68**	125.5 ± 0.27
Ours						
OmniDeepSheaf	–	**25.37 ± 0.12**	**35.55 ± 0.08**	**45.73 ± 0.46**	**56.13 ± 0.98**	**66.6 ± 0.60**
OmniSheafTransformer	–	40.05 ± 0.25	69.88 ± 1.8	104.8 ± 0.55	158.1 ± 1.3	206 ± 0.79
OmniPhaseSheaf	–	95.1 ± 0.57	147.6 ± 1.9	190.2 ± 0.22	232.3 ± 0.11	275 ± 0.49
OmniPhaseSheaf-T	–	155.2 ± 1.4	259.9 ± 0.68	364.4 ± 1.3	498.7 ± 2.8	627.3 ± 0.77
OmniCopreSheaf	–	56.23 ± 0.60	73.96 ± 1.1	89.37 ± 0.73	106.7 ± 0.13	**123.3 ± 0.33**
OmniCopreSheaf-T	–	74.36 ± 0.27	114.2 ± 1.2	157.1 ± 2.3	219.5 ± 0.87	277.8 ± 1.2

Table 19: Training time per step (ms) on ROMAN-EMPIRE (Diag restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			59.84 ± 1.3 (no stalk)			
AllSetTransformer			77.72 ± 2.2 (no stalk)			
SheafHyperGNN	160 ± 0.75	549.8 ± 8.3	1158 ± 13	OOM	OOM	OOM
SheafHyperGCN	897.4 ± 45	1005 ± 37	OOM	OOM	OOM	OOM
DSHN	505.8 ± 0.97	1817 ± 29	OOM	OOM	OOM	OOM
DSHNLight	**137.8 ± 0.02**	**286.5 ± 0.09**	**423.6 ± 0.27**	**560.2 ± 0.24**	**696.8 ± 0.16**	**835.5 ± 0.89**
Ours						
OmniDeepSheaf	**124.1 ± 1.3**	**256.7 ± 0.35**	**342.4 ± 0.63**	**425.4 ± 0.38**	**516.1 ± 1.5**	**600.3 ± 0.42**
OmniSheafTransformer	**152.7 ± 1.5**	**373.1 ± 1.1**	**610.8 ± 1.4**	881.8 ± 1.5	1299 ± 11	1684 ± 16
OmniPhaseSheaf	437.2 ± 0.05	970.4 ± 1.3	1354 ± 3.7	1733 ± 1.9	OOM	OOM
OmniPhaseSheaf-T	806.3 ± 0.13	1906 ± 0.25	OOM	OOM	OOM	OOM
OmniCopreSheaf	286.2 ± 0.28	517.3 ± 0.84	649.9 ± 1.9	**780 ± 0.31**	**919.6 ± 0.96**	**1059 ± 3.8**
OmniCopreSheaf-T	316 ± 1.0	639.5 ± 0.90	939.7 ± 1.3	1297 ± 5.0	1742 ± 5.7	OOM

Table 20: Training time per step (ms) on ROMAN-EMPIRE (General restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			59.84 ± 1.3 (no stalk)			
AllSetTransformer			77.72 ± 2.2 (no stalk)			
SheafHyperGNN	**159.9 ± 0.23**	592.5 ± 3.0	1303 ± 9.9	OOM	OOM	OOM
SheafHyperGCN	943.8 ± 42	1251 ± 46	OOM	OOM	OOM	OOM
DSHN	3554 ± 79	4917 ± 86	OOM	OOM	OOM	OOM
DSHNLight	3202 ± 74	3438 ± 80	3613 ± 25	3792 ± 97	3940 ± 35	4273 ± 139
Ours						
OmniDeepSheaf	**123.6 ± 1.1**	**256.4 ± 0.53**	**342.9 ± 1.2**	**425.1 ± 1.4**	**519.9 ± 0.41**	**599.8 ± 0.52**
OmniSheafTransformer	**150.4 ± 1.1**	**369.4 ± 1.6**	**619.8 ± 1.5**	**912.4 ± 17**	**1288 ± 11**	**1677 ± 14**
OmniPhaseSheaf	438.6 ± 0.17	979.4 ± 1.3	1360 ± 0.99	1734 ± 1.7	OOM	OOM
OmniPhaseSheaf-T	804.4 ± 0.44	1889 ± 2.9	OOM	OOM	OOM	OOM
OmniCopreSheaf	285.8 ± 0.10	**516.8 ± 1.1**	**652.7 ± 0.82**	**780.1 ± 2.6**	**919.5 ± 1.3**	**1060 ± 0.67**
OmniCopreSheaf-T	315 ± 0.51	645.2 ± 2.8	956.7 ± 3.2	1297 ± 4.3	1780 ± 11	OOM

Table 21: Training time per step (ms) on ROMAN-EMPIRE (Ortho restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			59.84 ± 1.3 (no stalk)			
AllSetTransformer			77.72 ± 2.2 (no stalk)			
SheafHyperGNN	–	549 ± 5.4	1200 ± 1.2	OOM	OOM	OOM
SheafHyperGCN	–	1004 ± 46	OOM	OOM	OOM	OOM
DSHN	–	1800 ± 14	OOM	OOM	OOM	OOM
DSHNLight	–	**292.1 ± 0.10**	**441.3 ± 0.70**	**612.1 ± 0.19**	**822.8 ± 0.45**	**1044 ± 1.3**
Ours						
OmniDeepSheaf	–	**262 ± 0.26**	**353.4 ± 1.4**	**439.6 ± 0.33**	**534 ± 2.4**	**628.4 ± 2.3**
OmniSheafTransformer	–	**378.4 ± 1.1**	**634.8 ± 5.8**	922.4 ± 2.5	1350 ± 13	1740 ± 20
OmniPhaseSheaf	–	987.5 ± 0.57	1366 ± 1.5	1750 ± 5.2	OOM	OOM
OmniPhaseSheaf-T	–	1916 ± 7.4	OOM	OOM	OOM	OOM
OmniCopreSheaf	–	539.1 ± 3.0	687.4 ± 2.8	**838 ± 2.6**	**1000 ± 1.9**	**1180 ± 0.17**
OmniCopreSheaf-T	–	670.1 ± 2.1	999.4 ± 5.5	1359 ± 4.0	1859 ± 8.3	OOM

PEAK TRAINING MEMORY

Table 22: Training peak memory (GB) on CORA (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.18 (no stalk)			
AllSetTransformer			0.22 (no stalk)			
Ours						
AllDeepSets			0.18 (no stalk)			
AllSetTransformer			0.22 (no stalk)			
SheafHyperGNN	**0.36**	**0.64**	**0.92**	**1.21**	**1.49**	**1.77**
SheafHyperGCN	**0.14**	**0.32**	**0.62**	**1.03**	**1.56**	2.20
OmniDeepSheaf	**0.39**	**0.63**	**0.86**	**1.12**	**1.34**	**1.57**
OmniSheafTransformer	0.40	0.70	1.01	1.36	1.70	**2.08**

Table 23: Training peak memory (GB) on CORA (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.18 (no stalk)			
AllSetTransformer			0.22 (no stalk)			
Ours						
AllDeepSets			0.18 (no stalk)			
AllSetTransformer			0.22 (no stalk)			
SheafHyperGNN	**0.35**	1.85	5.73	13.15	OOM	OOM
SheafHyperGCN	**0.14**	**0.33**	**0.63**	**1.04**	**1.57**	**2.22**
OmniDeepSheaf	**0.39**	**0.63**	**0.86**	**1.12**	**1.34**	**1.57**
OmniSheafTransformer	0.40	**0.70**	**1.01**	**1.36**	**1.70**	**2.08**

Table 24: Training peak memory (GB) on CORA (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			0.18 (no stalk)			
AllSetTransformer			0.22 (no stalk)			
Ours						
AllDeepSets			0.18 (no stalk)			
AllSetTransformer			0.22 (no stalk)			
SheafHyperGNN	–	**0.35**	**0.62**	**0.99**	**1.55**	2.39
SheafHyperGCN	–	**0.33**	**0.62**	**1.04**	**1.56**	**2.21**
OmniDeepSheaf	–	**0.63**	**0.86**	**1.12**	**1.34**	**1.58**
OmniSheafTransformer	–	0.70	**1.01**	1.36	1.70	**2.09**

Table 25: Training peak memory (GB) on TELEGRAM (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.07 (no stalk)			
AllSetTransformer			0.13 (no stalk)			
SheafHyperGNN	**0.28**	**0.51**	**0.73**	**0.96**	**1.19**	**1.41**
SheafHyperGCN	**0.08**	**0.17**	**0.25**	**0.35**	**0.47**	**0.61**
DSHN	0.34	0.62	0.91	1.19	1.48	1.76
DSHNLight	**0.10**	**0.12**	**0.15**	**0.18**	**0.21**	**0.24**
Ours						
OmniDeepSheaf	0.40	0.67	0.92	1.17	1.42	1.69
OmniSheafTransformer	0.41	0.72	1.03	1.38	1.74	2.18
OmniPhaseSheaf	0.72	1.30	1.87	2.45	3.00	3.58
OmniPhaseSheaf-T	0.91	1.71	2.53	3.36	4.17	5.07
OmniCopreSheaf	0.78	1.08	1.36	1.66	1.95	2.26
OmniCopreSheaf-T	0.78	1.13	1.50	1.93	2.39	2.93

Table 26: Training peak memory (GB) on TELEGRAM (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.07 (no stalk)			
AllSetTransformer			0.13 (no stalk)			
SheafHyperGNN	0.28	1.73	5.63	13.16	OOM	OOM
SheafHyperGCN	**0.08**	**0.17**	**0.25**	**0.35**	**0.47**	**0.61**
DSHN	**0.13**	**0.59**	1.80	4.14	7.98	13.68
DSHNLight	**0.10**	**0.26**	**0.70**	1.56	2.97	5.06
Ours						
OmniDeepSheaf	0.40	0.67	**0.92**	**1.17**	**1.42**	**1.69**
OmniSheafTransformer	0.41	0.72	1.03	**1.38**	**1.74**	**2.18**
OmniPhaseSheaf	0.72	1.30	1.87	2.46	3.00	3.59
OmniPhaseSheaf-T	0.91	1.71	2.53	3.36	4.17	5.07
OmniCopreSheaf	0.78	1.08	1.36	1.66	1.95	2.26
OmniCopreSheaf-T	0.78	1.13	1.50	1.93	2.39	2.93

Table 27: Training peak memory (GB) on TELEGRAM (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.07 (no stalk)			
AllSetTransformer			0.13 (no stalk)			
SheafHyperGNN	–	**0.48**	1.43	3.23	6.18	10.54
SheafHyperGCN	–	**0.17**	**0.25**	**0.35**	**0.47**	**0.61**
DSHN	–	0.59	1.80	4.14	7.98	13.69
DSHNLight	–	**0.26**	**0.71**	1.56	2.97	5.06
Ours						
OmniDeepSheaf	–	0.67	**0.92**	**1.18**	**1.42**	**1.70**
OmniSheafTransformer	–	0.72	1.03	**1.39**	**1.75**	**2.19**
OmniPhaseSheaf	–	1.30	1.88	2.46	3.01	3.60
OmniPhaseSheaf-T	–	1.71	2.53	3.36	4.18	5.08
OmniCopreSheaf	–	1.09	1.37	1.68	1.98	2.30
OmniCopreSheaf-T	–	1.13	1.51	1.95	2.42	2.97

Table 28: Training peak memory (GB) on ROMAN-EMPIRE (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			1.28 (no stalk)			
AllSetTransformer			1.65 (no stalk)			
SheafHyperGNN	**2.81**	9.28	19.78	OOM	OOM	OOM
SheafHyperGCN	4.44	16.54	OOM	OOM	OOM	OOM
DSHN	5.64	18.82	OOM	OOM	OOM	OOM
DSHNLight	**1.76**	**2.99**	**4.23**	**5.48**	**6.72**	**7.95**
Ours						
OmniDeepSheaf	**3.07**	**5.20**	**7.36**	**9.49**	**11.62**	**13.76**
OmniSheafTransformer	3.14	**5.34**	**7.60**	**9.85**	**12.13**	**14.45**
OmniPhaseSheaf	5.97	10.96	15.93	20.92	OOM	OOM
OmniPhaseSheaf-T	8.80	16.62	OOM	OOM	OOM	OOM
OmniCopreSheaf	6.52	9.46	12.42	15.36	18.31	21.27
OmniCopreSheaf-T	6.68	9.79	12.97	16.17	19.42	OOM

Table 29: Training peak memory (GB) on ROMAN-EMPIRE (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			1.28 (no stalk)			
AllSetTransformer			1.65 (no stalk)			
SheafHyperGNN	**2.79**	9.49	20.44	OOM	OOM	OOM
SheafHyperGCN	4.44	16.57	OOM	OOM	OOM	OOM
DSHN	5.64	18.90	OOM	OOM	OOM	OOM
DSHNLight	**1.76**	**3.02**	**4.30**	**5.62**	**6.95**	**8.30**
Ours						
OmniDeepSheaf	**3.07**	**5.20**	**7.36**	**9.48**	**11.61**	**13.75**
OmniSheafTransformer	3.14	**5.34**	**7.60**	**9.85**	**12.13**	**14.44**
OmniPhaseSheaf	5.97	10.96	15.94	20.92	OOM	OOM
OmniPhaseSheaf-T	8.80	16.62	OOM	OOM	OOM	OOM
OmniCopreSheaf	6.52	9.45	12.41	15.35	18.30	21.25
OmniCopreSheaf-T	6.68	9.79	12.96	16.16	19.41	OOM

Table 30: Training peak memory (GB) on ROMAN-EMPIRE (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			1.28 (no stalk)			
AllSetTransformer			1.65 (no stalk)			
SheafHyperGNN	–	9.20	19.78	OOM	OOM	OOM
SheafHyperGCN	–	16.55	OOM	OOM	OOM	OOM
DSHN	–	18.90	OOM	OOM	OOM	OOM
DSHNLight	–	**3.02**	**4.31**	**5.63**	**6.97**	**8.32**
Ours						
OmniDeepSheaf	–	**5.21**	**7.38**	**9.52**	**11.67**	**13.82**
OmniSheafTransformer	–	**5.35**	**7.62**	**9.88**	**12.18**	**14.52**
OmniPhaseSheaf	–	10.97	15.96	20.96	OOM	OOM
OmniPhaseSheaf-T	–	16.63	OOM	OOM	OOM	OOM
OmniCopreSheaf	–	9.49	12.49	15.48	18.51	21.54
OmniCopreSheaf-T	–	9.82	13.04	16.29	19.62	OOM

INFERENCE TIME PER STEP

Table 31: Inference time per step (ms) on CORA (Diag restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			2.71 ± 0.02 (no stalk)			
AllSetTransformer			4.872 ± 0.04 (no stalk)			
Ours						
AllDeepSets			2.71 ± 0.02 (no stalk)			
AllSetTransformer			4.872 ± 0.04 (no stalk)			
SheafHyperGNN	**13.96 ± 0.17**	**15.05 ± 0.08**	**17.15 ± 0.04**	**19.24 ± 0.06**	**21.61 ± 0.09**	**24.51 ± 0.02**
SheafHyperGCN	67.23 ± 0.11	73.71 ± 0.64	75.54 ± 0.70	78.24 ± 0.13	81.18 ± 0.40	86.61 ± 0.13
OmniDeepSheaf	**4.49 ± 0.03**	**8.477 ± 0.34**	**10.68 ± 0.05**	**13.78 ± 0.07**	**17.27 ± 0.07**	**20.89 ± 0.04**
OmniSheafTransformer	**5.277 ± 0.04**	**12.95 ± 0.18**	**21.95 ± 0.22**	**33.82 ± 0.18**	**50.15 ± 4.1**	**68.61 ± 3.1**

Table 32: Inference time per step (ms) on CORA (General restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			2.71 ± 0.02 (no stalk)			
AllSetTransformer			4.872 ± 0.04 (no stalk)			
Ours						
AllDeepSets			2.71 ± 0.02 (no stalk)			
AllSetTransformer			4.872 ± 0.04 (no stalk)			
SheafHyperGNN	**14.66 ± 0.28**	**19.9 ± 0.08**	**29.46 ± 0.13**	**41.47 ± 0.85**	OOM	OOM
SheafHyperGCN	68.18 ± 0.29	76.96 ± 0.40	84.31 ± 0.44	92.36 ± 0.61	**102.8 ± 0.40**	**115 ± 0.25**
OmniDeepSheaf	**4.375 ± 0.02**	**8.442 ± 0.29**	**10.83 ± 0.11**	**13.94 ± 0.06**	**17.3 ± 0.17**	**21.03 ± 0.14**
OmniSheafTransformer	**5.164 ± 0.03**	**12.91 ± 0.21**	**22.03 ± 0.06**	**34.59 ± 0.34**	**51.28 ± 4.1**	**69.39 ± 3.2**

Table 33: Inference time per step (ms) on CORA (Ortho restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			2.71 ± 0.02 (no stalk)			
AllSetTransformer			4.872 ± 0.04 (no stalk)			
Ours						
AllDeepSets			2.71 ± 0.02 (no stalk)			
AllSetTransformer			4.872 ± 0.04 (no stalk)			
SheafHyperGNN	–	**15.7 ± 0.04**	**20.25 ± 0.06**	**28.71 ± 0.07**	**48.65 ± 0.48**	**78.01 ± 0.90**
SheafHyperGCN	–	73.72 ± 0.13	76.15 ± 0.27	79.17 ± 0.22	82.56 ± 0.29	88.45 ± 0.06
OmniDeepSheaf	–	**9.267 ± 0.20**	**11.76 ± 0.11**	**15.0 ± 0.03**	**18.63 ± 0.13**	**22.48 ± 0.01**
OmniSheafTransformer	–	**13.73 ± 0.19**	**22.74 ± 0.13**	**35.03 ± 0.31**	**50.9 ± 3.9**	**69.38 ± 3.4**

Table 34: Inference time per step (ms) on TELEGRAM (Diag restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			2.492 ± 0.01 (no stalk)			
AllSetTransformer			4.634 ± 0.07 (no stalk)			
SheafHyperGNN	13.87 ± 0.28	15.01 ± 0.04	15.87 ± 0.01	16.53 ± 0.05	**17.14 ± 0.03**	**18.18 ± 0.05**
SheafHyperGCN	27.13 ± 0.04	47.38 ± 0.08	49.26 ± 0.09	51.02 ± 0.13	53.0 ± 0.19	53.91 ± 0.03
DSHN	8.31 ± 0.07	**8.554 ± 0.09**	**8.885 ± 0.04**	**9.199 ± 0.13**	**9.555 ± 0.18**	**10.06 ± 0.24**
DSHNLight	8.953 ± 0.01	**9.349 ± 0.03**	**9.741 ± 0.01**	**10.0 ± 0.03**	**10.38 ± 0.08**	**10.74 ± 0.08**
Ours						
OmniDeepSheaf	**4.648 ± 0.01**	**8.811 ± 0.22**	**11.29 ± 0.14**	**14.54 ± 0.31**	17.66 ± 0.08	20.81 ± 0.13
OmniSheafTransformer	**5.38 ± 0.03**	13.79 ± 0.06	23.08 ± 0.15	35.42 ± 0.15	51.13 ± 1.9	70.65 ± 2.0
OmniPhaseSheaf	**7.319 ± 0.14**	20.64 ± 0.08	32.94 ± 0.08	43.41 ± 0.08	53.57 ± 0.39	64.25 ± 0.05
OmniPhaseSheaf-T	13.9 ± 0.63	35.27 ± 0.14	62.52 ± 2.1	94.4 ± 0.78	132.4 ± 4.0	183 ± 0.77
OmniCopreSheaf	11.51 ± 0.05	13.43 ± 0.10	17.55 ± 0.08	21.98 ± 0.06	26.87 ± 0.21	31.94 ± 0.03
OmniCopreSheaf-T	12.72 ± 0.19	18.54 ± 0.14	29.97 ± 0.19	43.94 ± 0.93	60.97 ± 6.1	82.57 ± 4.3

Table 35: Inference time per step (ms) on TELEGRAM (General restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			2.492 ± 0.01 (no stalk)			
AllSetTransformer			4.634 ± 0.07 (no stalk)			
SheafHyperGNN	14.14 ± 0.37	17.51 ± 0.02	25.4 ± 0.03	41.09 ± 2.0	OOM	OOM
SheafHyperGCN	27.28 ± 0.09	47.99 ± 0.07	50.04 ± 0.09	51.77 ± 0.07	53.42 ± 0.13	**55.03 ± 0.02**
DSHN	50.53 ± 0.31	52.29 ± 0.17	56.9 ± 0.03	68.84 ± 0.16	99.78 ± 0.51	154.2 ± 0.50
DSHNLight	52.0 ± 0.40	53.37 ± 0.28	57.91 ± 0.51	69.34 ± 0.27	98.83 ± 0.76	151.5 ± 0.11
Ours						
OmniDeepSheaf	**4.422 ± 0.07**	**8.909 ± 0.28**	**11.35 ± 0.19**	**14.37 ± 0.10**	**17.53 ± 0.10**	**21.01 ± 0.06**
OmniSheafTransformer	**5.283 ± 0.02**	**13.81 ± 0.17**	**23.3 ± 0.13**	**35.75 ± 0.06**	**51.48 ± 2.1**	71.27 ± 2.9
OmniPhaseSheaf	**7.23 ± 0.10**	20.59 ± 0.08	32.96 ± 0.08	43.61 ± 0.11	54.06 ± 0.45	64.68 ± 0.42
OmniPhaseSheaf-T	12.91 ± 0.03	35.79 ± 0.28	64.75 ± 1.3	99.21 ± 1.8	118.7 ± 1.5	156.1 ± 0.45
OmniCopreSheaf	11.18 ± 0.05	**13.45 ± 0.03**	**17.52 ± 0.03**	**21.96 ± 0.08**	**26.98 ± 0.12**	**32.27 ± 0.01**
OmniCopreSheaf-T	12.39 ± 0.12	18.78 ± 0.05	29.99 ± 0.06	45.23 ± 1.6	66.96 ± 4.3	87.25 ± 5.7

Table 36: Inference time per step (ms) on TELEGRAM (Ortho restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			2.492 ± 0.01 (no stalk)			
AllSetTransformer			4.634 ± 0.07 (no stalk)			
SheafHyperGNN	–	16.26 ± 0.10	23.34 ± 0.21	39.04 ± 0.35	84.83 ± 1.1	187.2 ± 0.56
SheafHyperGCN	–	47.59 ± 0.19	49.55 ± 0.04	51.55 ± 0.10	53.22 ± 0.08	**54.91 ± 0.06**
DSHN	–	**10.19 ± 0.03**	**14.21 ± 0.03**	**25.44 ± 0.13**	54.58 ± 1.2	107.1 ± 0.47
DSHNLight	–	**11.04 ± 0.05**	**14.92 ± 0.02**	**25.79 ± 0.05**	54.11 ± 0.82	105.7 ± 0.29
Ours						
OmniDeepSheaf	–	**9.773 ± 0.35**	**12.26 ± 0.11**	**15.58 ± 0.03**	**18.95 ± 0.11**	**22.38 ± 0.05**
OmniSheafTransformer	–	14.46 ± 0.04	23.79 ± 0.57	36.54 ± 0.09	**52.27 ± 1.8**	71.76 ± 2.5
OmniPhaseSheaf	–	21.93 ± 0.37	34.45 ± 0.59	45.61 ± 0.30	55.73 ± 0.50	67.3 ± 1.5
OmniPhaseSheaf-T	–	37.0 ± 0.50	66.14 ± 1.7	98.92 ± 0.90	135.8 ± 1.9	181.5 ± 0.64
OmniCopreSheaf	–	17.87 ± 0.09	22.38 ± 0.03	27.64 ± 0.10	**33.1 ± 0.08**	**38.85 ± 0.02**
OmniCopreSheaf-T	–	22.75 ± 0.11	33.9 ± 0.16	48.68 ± 1.3	67.88 ± 2.2	89.23 ± 2.2

Table 37: Inference time per step (ms) on ROMAN-EMPIRE (Diag restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			25.8 ± 0.06 (no stalk)			
AllSetTransformer			32.29 ± 0.03 (no stalk)			
SheafHyperGNN	**25.04 ± 0.03**	**39.46 ± 0.40**	**53.83 ± 4.0**	OOM	OOM	OOM
SheafHyperGCN	873.7 ± 46	1001 ± 31	OOM	OOM	OOM	OOM
DSHN	**53.21 ± 0.05**	**98.95 ± 0.09**	OOM	OOM	OOM	OOM
DSHNLight	54.2 ± 0.15	**99.91 ± 0.03**	**141.7 ± 0.20**	**183.8 ± 0.08**	**225.4 ± 0.14**	**267.8 ± 0.18**
Ours						
OmniDeepSheaf	**45.44 ± 0.58**	101.9 ± 1.4	**132.2 ± 0.19**	**161.6 ± 1.4**	**192.2 ± 1.1**	**227.4 ± 5.9**
OmniSheafTransformer	56.18 ± 3.6	147.6 ± 1.4	239.9 ± 0.52	338.4 ± 1.0	516.4 ± 3.8	625.5 ± 1.6
OmniPhaseSheaf	117.5 ± 0.19	277.5 ± 0.42	378.4 ± 1.4	473.3 ± 0.65	OOM	OOM
OmniPhaseSheaf-T	209.8 ± 0.08	525.8 ± 0.54	OOM	OOM	OOM	OOM
OmniCopreSheaf	113.9 ± 0.75	168.1 ± 0.57	216.8 ± 0.34	**262.4 ± 0.41**	**312.9 ± 1.2**	**369 ± 1.7**
OmniCopreSheaf-T	124.6 ± 0.78	215.6 ± 1.1	332.5 ± 2.6	466.7 ± 1.2	630.5 ± 0.65	OOM

Table 38: Inference time per step (ms) on ROMAN-EMPIRE (General restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			25.8 ± 0.06 (no stalk)			
AllSetTransformer			32.29 ± 0.03 (no stalk)			
SheafHyperGNN	**29.05 ± 0.09**	**69.54 ± 0.53**	**126.5 ± 0.59**	OOM	OOM	OOM
SheafHyperGCN	970.6 ± 42	1244 ± 44	OOM	OOM	OOM	OOM
DSHN	3096 ± 15	3355 ± 91	OOM	OOM	OOM	OOM
DSHNLight	3100 ± 19	3407 ± 50	3378 ± 118	3457 ± 52	3591 ± 121	3668 ± 131
Ours						
OmniDeepSheaf	**45.44 ± 0.49**	**101.5 ± 0.34**	**132.3 ± 0.92**	**161.9 ± 1.3**	**192.4 ± 1.5**	**225.7 ± 5.7**
OmniSheafTransformer	**55.6 ± 4.4**	**148 ± 1.0**	240 ± 0.31	**344.9 ± 3.4**	**498.1 ± 2.0**	**628.6 ± 3.1**
OmniPhaseSheaf	117.9 ± 0.50	279.6 ± 0.92	379.7 ± 0.44	474.4 ± 0.34	OOM	OOM
OmniPhaseSheaf-T	208.6 ± 0.12	521.8 ± 0.87	OOM	OOM	OOM	OOM
OmniCopreSheaf	113.6 ± 0.56	168.7 ± 0.24	**217.8 ± 0.71**	**261.2 ± 0.33**	**314.7 ± 0.92**	**366.9 ± 2.0**
OmniCopreSheaf-T	124.4 ± 0.66	219.9 ± 0.57	336.7 ± 2.7	470.7 ± 4.9	656.2 ± 2.9	OOM

Table 39: Inference time per step (ms) on ROMAN-EMPIRE (Ortho restriction maps, width 256, 3 layers). Median ± half-range across 3 timing trials. Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			25.8 ± 0.06 (no stalk)			
AllSetTransformer			32.29 ± 0.03 (no stalk)			
SheafHyperGNN	–	**39.83 ± 0.02**	**62.4 ± 3.2**	OOM	OOM	OOM
SheafHyperGCN	–	996 ± 46	OOM	OOM	OOM	OOM
DSHN	–	**102.4 ± 0.16**	OOM	OOM	OOM	OOM
DSHNLight	–	**103.6 ± 0.12**	**153.8 ± 0.26**	**217.3 ± 0.58**	**306.6 ± 0.25**	**406.5 ± 0.39**
Ours						
OmniDeepSheaf	–	**102.4 ± 1.1**	**134.8 ± 1.0**	**165.3 ± 0.60**	**197.9 ± 0.76**	**231.8 ± 1.5**
OmniSheafTransformer	–	149.3 ± 0.79	240.2 ± 1.00	351.8 ± 1.1	522.7 ± 1.1	629.7 ± 1.0
OmniPhaseSheaf	–	281.6 ± 1.1	380.1 ± 0.46	476.4 ± 0.51	OOM	OOM
OmniPhaseSheaf-T	–	527.5 ± 0.31	OOM	OOM	OOM	OOM
OmniCopreSheaf	–	172.3 ± 0.05	226.9 ± 0.38	**275.8 ± 0.23**	**335.1 ± 0.79**	**394.4 ± 1.3**
OmniCopreSheaf-T	–	222.1 ± 0.73	340 ± 2.6	477.6 ± 3.4	658.9 ± 5.1	OOM

PEAK INFERENCE MEMORY

Table 40: Inference peak memory (GB) on CORA (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.08 (no stalk)			
AllSetTransformer			0.08 (no stalk)			
Ours						
AllDeepSets			0.08 (no stalk)			
AllSetTransformer			0.08 (no stalk)			
SheafHyperGNN	**0.10**	**0.11**	**0.12**	**0.14**	**0.17**	**0.20**
SheafHyperGCN	**0.10**	0.17	0.32	0.53	0.79	1.11
OmniDeepSheaf	**0.10**	**0.14**	**0.19**	**0.24**	**0.29**	**0.34**
OmniSheafTransformer	**0.10**	0.17	0.25	0.35	0.45	0.55

Table 41: Inference peak memory (GB) on CORA (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.08 (no stalk)			
AllSetTransformer			0.08 (no stalk)			
Ours						
AllDeepSets			0.08 (no stalk)			
AllSetTransformer			0.08 (no stalk)			
SheafHyperGNN	**0.11**	**0.12**	**0.18**	**0.30**	OOM	OOM
SheafHyperGCN	**0.11**	0.28	0.58	0.99	1.52	2.16
OmniDeepSheaf	**0.10**	**0.14**	**0.19**	**0.24**	**0.29**	**0.34**
OmniSheafTransformer	**0.10**	0.17	0.25	0.35	**0.45**	**0.55**

Table 42: Inference peak memory (GB) on CORA (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			0.08 (no stalk)			
AllSetTransformer			0.08 (no stalk)			
Ours						
AllDeepSets			0.08 (no stalk)			
AllSetTransformer			0.08 (no stalk)			
SheafHyperGNN	–	**0.11**	**0.15**	**0.26**	**0.43**	**0.67**
SheafHyperGCN	–	**0.17**	0.32	0.53	0.80	1.12
OmniDeepSheaf	–	**0.14**	**0.19**	**0.24**	**0.29**	**0.34**
OmniSheafTransformer	–	**0.17**	**0.25**	**0.35**	**0.45**	**0.55**

Table 43: Inference peak memory (GB) on TELEGRAM (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.06 (no stalk)			
AllSetTransformer			0.06 (no stalk)			
SheafHyperGNN	**0.07**	**0.08**	**0.08**	**0.10**	**0.12**	**0.14**
SheafHyperGCN	**0.07**	0.11	0.18	0.28	0.38	0.51
DSHN	**0.09**	**0.10**	**0.10**	**0.12**	**0.15**	**0.18**
DSHNLight	**0.09**	**0.10**	**0.10**	0.13	**0.15**	**0.18**
Ours						
OmniDeepSheaf	**0.09**	0.13	0.19	0.25	0.30	0.36
OmniSheafTransformer	**0.09**	0.17	0.25	0.35	0.46	0.58
OmniPhaseSheaf	0.15	0.29	0.42	0.55	0.68	0.81
OmniPhaseSheaf-T	0.16	0.31	0.46	0.63	0.80	0.99
OmniCopreSheaf	0.11	0.20	0.28	0.37	0.45	0.54
OmniCopreSheaf-T	0.12	0.24	0.36	0.51	0.66	0.85

Table 44: Inference peak memory (GB) on TELEGRAM (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.06 (no stalk)			
AllSetTransformer			0.06 (no stalk)			
SheafHyperGNN	**0.09**	0.16	0.49	1.14	OOM	OOM
SheafHyperGCN	**0.07**	**0.11**	**0.18**	**0.28**	**0.38**	**0.52**
DSHN	**0.09**	0.21	0.65	1.51	2.91	4.99
DSHNLight	**0.09**	0.21	0.66	1.51	2.92	5.00
Ours						
OmniDeepSheaf	**0.09**	**0.13**	**0.19**	**0.25**	**0.30**	**0.36**
OmniSheafTransformer	**0.09**	0.17	0.25	0.35	0.46	0.58
OmniPhaseSheaf	0.15	0.29	0.42	0.55	0.68	0.81
OmniPhaseSheaf-T	0.16	0.31	0.47	0.63	0.80	0.99
OmniCopreSheaf	0.11	0.20	0.28	0.37	0.45	0.54
OmniCopreSheaf-T	0.12	0.24	0.36	0.51	0.66	0.85

Table 45: Inference peak memory (GB) on TELEGRAM (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.06 (no stalk)			
AllSetTransformer			0.06 (no stalk)			
SheafHyperGNN	–	**0.16**	0.49	1.14	2.18	3.74
SheafHyperGCN	–	**0.11**	**0.18**	**0.28**	**0.38**	**0.52**
DSHN	–	0.21	0.65	1.51	2.91	4.99
DSHNLight	–	0.21	0.66	1.51	2.92	5.00
Ours						
OmniDeepSheaf	–	**0.13**	**0.19**	**0.25**	**0.30**	**0.36**
OmniSheafTransformer	–	0.17	**0.25**	**0.35**	0.46	0.58
OmniPhaseSheaf	–	0.29	0.42	0.55	0.68	0.81
OmniPhaseSheaf-T	–	0.31	0.47	0.63	0.80	0.99
OmniCopreSheaf	–	0.20	0.28	0.37	**0.45**	**0.54**
OmniCopreSheaf-T	–	0.24	0.36	0.51	0.66	0.85

Table 46: Inference peak memory (GB) on ROMAN-EMPIRE (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.41 (no stalk)			
AllSetTransformer			0.44 (no stalk)			
SheafHyperGNN	**0.58**	**0.64**	**0.69**	OOM	OOM	OOM
SheafHyperGCN	2.13	8.22	OOM	OOM	OOM	OOM
DSHN	0.90	**1.05**	OOM	OOM	OOM	OOM
DSHNLight	0.90	**1.05**	**1.51**	**1.97**	**2.43**	**2.89**
Ours						
OmniDeepSheaf	**0.62**	**0.94**	**1.39**	**1.84**	**2.29**	**2.73**
OmniSheafTransformer	**0.62**	1.12	1.67	**2.23**	**2.79**	**3.37**
OmniPhaseSheaf	1.05	2.05	3.04	4.04	OOM	OOM
OmniPhaseSheaf-T	1.40	2.76	OOM	OOM	OOM	OOM
OmniCopreSheaf	**0.86**	1.50	2.24	2.96	3.69	4.42
OmniCopreSheaf-T	0.87	1.68	2.53	3.38	4.25	OOM

Table 47: Inference peak memory (GB) on ROMAN-EMPIRE (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.41 (no stalk)			
AllSetTransformer			0.44 (no stalk)			
SheafHyperGNN	**0.71**	**0.76**	**0.82**	OOM	OOM	OOM
SheafHyperGCN	4.14	16.23	OOM	OOM	OOM	OOM
DSHN	0.90	**1.06**	OOM	OOM	OOM	OOM
DSHNLight	0.90	**1.06**	**1.54**	**2.04**	**2.54**	**3.06**
Ours						
OmniDeepSheaf	**0.62**	**0.94**	**1.39**	**1.84**	**2.29**	**2.73**
OmniSheafTransformer	**0.62**	1.11	1.67	**2.22**	**2.79**	**3.36**
OmniPhaseSheaf	1.05	2.05	3.04	4.04	OOM	OOM
OmniPhaseSheaf-T	1.40	2.76	OOM	OOM	OOM	OOM
OmniCopreSheaf	**0.86**	1.50	2.24	2.96	3.69	4.42
OmniCopreSheaf-T	0.87	1.68	2.53	3.38	4.25	OOM

Table 48: Inference peak memory (GB) on ROMAN-EMPIRE (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			0.41 (no stalk)			
AllSetTransformer			0.44 (no stalk)			
SheafHyperGNN	–	**0.64**	**0.69**	OOM	OOM	OOM
SheafHyperGCN	–	8.22	OOM	OOM	OOM	OOM
DSHN	–	**1.06**	OOM	OOM	OOM	OOM
DSHNLight	–	**1.06**	**1.54**	**2.04**	**2.54**	**3.06**
Ours						
OmniDeepSheaf	–	**0.94**	**1.39**	**1.84**	**2.28**	**2.73**
OmniSheafTransformer	–	1.11	1.67	**2.22**	**2.79**	**3.36**
OmniPhaseSheaf	–	2.05	3.04	4.04	OOM	OOM
OmniPhaseSheaf-T	–	2.76	OOM	OOM	OOM	OOM
OmniCopreSheaf	–	1.50	2.24	2.96	3.69	4.42
OmniCopreSheaf-T	–	1.68	2.53	3.38	4.25	OOM

FORWARD GFLOP

Table 49: Forward GFLOP on CORA (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			6.994 (no stalk)			
AllSetTransformer			10.31 (no stalk)			
Ours						
AllDeepSets			6.994 (no stalk)			
AllSetTransformer			10.31 (no stalk)			
SheafHyperGNN	**4.217**	**8.079**	**11.94**	**15.8**	**19.66**	**23.53**
SheafHyperGCN	**3.865**	**8.056**	**12.93**	**18.49**	**24.72**	**31.64**
OmniDeepSheaf	**10.26**	**19.55**	**28.85**	**38.2**	**47.59**	**57.03**
OmniSheafTransformer	15.76	55.91	121.4	212.4	328.7	470.5

Table 50: Forward GFLOP on CORA (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			6.994 (no stalk)			
AllSetTransformer			10.31 (no stalk)			
Ours						
AllDeepSets			6.994 (no stalk)			
AllSetTransformer			10.31 (no stalk)			
SheafHyperGNN	**4.217**	**8.093**	**11.98**	**15.89**	OOM	OOM
SheafHyperGCN	**3.865**	**8.071**	**12.97**	**18.57**	**24.87**	**31.86**
OmniDeepSheaf	**10.26**	**19.59**	**28.98**	**38.46**	**48.02**	**57.67**
OmniSheafTransformer	15.76	55.95	121.6	212.6	**329.1**	**471.1**

Table 51: Forward GFLOP on CORA (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			6.994 (no stalk)			
AllSetTransformer			10.31 (no stalk)			
Ours						
AllDeepSets			6.994 (no stalk)			
AllSetTransformer			10.31 (no stalk)			
SheafHyperGNN	–	**8.072**	**11.94**	**15.82**	**19.7**	**23.6**
SheafHyperGCN	–	**8.049**	**12.93**	**18.5**	**24.76**	**31.71**
OmniDeepSheaf	–	**19.53**	**28.86**	**38.25**	**47.71**	**57.24**
OmniSheafTransformer	–	55.89	121.4	212.4	328.8	470.7

Table 52: Forward GFLOP on TELEGRAM (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.3057 (no stalk)			
AllSetTransformer			0.442 (no stalk)			
SheafHyperGNN	**0.1064**	**0.1806**	**0.2549**	**0.3292**	**0.4034**	**0.4777**
SheafHyperGCN	**0.05541**	**0.09281**	**0.1443**	**0.21**	**0.2897**	**0.3836**
DSHN	**0.2127**	**0.3613**	**0.5098**	**0.6583**	**0.8068**	**0.9553**
DSHNLight	**0.2127**	**0.3613**	**0.5098**	**0.6583**	**0.8068**	**0.9553**
Ours						
OmniDeepSheaf	7.446	14.91	22.4	29.94	37.54	45.2
OmniSheafTransformer	14.6	58.16	130.7	232.2	362.8	522.3
OmniPhaseSheaf	14.77	29.55	44.36	59.23	74.15	89.13
OmniPhaseSheaf-T	29.07	116.1	261	463.8	724.6	1043
OmniCopreSheaf	8.035	15.91	23.84	31.88	40.03	48.29
OmniCopreSheaf-T	15.19	59.3	132.6	235	366.7	527.6

Table 53: Forward GFLOP on TELEGRAM (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
AllDeepSets			0.3057 (no stalk)			
AllSetTransformer			0.442 (no stalk)			
SheafHyperGNN	**0.1064**	**0.1996**	**0.3118**	**0.4433**	OOM	OOM
SheafHyperGCN	**0.05542**	**0.1116**	**0.2007**	**0.323**	**0.4784**	**0.6672**
DSHN	**0.2128**	**0.3987**	**0.622**	**0.883**	**1.182**	**1.518**
DSHNLight	**0.2128**	**0.3987**	**0.622**	**0.883**	**1.182**	**1.518**
Ours						
OmniDeepSheaf	7.446	14.96	22.56	30.28	**38.1**	**46.04**
OmniSheafTransformer	14.6	58.22	130.9	232.6	363.3	523.2
OmniPhaseSheaf	14.77	29.61	44.53	59.56	74.71	89.97
OmniPhaseSheaf-T	29.07	116.1	261.1	464.1	725.2	1044
OmniCopreSheaf	8.035	16.13	24.51	33.22	42.26	51.64
OmniCopreSheaf-T	15.19	59.53	133.2	236.4	368.9	530.9

Table 54: Forward GFLOP on TELEGRAM (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			0.3057 (no stalk)			
AllSetTransformer			0.442 (no stalk)			
SheafHyperGNN	–	**0.1716**	**0.2559**	**0.3501**	**0.4545**	**0.5693**
SheafHyperGCN	–	**0.0838**	**0.1453**	**0.231**	**0.3409**	**0.4754**
DSHN	–	**0.3429**	**0.5108**	**0.6979**	**0.9045**	**1.131**
DSHNLight	–	**0.3429**	**0.5108**	**0.6979**	**0.9045**	**1.131**
Ours						
OmniDeepSheaf	–	14.88	22.4	30.0	37.7	45.47
OmniSheafTransformer	–	58.13	130.7	232.3	362.9	522.6
OmniPhaseSheaf	–	29.52	44.36	59.29	74.3	89.4
OmniPhaseSheaf-T	–	116	261	463.9	724.8	1044
OmniCopreSheaf	–	15.8	23.85	32.13	40.64	49.39
OmniCopreSheaf-T	–	59.2	132.6	235.3	367.3	528.7

Table 55: Forward GFLOP on ROMAN-EMPIRE (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			39.89 (no stalk)			
AllSetTransformer			58.22 (no stalk)			
SheafHyperGNN	**16.14**	**29.32**	**42.49**	OOM	OOM	OOM
SheafHyperGCN	**45.19**	157.3	OOM	OOM	OOM	OOM
DSHN	**32.29**	**58.64**	OOM	OOM	OOM	OOM
DSHNLight	**32.29**	**58.64**	**84.99**	**111.3**	**137.7**	**164**
Ours						
OmniDeepSheaf	79.22	**150.3**	**221.5**	**293.2**	**365.2**	**437.6**
OmniSheafTransformer	127.8	477.6	1058	1868	2909	4180
OmniPhaseSheaf	149.3	290.5	431.9	573.6	OOM	OOM
OmniPhaseSheaf-T	246.5	945.1	OOM	OOM	OOM	OOM
OmniCopreSheaf	133.3	232.4	331.9	**432.2**	**533.2**	**635**
OmniCopreSheaf-T	181.9	577.5	1221	2114	3255	OOM

Table 56: Forward GFLOP on ROMAN-EMPIRE (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			39.89 (no stalk)			
AllSetTransformer			58.22 (no stalk)			
SheafHyperGNN	**16.15**	**29.45**	**42.88**	OOM	OOM	OOM
SheafHyperGCN	**45.19**	157.5	OOM	OOM	OOM	OOM
DSHN	**32.29**	**58.89**	OOM	OOM	OOM	OOM
DSHNLight	**32.29**	**58.89**	**85.75**	**112.9**	**140.2**	**167.9**
Ours						
OmniDeepSheaf	79.22	**150.7**	**222.7**	**295.5**	**369**	**443.3**
OmniSheafTransformer	127.8	477.9	1059	1870	2913	4186
OmniPhaseSheaf	149.3	290.9	433	575.9	OOM	OOM
OmniPhaseSheaf-T	246.5	945.5	OOM	OOM	OOM	OOM
OmniCopreSheaf	133.3	233.9	336.4	**441.3**	**548.4**	**657.8**
OmniCopreSheaf-T	181.9	579	1226	2123	3270	OOM

Table 57: Forward GFLOP on ROMAN-EMPIRE (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			39.89 (no stalk)			
AllSetTransformer			58.22 (no stalk)			
SheafHyperGNN	–	**29.26**	**42.5**	OOM	OOM	OOM
SheafHyperGCN	–	157.3	OOM	OOM	OOM	OOM
DSHN	–	**58.51**	OOM	OOM	OOM	OOM
DSHNLight	–	**58.51**	**85.0**	**111.6**	**138.4**	**165.2**
Ours						
OmniDeepSheaf	–	**150.1**	**221.6**	**293.6**	**366.2**	**439.5**
OmniSheafTransformer	–	477.4	1058	1868	2910	4182
OmniPhaseSheaf	–	290.3	431.9	574.1	OOM	OOM
OmniPhaseSheaf-T	–	944.9	OOM	OOM	OOM	OOM
OmniCopreSheaf	–	231.6	332	**433.9**	**537.4**	**642.5**
OmniCopreSheaf-T	–	576.7	1221	2116	3259	OOM

TRAINING GFLOP

Table 58: Training GFLOP on CORA (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			19.0 (no stalk)			
AllSetTransformer			24.96 (no stalk)			
Ours						
AllDeepSets			19.0 (no stalk)			
AllSetTransformer			24.96 (no stalk)			
SheafHyperGNN	**23.39**	**75.74**	**158.1**	**270.6**	**413**	**585.5**
SheafHyperGCN	**6.235**	**13.47**	**22.06**	**32.01**	**43.32**	**55.98**
OmniDeepSheaf	**28.8**	**54.67**	**80.6**	**106.7**	**132.9**	**159.2**
OmniSheafTransformer	45.3	163.8	358.4	629.2	976.2	1399

Table 59: Training GFLOP on CORA (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			19.0 (no stalk)			
AllSetTransformer			24.96 (no stalk)			
Ours						
AllDeepSets			19.0 (no stalk)			
AllSetTransformer			24.96 (no stalk)			
SheafHyperGNN	**23.39**	**75.79**	**158.3**	**270.8**	OOM	OOM
SheafHyperGCN	**6.235**	**13.49**	**22.11**	**32.1**	**43.47**	**56.2**
OmniDeepSheaf	**28.8**	**54.8**	**80.99**	**107.4**	**134.1**	**161.1**
OmniSheafTransformer	45.3	163.9	358.7	629.9	**977.5**	**1401**

Table 60: Training GFLOP on CORA (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			19.0 (no stalk)			
AllSetTransformer			24.96 (no stalk)			
Ours						
AllDeepSets			19.0 (no stalk)			
AllSetTransformer			24.96 (no stalk)			
SheafHyperGNN	–	**75.72**	**158.1**	**270.6**	**413.2**	**585.8**
SheafHyperGCN	–	**13.46**	**22.07**	**32.03**	**43.36**	**56.05**
OmniDeepSheaf	–	**54.61**	**80.61**	**106.8**	**133.2**	**159.8**
OmniSheafTransformer	–	163.7	358.4	629.3	976.6	1400

Table 61: Training GFLOP on TELEGRAM (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			0.9171 (no stalk)			
AllSetTransformer			1.326 (no stalk)			
SheafHyperGNN	**0.4417**	**1.033**	**1.87**	**2.953**	**4.282**	**5.857**
SheafHyperGCN	**0.06202**	**0.1109**	**0.1786**	**0.2653**	**0.3708**	**0.4953**
DSHN	0.7605	1.574	2.634	3.939	5.491	7.288
DSHNLight	**0.5334**	**1.125**	**1.84**	**2.678**	**3.639**	**4.723**
Ours						
OmniDeepSheaf	22.34	44.72	67.19	89.82	112.6	135.6
OmniSheafTransformer	43.8	174.5	392.1	696.7	1088	1567
OmniPhaseSheaf	44.3	88.65	133.1	177.7	222.4	267.4
OmniPhaseSheaf-T	87.22	348.2	782.9	1391	2174	3130
OmniCopreSheaf	24.1	47.72	71.51	95.63	120.1	144.9
OmniCopreSheaf-T	45.56	177.9	397.7	705.1	1100	1583

Table 62: Training GFLOP on TELEGRAM (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			0.9171 (no stalk)			
AllSetTransformer			1.326 (no stalk)			
SheafHyperGNN	**0.4418**	**1.09**	**2.041**	**3.296**	OOM	OOM
SheafHyperGCN	**0.06204**	**0.1296**	**0.235**	**0.3783**	**0.5595**	**0.7789**
DSHN	0.7606	1.686	2.97	4.611	**6.611**	**8.968**
DSHNLight	**0.5334**	**1.163**	**1.953**	**2.903**	**4.014**	**5.286**
Ours						
OmniDeepSheaf	22.34	44.89	67.69	90.83	114.3	138.1
OmniSheafTransformer	43.8	174.6	392.6	697.7	1090	1569
OmniPhaseSheaf	44.3	88.82	133.6	178.7	224.1	269.9
OmniPhaseSheaf-T	87.22	348.3	783.4	1392	2176	3133
OmniCopreSheaf	24.1	48.39	73.52	99.65	126.8	154.9
OmniCopreSheaf-T	45.56	178.6	399.7	709.1	1107	1593

Table 63: Training GFLOP on TELEGRAM (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets			0.9171 (no stalk)			
AllSetTransformer			1.326 (no stalk)			
SheafHyperGNN	–	**1.006**	**1.874**	**3.018**	**4.44**	**6.14**
SheafHyperGCN	–	**0.1018**	**0.1796**	**0.2863**	**0.4221**	**0.5872**
DSHN	–	1.52	2.638	4.06	5.788	7.822
DSHNLight	–	**1.107**	**1.841**	**2.718**	**3.737**	**4.899**
Ours						
OmniDeepSheaf	–	44.64	67.2	90.02	113.1	136.4
OmniSheafTransformer	–	174.4	392.1	696.9	1089	1568
OmniPhaseSheaf	–	88.57	133.1	177.9	222.9	268.2
OmniPhaseSheaf-T	–	348.1	782.9	1392	2174	3131
OmniCopreSheaf	–	47.4	71.55	96.41	122	148.3
OmniCopreSheaf-T	–	177.6	397.7	705.9	1102	1586

Table 64: Training GFLOP on ROMAN-EMPIRE (Diag restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets				116.2 (no stalk)		
AllSetTransformer				164.2 (no stalk)		
SheafHyperGNN	1090	4274	9562	OOM	OOM	OOM
SheafHyperGCN	**84.02**	**304.8**	OOM	OOM	OOM	OOM
DSHN	1128	4341	OOM	OOM	OOM	OOM
DSHNLight	589.7	2225	4913	8652	**1.344e + 04**	**1.929e + 04**
Ours						
OmniDeepSheaf	**234.2**	**443.9**	**654.2**	**865.6**	**1078**	**1292**
OmniSheafTransformer	**380**	1426	3162	5590	8709	**1.252e + 04**
OmniPhaseSheaf	441	857.6	**1275**	**1693**	OOM	OOM
OmniPhaseSheaf-T	732.7	2821	OOM	OOM	OOM	OOM
OmniCopreSheaf	396.3	**690.2**	**985.2**	**1283**	**1582**	1884
OmniCopreSheaf-T	542.1	1725	3654	6328	9748	OOM

Table 65: Training GFLOP on ROMAN-EMPIRE (General restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets				116.2 (no stalk)		
AllSetTransformer				164.2 (no stalk)		
SheafHyperGNN	1090	4275	9563	OOM	OOM	OOM
SheafHyperGCN	**84.03**	**305**	OOM	OOM	OOM	OOM
DSHN	1128	4342	OOM	OOM	OOM	OOM
DSHNLight	589.7	2226	4913	8653	**1.345e + 04**	**1.929e + 04**
Ours						
OmniDeepSheaf	**234.2**	**445**	**657.6**	**872.4**	**1090**	**1309**
OmniSheafTransformer	**380**	1427	3166	5597	8721	**1.254e + 04**
OmniPhaseSheaf	441	858.8	**1278**	**1700**	OOM	OOM
OmniPhaseSheaf-T	732.7	2822	OOM	OOM	OOM	OOM
OmniCopreSheaf	396.3	**694.7**	**998.9**	**1310**	**1628**	1952
OmniCopreSheaf-T	542.1	1730	3667	6355	9793	OOM

Table 66: Training GFLOP on ROMAN-EMPIRE (Ortho restriction maps, width 256, 3 layers). Colors mark the first, second, and third lowest value per column. OOM = out-of-memory on a 24 GB NVIDIA L4; – = not applicable.

Method	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$	$d = 6$
AllDeepSets				116.2 (no stalk)		
AllSetTransformer				164.2 (no stalk)		
SheafHyperGNN	–	4274	9562	OOM	OOM	OOM
SheafHyperGCN	–	**304.8**	OOM	OOM	OOM	OOM
DSHN	–	4341	OOM	OOM	OOM	OOM
DSHNLight	–	2225	4913	8652	**1.344e + 04**	**1.929e + 04**
Ours						
OmniDeepSheaf	–	**443.3**	**654.3**	**866.9**	**1081**	**1298**
OmniSheafTransformer	–	1425	3162	5591	8712	**1.253e + 04**
OmniPhaseSheaf	–	857.1	**1275**	**1694**	OOM	OOM
OmniPhaseSheaf-T	–	2821	OOM	OOM	OOM	OOM
OmniCopreSheaf	–	**688**	**985.6**	**1288**	**1595**	1907
OmniCopreSheaf-T	–	1723	3654	6333	9761	OOM

G HYPERPARAMETER ABLATIONS

Because the stalk dimension d, convolutional depth L_{conv}, and hidden width w were sampled jointly during our random search (Table 6), we isolate the effect of each hyperparameter independently. Specifically, we fix the parameter of interest to a single value and evaluate the best validation run at that setting, allowing all other hyperparameters to adapt around it. Following our main results, restriction-map families are pooled together. Full ablation tables are provided for d (Tables 67 to 70), L_{conv} (Tables 71 to 74), and w (Tables 75 to 78).

- **Stalk dimension.** Increasing the stalk dimension consistently yields the largest performance boost. Across all 118 dataset and model combinations, higher dimensions are strongly favored: $d \geq 3$ delivers the best results in 78.0% of cases, whereas $d = 1$ is optimal only 6.8% of the time. Upgrading from $d = 1$ to the best-performing d raises test accuracy by an average of 5.66 percentage points.

  This benefit is especially pronounced on directed graphs when models lack an explicit directional mechanism, leaving stalk capacity as the sole engine for expressive power. For instance, on the synthetic benchmarks, OST climbs from 38.13% at $d = 1$ up to 86.67% at $d = 6$ ($+48.53$ points on $I_o = 50$), while ODS jumps from 30.40% at $d = 1$ to 72.80% at $d = 4$ ($+42.40$ points on $I_o = 10$). Models with directional inductive biases also gain significantly; OPS on the Roman-empire dataset rises from 76.26% at $d = 1$ to 88.00% at $d = 4$. Viewed alongside our computational cost analysis (Section 4.4), these trends show that OS can scale to larger stalk dimensions under tight memory budgets and translates that additional capacity directly into accuracy.
- **Depth.** Shallower architectures consistently outperform deeper alternatives. A compact backbone of $L_{\text{conv}} \leq 2$ is optimal in 61.9% of configurations, while deeper networks with $L_{\text{conv}} \geq 4$ win in just 15.3% of experiments. Because every convolution executes a full round trip between vertices and edges ($\mathcal{V} \to \mathcal{E} \to \mathcal{V}$), two layers already span a 4-hop neighborhood on the incidence graph. Pushing depth beyond this threshold adds optimization overhead with diminishing practical returns.
- **Width.** Hidden width w has the mildest influence on overall accuracy. Upgrading from the baseline of $w = 32$ to the top-performing width yields a modest average improvement of just 1.98 points, with modest sizes ($w \leq 64$) selected in 55.1% of trials. When allocating parameter budgets, compute is substantially better spent expanding stalk dimension rather than widening hidden channels.

Table 67: Ablation on the stalk dimension d (directed node classification). Test accuracy (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given d, all other hyperparameters re-tuned. Colors mark the first, second, and third d per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
Roman-empire	**OmniPhaseSheaf**	76.26 ± 2.95	**87.82 ± 0.25**	**87.43 ± 0.12**	**88.00 ± 0.06**	–	–
	OmniPhaseSheaf-T	92.01 ± 0.25	**92.85 ± 0.30**	**93.08 ± 0.18**	**92.56 ± 0.19**	–	–
	OmniCopreSheaf	**91.93 ± 0.73**	**92.61 ± 0.49**	89.46 ± 0.47	**90.35 ± 0.39**	–	–
	OmniCopreSheaf-T	89.70 ± 1.09	**92.20 ± 0.28**	**93.25 ± 0.08**	**93.35 ± 0.27**	–	–
	OmniDeepSheaf	**80.58 ± 1.49**	**85.20 ± 0.78**	**79.86 ± 1.10**	79.25 ± 0.72	–	–
	OmniSheafTransformer	**82.77 ± 0.47**	81.89 ± 0.34	**84.34 ± 0.97**	**86.61 ± 0.24**	–	–
Squirrel	**OmniPhaseSheaf**	41.41 ± 0.73	**41.41 ± 1.16**	40.97 ± 0.62	**41.79 ± 0.16**	41.11 ± 0.67	**42.23 ± 0.16**
	OmniPhaseSheaf-T	40.60 ± 0.91	**45.19 ± 1.99**	40.45 ± 1.17	42.98 ± 0.46	**44.75 ± 1.72**	**43.65 ± 1.62**
	OmniCopreSheaf	**43.08 ± 3.92**	**44.62 ± 1.36**	41.50 ± 1.65	42.77 ± 2.30	**44.02 ± 1.26**	42.54 ± 1.94
	OmniCopreSheaf-T	**44.99 ± 1.62**	**43.14 ± 1.66**	42.83 ± 0.92	**44.24 ± 0.78**	42.84 ± 2.78	42.02 ± 3.68
	OmniDeepSheaf	**40.37 ± 0.72**	**40.60 ± 1.11**	39.93 ± 0.90	39.41 ± 0.71	**41.04 ± 0.54**	40.07 ± 1.87
	OmniSheafTransformer	42.83 ± 0.54	**44.60 ± 1.17**	43.64 ± 0.70	**44.46 ± 1.18**	**43.72 ± 0.62**	42.99 ± 1.38
email-EU	**OmniPhaseSheaf**	**62.48 ± 2.44**	51.55 ± 1.34	50.61 ± 0.33	**54.93 ± 3.31**	**63.43 ± 7.09**	49.93 ± 5.59
	OmniPhaseSheaf-T	**83.54 ± 0.50**	82.73 ± 1.88	**85.56 ± 0.83**	83.27 ± 1.06	**83.54 ± 1.49**	83.40 ± 1.51
	OmniCopreSheaf	56.28 ± 12.02	55.47 ± 7.69	51.55 ± 14.27	**67.34 ± 4.77**	**71.66 ± 0.99**	**57.89 ± 0.87**
	OmniCopreSheaf-T	56.01 ± 2.57	**74.49 ± 2.62**	63.16 ± 2.64	**73.28 ± 2.62**	70.45 ± 6.93	**76.65 ± 2.25**
	OmniDeepSheaf	42.24 ± 5.27	58.57 ± 7.77	**64.10 ± 2.02**	62.08 ± 4.63	**63.83 ± 1.88**	**62.62 ± 2.39**
	OmniSheafTransformer	50.61 ± 0.99	**80.43 ± 0.76**	**79.22 ± 1.66**	67.88 ± 2.50	68.29 ± 0.38	**68.42 ± 7.90**
Telegram	**OmniPhaseSheaf**	80.13 ± 8.93	**80.13 ± 2.40**	**80.77 ± 8.74**	76.28 ± 5.05	**87.82 ± 3.95**	71.79 ± 3.95
	OmniPhaseSheaf-T	82.05 ± 5.51	**84.62 ± 1.57**	**83.97 ± 4.80**	82.05 ± 2.40	80.77 ± 1.57	**82.69 ± 2.72**
	OmniCopreSheaf	66.67 ± 5.05	**85.90 ± 4.80**	80.13 ± 6.35	**86.54 ± 2.72**	**80.77 ± 4.15**	77.56 ± 0.91
	OmniCopreSheaf-T	78.85 ± 6.84	**83.97 ± 0.91**	**83.33 ± 1.81**	**83.33 ± 5.94**	**83.33 ± 0.91**	**80.13 ± 5.05**
	OmniDeepSheaf	60.26 ± 3.63	60.26 ± 1.81	**64.74 ± 6.54**	60.90 ± 5.05	**63.46 ± 1.57**	**64.74 ± 2.40**
	OmniSheafTransformer	**60.26 ± 1.81**	**67.95 ± 3.95**	**67.95 ± 2.40**	**67.95 ± 3.95**	58.97 ± 3.27	60.26 ± 9.20
Chameleon	**OmniPhaseSheaf**	**43.30 ± 4.83**	38.75 ± 4.69	38.64 ± 1.91	**42.73 ± 3.68**	37.52 ± 1.27	**44.84 ± 3.50**
	OmniPhaseSheaf-T	41.41 ± 4.32	**41.50 ± 5.17**	**41.47 ± 2.71**	**43.39 ± 4.49**	40.51 ± 2.83	39.83 ± 4.33
	OmniCopreSheaf	39.98 ± 5.49	**42.57 ± 2.71**	**45.06 ± 5.94**	**43.07 ± 2.26**	40.12 ± 3.59	41.79 ± 3.40
	OmniCopreSheaf-T	**43.16 ± 3.11**	41.58 ± 3.89	**44.97 ± 5.82**	40.84 ± 1.72	40.64 ± 1.59	**44.62 ± 1.81**
	OmniDeepSheaf	41.77 ± 3.64	**41.96 ± 3.97**	41.38 ± 5.11	**44.70 ± 3.45**	40.73 ± 1.80	**41.81 ± 4.17**
	OmniSheafTransformer	42.57 ± 0.92	**44.63 ± 2.59**	43.23 ± 1.57	**45.94 ± 4.05**	**44.26 ± 1.14**	40.91 ± 2.71
email-Enron	**OmniPhaseSheaf**	56.76 ± 4.41	49.55 ± 2.55	**66.67 ± 4.59**	**57.66 ± 6.74**	53.15 ± 4.59	**64.86 ± 2.21**
	OmniPhaseSheaf-T	**73.87 ± 5.55**	**73.87 ± 4.59**	46.85 ± 21.77	**53.15 ± 12.74**	**57.66 ± 3.37**	35.14 ± 4.41
	OmniCopreSheaf	36.04 ± 5.55	53.15 ± 12.15	55.86 ± 6.37	**58.56 ± 4.59**	**60.36 ± 4.59**	**58.56 ± 8.92**
	OmniCopreSheaf-T	**54.05 ± 5.84**	**58.56 ± 2.55**	51.35 ± 7.96	**61.26 ± 6.37**	51.35 ± 10.11	49.55 ± 8.92
	OmniDeepSheaf	53.15 ± 7.75	52.25 ± 1.27	**59.46 ± 5.84**	**58.56 ± 14.36**	46.85 ± 14.19	**62.16 ± 9.62**
	OmniSheafTransformer	55.86 ± 16.71	**62.16 ± 9.62**	52.25 ± 6.74	**67.57 ± 7.96**	**67.57 ± 6.62**	60.36 ± 7.09
Cornell	**OmniPhaseSheaf**	73.87 ± 2.55	71.17 ± 6.74	**80.18 ± 2.55**	**75.68 ± 2.21**	**76.58 ± 4.59**	74.77 ± 1.27
	OmniPhaseSheaf-T	**74.77 ± 4.59**	**75.68 ± 4.41**	73.87 ± 5.55	**75.68 ± 2.21**	**75.68 ± 2.21**	**77.48 ± 3.37**
	OmniCopreSheaf	**81.08 ± 6.62**	73.87 ± 3.37	74.77 ± 3.37	**78.38 ± 2.21**	**77.48 ± 1.27**	76.58 ± 1.27
	OmniCopreSheaf-T	**77.48 ± 4.59**	71.17 ± 5.55	**72.97 ± 0.00**	**75.68 ± 2.21**	**75.68 ± 3.82**	**75.68 ± 3.82**
	OmniDeepSheaf	74.77 ± 3.37	69.37 ± 2.55	**76.58 ± 1.27**	**77.48 ± 2.55**	**79.28 ± 4.59**	**76.58 ± 6.37**
	OmniSheafTransformer	71.17 ± 1.27	**75.68 ± 2.21**	**78.38 ± 2.21**	74.77 ± 8.35	**76.58 ± 1.27**	**76.58 ± 5.55**
Wisconsin	**OmniPhaseSheaf**	79.08 ± 2.45	**86.93 ± 5.62**	84.31 ± 6.98	81.05 ± 6.06	**85.62 ± 3.33**	**84.31 ± 5.55**
	OmniPhaseSheaf-T	80.39 ± 4.24	**86.27 ± 5.77**	81.70 ± 4.89	**84.97 ± 4.62**	81.70 ± 3.33	**83.01 ± 4.89**
	OmniCopreSheaf	83.01 ± 2.45	81.70 ± 0.92	**84.31 ± 4.24**	**85.62 ± 4.03**	**84.97 ± 3.33**	**84.31 ± 5.77**
	OmniCopreSheaf-T	**81.05 ± 10.66**	81.05 ± 3.33	79.74 ± 4.03	**81.70 ± 2.45**	**84.97 ± 0.92**	**81.70 ± 8.22**
	OmniDeepSheaf	83.66 ± 1.85	82.35 ± 5.77	83.01 ± 4.03	**85.62 ± 1.85**	**85.62 ± 5.15**	**86.27 ± 5.77**
	OmniSheafTransformer	82.35 ± 3.20	**84.31 ± 4.24**	81.05 ± 6.47	83.66 ± 0.92	**83.66 ± 4.03**	**86.27 ± 4.80**
Amazon-ratings	**OmniPhaseSheaf**	**48.98 ± 0.41**	47.97 ± 0.22	**49.26 ± 0.34**	**49.21 ± 1.55**	–	–
	OmniPhaseSheaf-T	47.39 ± 0.30	**48.27 ± 0.43**	**48.47 ± 0.28**	**48.49 ± 0.58**	–	–
	OmniCopreSheaf	**40.45 ± 0.20**	**44.01 ± 0.24**	–	**46.59 ± 1.21**	–	–
	OmniCopreSheaf-T	**47.84 ± 0.68**	**48.17 ± 0.47**	47.62 ± 0.85	**48.63 ± 0.90**	–	–
	OmniDeepSheaf	**48.81 ± 0.05**	48.32 ± 0.70	**51.61 ± 0.73**	**49.00 ± 0.38**	–	–
	OmniSheafTransformer	44.81 ± 0.19	**48.59 ± 0.81**	**47.99 ± 0.67**	**46.51 ± 0.39**	–	–
Texas	**OmniPhaseSheaf**	77.48 ± 6.74	**81.98 ± 7.75**	**80.18 ± 6.74**	78.38 ± 2.21	**82.88 ± 6.74**	**81.98 ± 1.27**
	OmniPhaseSheaf-T	76.58 ± 1.27	77.48 ± 2.55	78.38 ± 2.21	**81.08 ± 3.82**	**80.18 ± 2.55**	**82.88 ± 1.27**
	OmniCopreSheaf	**81.98 ± 3.37**	74.77 ± 2.55	79.28 ± 7.09	**81.08 ± 7.64**	**81.98 ± 3.37**	**83.78 ± 5.84**
	OmniCopreSheaf-T	**81.08 ± 4.41**	80.18 ± 6.74	80.18 ± 5.55	**83.78 ± 3.82**	**81.98 ± 5.10**	78.38 ± 6.62
	OmniDeepSheaf	76.58 ± 4.59	81.08 ± 6.62	81.98 ± 5.10	**82.88 ± 6.37**	**81.98 ± 7.75**	**83.78 ± 5.84**
	OmniSheafTransformer	76.58 ± 2.55	**81.98 ± 2.55**	**83.78 ± 4.41**	81.08 ± 5.84	79.28 ± 3.37	**81.98 ± 3.37**
Citeseer	**OmniPhaseSheaf**	**76.61 ± 1.78**	75.97 ± 0.43	76.33 ± 0.97	**76.41 ± 0.77**	**77.54 ± 0.69**	75.60 ± 1.90
	OmniPhaseSheaf-T	76.09 ± 1.14	76.33 ± 0.95	**76.49 ± 0.25**	**76.81 ± 0.34**	**76.81 ± 0.45**	**76.81 ± 0.39**
	OmniCopreSheaf	74.36 ± 2.22	**76.21 ± 1.39**	**75.48 ± 1.54**	74.88 ± 0.26	**76.25 ± 0.94**	74.19 ± 2.02
	OmniCopreSheaf-T	74.68 ± 1.43	75.81 ± 0.94	**76.05 ± 0.74**	75.93 ± 0.73	**76.93 ± 0.52**	**76.97 ± 0.77**
	OmniDeepSheaf	76.25 ± 0.35	76.25 ± 1.63	76.41 ± 0.28	**77.62 ± 1.38**	**77.17 ± 0.62**	**76.97 ± 0.97**
	OmniSheafTransformer	**77.38 ± 0.50**	**76.77 ± 0.58**	**76.85 ± 0.28**	76.77 ± 0.50	76.45 ± 0.77	76.37 ± 1.22
Cora	**OmniPhaseSheaf**	76.42 ± 0.54	84.05 ± 1.99	**85.82 ± 0.48**	**85.38 ± 1.15**	85.08 ± 0.87	**85.57 ± 0.49**
	OmniPhaseSheaf-T	**86.56 ± 1.46**	**86.21 ± 1.50**	**85.92 ± 1.11**	**86.21 ± 1.00**	85.57 ± 1.65	85.72 ± 0.89
	OmniCopreSheaf	84.88 ± 1.81	**86.56 ± 1.07**	**85.33 ± 1.87**	85.13 ± 1.12	**85.52 ± 1.15**	85.08 ± 1.03
	OmniCopreSheaf-T	85.57 ± 1.30	**86.71 ± 0.63**	84.74 ± 1.32	**85.87 ± 1.22**	84.84 ± 0.94	**85.87 ± 0.49**
	OmniDeepSheaf	84.15 ± 1.22	86.12 ± 0.55	**86.95 ± 0.57**	**86.95 ± 0.87**	**86.16 ± 0.70**	**86.21 ± 0.97**
	OmniSheafTransformer	**86.16 ± 1.95**	**86.31 ± 1.46**	85.77 ± 1.39	85.38 ± 1.45	**86.51 ± 1.09**	85.77 ± 0.87

Table 68: Ablation on the stalk dimension d (hyperedge classification). Test macro-F1 (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given d, all other hyperparameters re-tuned. Colors mark the first, second, and third d per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
Molecular-1	**OmniPhaseSheaf**	**65.88 ± 1.52**	36.27 ± 4.29	–	**72.58 ± 0.52**	–	**70.77 ± 0.51**
	OmniPhaseSheaf-T	89.62 ± 0.18	89.24 ± 0.25	**91.09 ± 0.32**	89.89 ± 0.15	**90.71 ± 0.10**	**89.97 ± 0.26**
	OmniCopreSheaf	84.58 ± 0.63	**85.21 ± 0.64**	74.68 ± 1.42	**86.49 ± 0.30**	84.78 ± 0.32	**85.64 ± 0.16**
	OmniCopreSheaf-T	86.42 ± 0.18	**87.80 ± 0.31**	**87.67 ± 0.42**	87.37 ± 0.44	87.32 ± 0.13	**90.29 ± 0.12**
	OmniDeepSheaf	57.17 ± 5.39	77.23 ± 0.47	**77.83 ± 0.49**	**77.66 ± 0.94**	**78.19 ± 0.60**	74.57 ± 3.52
	OmniSheafTransformer	81.39 ± 0.79	79.94 ± 0.92	**82.09 ± 0.39**	**81.47 ± 0.18**	79.18 ± 1.06	**81.72 ± 0.42**
Molecular-2	**OmniPhaseSheaf**	79.33 ± 3.61	79.78 ± 1.82	**87.54 ± 1.99**	**84.28 ± 1.16**	78.57 ± 3.56	**85.82 ± 1.40**
	OmniPhaseSheaf-T	**89.40 ± 1.25**	88.04 ± 2.02	**90.46 ± 0.76**	88.73 ± 1.42	**89.87 ± 1.65**	88.35 ± 1.42
	OmniCopreSheaf	78.05 ± 4.75	85.63 ± 1.29	**87.48 ± 1.39**	86.34 ± 0.86	**87.87 ± 1.67**	**86.54 ± 1.87**
	OmniCopreSheaf-T	83.88 ± 2.75	**84.68 ± 2.57**	**86.32 ± 1.67**	83.92 ± 2.59	**88.33 ± 2.54**	83.53 ± 1.61
	OmniDeepSheaf	**88.91 ± 1.72**	**88.45 ± 2.56**	85.84 ± 1.98	82.58 ± 5.96	85.95 ± 2.47	**89.94 ± 1.64**
	OmniSheafTransformer	86.80 ± 1.36	**88.20 ± 1.99**	**90.02 ± 1.62**	87.03 ± 2.05	**90.34 ± 2.17**	88.04 ± 1.28

Table 69: Ablation on the stalk dimension d (synthetic directed hypergraphs). Test accuracy (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given d, all other hyperparameters re-tuned. Colors mark the first, second, and third d per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
$I_o=10$	**OmniPhaseSheaf**	92.00 ± 2.61	**96.00 ± 2.36**	**96.80 ± 0.65**	**96.27 ± 3.02**	**96.00 ± 2.61**	**96.80 ± 0.00**
	OmniPhaseSheaf-T	**98.13 ± 1.51**	96.80 ± 1.96	97.07 ± 0.38	**97.07 ± 1.00**	97.07 ± 0.75	**98.40 ± 1.31**
	OmniCopreSheaf	81.33 ± 17.40	**98.67 ± 0.38**	**97.60 ± 0.65**	97.07 ± 0.75	**98.13 ± 1.00**	**98.13 ± 0.38**
	OmniCopreSheaf-T	**96.80 ± 1.73**	94.93 ± 2.64	**97.07 ± 0.75**	96.27 ± 1.00	96.53 ± 3.77	**98.13 ± 1.00**
	OmniDeepSheaf	30.40 ± 1.73	**61.87 ± 12.97**	24.27 ± 1.36	**72.80 ± 8.84**	48.53 ± 4.63	**67.47 ± 12.36**
	OmniSheafTransformer	63.73 ± 11.85	71.73 ± 10.25	63.47 ± 20.62	**77.33 ± 21.31**	**88.80 ± 2.99**	**82.93 ± 2.72**
$I_o=30$	**OmniPhaseSheaf**	**99.20 ± 0.65**	**99.47 ± 0.75**	98.13 ± 1.00	**99.73 ± 0.38**	**99.47 ± 0.75**	98.67 ± 1.00
	OmniPhaseSheaf-T	**99.47 ± 0.38**	**99.47 ± 0.75**	98.67 ± 0.75	98.93 ± 0.38	**99.20 ± 0.65**	**100.00 ± 0.00**
	OmniCopreSheaf	**99.47 ± 0.75**	**98.67 ± 1.00**	98.40 ± 1.31	**99.73 ± 0.38**	97.87 ± 1.36	98.40 ± 1.31
	OmniCopreSheaf-T	**98.93 ± 1.00**	**99.47 ± 0.38**	98.67 ± 1.89	**99.20 ± 0.65**	98.40 ± 1.13	98.13 ± 1.64
	OmniDeepSheaf	26.40 ± 4.08	28.27 ± 4.90	**54.40 ± 6.53**	30.40 ± 1.96	**37.60 ± 3.46**	**34.93 ± 3.22**
	OmniSheafTransformer	**35.47 ± 7.54**	**69.60 ± 13.50**	32.00 ± 5.69	**52.27 ± 19.64**	28.80 ± 4.28	28.27 ± 4.90
$I_o=50$	**OmniPhaseSheaf**	**99.73 ± 0.38**	**99.20 ± 0.65**	**99.20 ± 0.65**	98.93 ± 1.00	98.67 ± 1.36	**100.00 ± 0.00**
	OmniPhaseSheaf-T	**99.73 ± 0.38**	**99.47 ± 0.75**	**99.47 ± 0.38**	98.40 ± 1.13	97.87 ± 1.00	**99.20 ± 1.13**
	OmniCopreSheaf	97.87 ± 1.64	97.87 ± 1.64	**98.93 ± 1.51**	**99.47 ± 0.75**	**100.00 ± 0.00**	98.40 ± 1.73
	OmniCopreSheaf-T	**99.47 ± 0.75**	97.87 ± 2.00	**99.73 ± 0.38**	**99.20 ± 1.13**	**99.47 ± 0.38**	**99.73 ± 0.38**
	OmniDeepSheaf	27.73 ± 2.29	**33.60 ± 2.99**	**35.47 ± 2.29**	**38.93 ± 7.57**	32.80 ± 4.57	30.67 ± 6.07
	OmniSheafTransformer	**38.13 ± 14.76**	**66.13 ± 11.54**	34.13 ± 2.64	22.13 ± 5.32	29.60 ± 5.58	**86.67 ± 5.89**

Table 70: Ablation on the stalk dimension d (undirected hypergraphs). Test accuracy (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given d, all other hyperparameters re-tuned. Colors mark the first, second, and third d per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
Cora	**OmniDeepSheaf**	77.79 ± 1.09	76.81 ± 0.48	**78.73 ± 1.15**	76.76 ± 1.50	**78.24 ± 1.29**	**78.58 ± 1.47**
	OmniSheafTransformer	78.09 ± 1.03	**79.17 ± 1.28**	78.09 ± 0.25	**79.76 ± 1.81**	78.68 ± 1.12	**79.37 ± 1.55**
Citeseer	**OmniDeepSheaf**	72.71 ± 0.51	74.11 ± 1.68	**74.48 ± 0.70**	73.75 ± 0.40	**74.72 ± 0.92**	**74.64 ± 1.71**
	OmniSheafTransformer	72.18 ± 0.92	72.10 ± 1.59	71.62 ± 1.21	**74.52 ± 0.65**	**72.46 ± 0.68**	**72.42 ± 0.25**
Pubmed	**OmniDeepSheaf**	**87.76 ± 0.20**	87.25 ± 0.22	87.61 ± 0.10	**87.69 ± 0.42**	**87.86 ± 0.13**	87.48 ± 0.14
	OmniSheafTransformer	86.74 ± 0.35	87.39 ± 0.28	**87.69 ± 0.28**	**87.51 ± 0.24**	87.48 ± 0.24	**87.82 ± 0.29**
Cora-CA	**OmniDeepSheaf**	80.50 ± 0.84	81.54 ± 1.11	**82.18 ± 1.72**	81.98 ± 0.53	**82.82 ± 0.97**	**82.27 ± 0.53**
	OmniSheafTransformer	**83.21 ± 1.00**	**83.01 ± 0.91**	82.72 ± 0.79	**83.16 ± 1.03**	81.04 ± 0.18	81.09 ± 0.96
DBLP-CA	**OmniDeepSheaf**	91.41 ± 0.28	**91.51 ± 0.14**	**91.47 ± 0.11**	**91.64 ± 0.16**	–	–
	OmniSheafTransformer	91.31 ± 0.22	**91.70 ± 0.21**	**91.62 ± 0.27**	**91.62 ± 0.14**	–	–
Senate	**OmniDeepSheaf**	**80.28 ± 1.15**	77.93 ± 2.89	77.46 ± 3.98	**80.28 ± 1.99**	**78.87 ± 1.15**	**80.28 ± 2.30**
	OmniSheafTransformer	77.93 ± 0.66	**78.87 ± 1.99**	**81.22 ± 1.33**	75.59 ± 3.70	**80.28 ± 3.04**	77.46 ± 1.15
House	**OmniDeepSheaf**	77.19 ± 0.29	76.78 ± 2.53	**77.71 ± 1.77**	77.30 ± 0.96	**77.81 ± 1.27**	**77.61 ± 0.29**
	OmniSheafTransformer	76.88 ± 1.02	76.88 ± 1.54	**77.50 ± 0.39**	77.30 ± 0.77	**77.61 ± 1.27**	**77.40 ± 1.10**
Congress	**OmniDeepSheaf**	**94.34 ± 1.29**	93.80 ± 0.40	**94.73 ± 0.58**	94.11 ± 0.55	**94.11 ± 1.59**	94.03 ± 0.48
	OmniSheafTransformer	92.48 ± 0.67	**94.81 ± 0.55**	**95.04 ± 0.29**	**95.19 ± 0.77**	94.57 ± 0.67	94.11 ± 1.10

Table 71: Ablation on the number of convolutional layers L_{conv} (directed node classification). Test accuracy (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given L_{conv}, all other hyperparameters re-tuned. Colors mark the first, second, and third L_{conv} per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$L_{\text{conv}} = 1$	$L_{\text{conv}} = 2$	$L_{\text{conv}} = 3$	$L_{\text{conv}} = 4$	$L_{\text{conv}} = 5$
Roman-empire	**OmniPhaseSheaf**	**86.78 ± 0.21**	**85.77 ± 0.28**	**87.82 ± 0.25**	–	–
	OmniPhaseSheaf-T	**90.30 ± 0.18**	**93.08 ± 0.18**	**92.85 ± 0.30**	–	–
	OmniCopreSheaf	**90.35 ± 0.39**	**88.29 ± 0.19**	**92.61 ± 0.49**	–	–
	OmniCopreSheaf-T	**90.88 ± 0.35**	**93.35 ± 0.27**	**93.15 ± 0.38**	–	–
	OmniDeepSheaf	**79.44 ± 0.08**	**79.86 ± 1.10**	**85.20 ± 0.78**	–	–
	OmniSheafTransformer	**82.89 ± 0.64**	**85.50 ± 0.46**	**86.61 ± 0.24**	–	–
Squirrel	**OmniPhaseSheaf**	**40.60 ± 0.71**	40.01 ± 2.76	**40.97 ± 0.62**	38.53 ± 2.48	**42.23 ± 0.16**
	OmniPhaseSheaf-T	**41.56 ± 0.65**	**45.19 ± 1.99**	**44.75 ± 1.72**	–	–
	OmniCopreSheaf	42.10 ± 1.85	**44.02 ± 1.26**	42.76 ± 2.50	**42.77 ± 2.30**	**43.29 ± 2.76**
	OmniCopreSheaf-T	**44.24 ± 0.78**	**44.01 ± 1.26**	**42.83 ± 0.92**	–	–
	OmniDeepSheaf	**40.82 ± 0.87**	**40.60 ± 1.11**	40.59 ± 1.02	**41.26 ± 0.72**	40.52 ± 1.13
	OmniSheafTransformer	**42.83 ± 0.54**	**42.44 ± 1.87**	**44.46 ± 1.18**	–	–
email-EU	**OmniPhaseSheaf**	49.93 ± 5.59	**63.43 ± 7.09**	49.53 ± 4.36	**50.47 ± 3.12**	**62.48 ± 2.44**
	OmniPhaseSheaf-T	**73.28 ± 3.19**	**85.56 ± 0.83**	**83.40 ± 1.19**	–	–
	OmniCopreSheaf	58.84 ± 4.49	**71.66 ± 0.99**	55.33 ± 2.32	**70.58 ± 2.69**	**67.34 ± 4.77**
	OmniCopreSheaf-T	**66.13 ± 0.83**	**76.65 ± 2.25**	**74.49 ± 2.62**	–	–
	OmniDeepSheaf	56.95 ± 3.34	**64.10 ± 2.02**	**62.62 ± 2.39**	56.14 ± 4.51	**63.83 ± 1.88**
	OmniSheafTransformer	**58.30 ± 8.10**	**79.22 ± 1.66**	**68.42 ± 7.90**	–	–
Telegram	**OmniPhaseSheaf**	80.13 ± 2.40	**84.62 ± 2.72**	**80.77 ± 8.74**	**87.82 ± 3.95**	76.28 ± 5.05
	OmniPhaseSheaf-T	**82.69 ± 2.72**	**82.05 ± 2.40**	**80.77 ± 1.57**	–	–
	OmniCopreSheaf	77.56 ± 0.91	**79.49 ± 3.27**	**80.77 ± 4.15**	75.64 ± 0.91	**86.54 ± 2.72**
	OmniCopreSheaf-T	**82.05 ± 1.81**	**83.97 ± 0.91**	**83.33 ± 1.81**	–	–
	OmniDeepSheaf	**63.46 ± 1.57**	**64.74 ± 2.40**	60.90 ± 5.05	**66.67 ± 4.80**	60.26 ± 3.63
	OmniSheafTransformer	**60.26 ± 1.81**	**67.95 ± 3.95**	**67.95 ± 2.40**	–	–
Chameleon	**OmniPhaseSheaf**	37.92 ± 2.99	**41.34 ± 2.72**	**44.84 ± 3.50**	37.52 ± 1.27	**42.73 ± 3.68**
	OmniPhaseSheaf-T	**40.62 ± 3.78**	**41.50 ± 5.17**	**40.51 ± 2.83**	–	–
	OmniCopreSheaf	**42.07 ± 3.23**	40.92 ± 5.05	**44.99 ± 6.02**	**43.07 ± 2.26**	41.79 ± 3.40
	OmniCopreSheaf-T	**40.84 ± 1.72**	**45.60 ± 4.83**	**44.62 ± 1.81**	–	–
	OmniDeepSheaf	39.16 ± 3.65	**43.86 ± 3.27**	**41.81 ± 4.17**	40.29 ± 1.95	**41.96 ± 3.97**
	OmniSheafTransformer	**42.83 ± 3.76**	**40.91 ± 2.71**	**42.71 ± 1.63**	–	–
email-Enron	**OmniPhaseSheaf**	53.15 ± 5.10	**64.86 ± 2.21**	57.66 ± 6.74	**63.96 ± 4.59**	**66.67 ± 4.59**
	OmniPhaseSheaf-T	**64.86 ± 2.21**	**65.77 ± 3.37**	**73.87 ± 5.55**	–	–
	OmniCopreSheaf	43.24 ± 0.00	**60.36 ± 4.59**	56.76 ± 3.82	**58.56 ± 7.75**	**58.56 ± 4.59**
	OmniCopreSheaf-T	**56.76 ± 4.41**	**62.16 ± 0.00**	**61.26 ± 6.37**	–	–
	OmniDeepSheaf	56.76 ± 6.62	58.56 ± 14.36	**62.16 ± 9.62**	**62.16 ± 8.83**	**59.46 ± 5.84**
	OmniSheafTransformer	**57.66 ± 3.37**	**67.57 ± 6.62**	**60.36 ± 7.09**	–	–
Cornell	**OmniPhaseSheaf**	**76.58 ± 1.27**	**77.48 ± 2.55**	**80.18 ± 2.55**	75.68 ± 2.21	73.87 ± 1.27
	OmniPhaseSheaf-T	**73.87 ± 5.55**	**71.17 ± 3.37**	**75.68 ± 2.21**	–	–
	OmniCopreSheaf	**76.58 ± 1.27**	71.17 ± 6.37	**76.58 ± 1.27**	**78.38 ± 2.21**	**77.48 ± 4.59**
	OmniCopreSheaf-T	**78.38 ± 5.84**	**75.68 ± 3.82**	**72.97 ± 0.00**	–	–
	OmniDeepSheaf	**76.58 ± 1.27**	**79.28 ± 4.59**	**76.58 ± 6.37**	**77.48 ± 2.55**	**77.48 ± 2.55**
	OmniSheafTransformer	**76.58 ± 1.27**	**76.58 ± 5.55**	**75.68 ± 3.82**	–	–
Wisconsin	**OmniPhaseSheaf**	**85.62 ± 3.33**	81.05 ± 7.90	**86.93 ± 5.62**	**85.62 ± 5.62**	81.05 ± 6.06
	OmniPhaseSheaf-T	**86.93 ± 3.33**	**81.70 ± 4.03**	**81.70 ± 3.33**	–	–
	OmniCopreSheaf	**84.31 ± 5.77**	**83.01 ± 2.45**	**84.31 ± 4.24**	**81.70 ± 0.92**	**83.01 ± 8.82**
	OmniCopreSheaf-T	**81.70 ± 2.45**	**81.05 ± 10.66**	**83.66 ± 4.03**	–	–
	OmniDeepSheaf	**85.62 ± 1.85**	82.35 ± 5.77	**84.97 ± 2.45**	**84.31 ± 4.80**	**85.62 ± 3.33**
	OmniSheafTransformer	**83.66 ± 0.92**	**83.66 ± 4.03**	**83.01 ± 5.62**	–	–
Amazon-ratings	**OmniPhaseSheaf**	**48.33 ± 0.42**	**49.26 ± 0.34**	**49.31 ± 0.42**	–	–
	OmniPhaseSheaf-T	**48.47 ± 0.28**	**48.49 ± 0.58**	**49.90 ± 0.83**	–	–
	OmniCopreSheaf	**44.01 ± 0.24**	**43.91 ± 0.73**	**46.59 ± 1.21**	–	–
	OmniCopreSheaf-T	**47.65 ± 0.67**	**47.67 ± 0.53**	**48.63 ± 0.90**	–	–
	OmniDeepSheaf	**49.00 ± 0.38**	**51.61 ± 0.73**	**47.87 ± 0.27**	–	–
	OmniSheafTransformer	**47.07 ± 0.75**	**48.59 ± 0.81**	**47.99 ± 0.67**	–	–
Texas	**OmniPhaseSheaf**	**81.08 ± 2.21**	**82.88 ± 8.35**	**82.88 ± 6.74**	**81.98 ± 7.75**	78.38 ± 2.21
	OmniPhaseSheaf-T	**82.88 ± 1.27**	**78.38 ± 2.21**	**80.18 ± 2.55**	–	–
	OmniCopreSheaf	**81.08 ± 7.64**	79.28 ± 7.09	**81.98 ± 3.37**	**83.78 ± 5.84**	75.68 ± 2.21
	OmniCopreSheaf-T	**81.98 ± 5.10**	**80.18 ± 1.27**	**81.08 ± 4.41**	–	–
	OmniDeepSheaf	**84.68 ± 5.55**	**83.78 ± 5.84**	**84.68 ± 5.10**	81.08 ± 6.62	**81.98 ± 5.10**
	OmniSheafTransformer	**81.08 ± 5.84**	**81.98 ± 3.37**	**83.78 ± 4.41**	–	–
Citeseer	**OmniPhaseSheaf**	74.64 ± 1.81	75.08 ± 0.54	**76.41 ± 0.77**	**77.54 ± 0.69**	**76.41 ± 1.05**
	OmniPhaseSheaf-T	**76.89 ± 0.63**	**76.81 ± 0.45**	**76.81 ± 0.34**	–	–
	OmniCopreSheaf	**75.00 ± 0.20**	74.96 ± 1.09	74.03 ± 1.71	**75.48 ± 1.54**	**76.25 ± 0.94**
	OmniCopreSheaf-T	**76.05 ± 0.74**	**76.93 ± 0.52**	**75.60 ± 1.38**	–	–
	OmniDeepSheaf	**77.62 ± 1.38**	**77.17 ± 0.62**	76.09 ± 1.72	**76.25 ± 0.35**	75.68 ± 1.58
	OmniSheafTransformer	**76.85 ± 0.28**	**76.25 ± 0.85**	**76.37 ± 1.22**	–	–
Cora	**OmniPhaseSheaf**	84.10 ± 0.07	**85.57 ± 0.49**	85.38 ± 1.15	**85.82 ± 0.48**	**85.82 ± 1.71**
	OmniPhaseSheaf-T	**86.21 ± 1.50**	**84.49 ± 0.36**	**85.72 ± 0.89**	–	–
	OmniCopreSheaf	**86.46 ± 1.22**	**85.52 ± 1.15**	85.33 ± 1.87	84.84 ± 1.74	**85.97 ± 1.26**
	OmniCopreSheaf-T	**86.71 ± 0.63**	**85.57 ± 1.30**	**84.74 ± 1.32**	–	–
	OmniDeepSheaf	**86.95 ± 0.87**	**86.76 ± 0.54**	**86.95 ± 0.57**	86.21 ± 0.97	**86.31 ± 1.26**
	OmniSheafTransformer	**86.31 ± 1.46**	**86.16 ± 1.95**	**85.23 ± 0.84**	–	–

Table 72: Ablation on the number of convolutional layers L_{conv} (hyperedge classification). Test macro-F1 (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given L_{conv}, all other hyperparameters re-tuned. Colors mark the first, second, and third L_{conv} per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$L_{\text{conv}} = 1$	$L_{\text{conv}} = 2$	$L_{\text{conv}} = 3$	$L_{\text{conv}} = 4$	$L_{\text{conv}} = 5$
Molecular-1	**OmniPhaseSheaf**	**72.58 ± 0.52**	**70.77 ± 0.51**	24.25 ± 8.93	**65.88 ± 1.52**	–
	OmniPhaseSheaf-T	**89.97 ± 0.26**	**91.09 ± 0.32**	**90.71 ± 0.10**	–	–
	OmniCopreSheaf	**86.49 ± 0.30**	**84.78 ± 0.32**	69.33 ± 1.13	**85.21 ± 0.64**	75.51 ± 2.17
	OmniCopreSheaf-T	**90.29 ± 0.12**	**86.80 ± 0.55**	**87.56 ± 0.35**	–	–
	OmniDeepSheaf	**78.19 ± 0.60**	**76.91 ± 2.45**	**77.66 ± 0.94**	69.64 ± 5.79	74.66 ± 0.19
	OmniSheafTransformer	**81.47 ± 0.18**	**82.09 ± 0.39**	**81.39 ± 0.79**	–	–
Molecular-2	**OmniPhaseSheaf**	**87.54 ± 1.99**	78.25 ± 0.74	78.00 ± 3.23	**79.33 ± 3.61**	**79.32 ± 1.35**
	OmniPhaseSheaf-T	**86.71 ± 2.00**	**89.40 ± 1.25**	**89.87 ± 1.65**	–	–
	OmniCopreSheaf	**85.63 ± 1.29**	**86.34 ± 0.86**	**84.42 ± 3.73**	82.42 ± 2.81	83.77 ± 0.74
	OmniCopreSheaf-T	**88.33 ± 2.54**	**84.68 ± 2.57**	**84.20 ± 1.17**	–	–
	OmniDeepSheaf	**89.94 ± 1.64**	83.91 ± 1.55	**88.91 ± 1.72**	76.78 ± 5.05	**86.78 ± 2.11**
	OmniSheafTransformer	**90.02 ± 1.62**	**86.80 ± 1.36**	**90.34 ± 2.17**	–	–

Table 73: Ablation on the number of convolutional layers L_{conv} (synthetic directed hypergraphs). Test accuracy (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given L_{conv}, all other hyperparameters re-tuned. Colors mark the first, second, and third L_{conv} per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$L_{\text{conv}} = 1$	$L_{\text{conv}} = 2$	$L_{\text{conv}} = 3$	$L_{\text{conv}} = 4$	$L_{\text{conv}} = 5$
$I_o = 10$	**OmniPhaseSheaf**	61.60 ± 7.53	**90.67 ± 2.00**	**96.00 ± 2.61**	**96.00 ± 2.85**	**96.80 ± 0.65**
	OmniPhaseSheaf-T	**76.53 ± 2.29**	**97.07 ± 0.75**	**97.07 ± 0.38**	–	–
	OmniCopreSheaf	61.07 ± 4.73	**96.80 ± 1.13**	**98.13 ± 0.38**	**98.13 ± 1.00**	**97.07 ± 0.75**
	OmniCopreSheaf-T	**70.40 ± 4.28**	**95.73 ± 2.29**	**98.13 ± 1.00**	–	–
	OmniDeepSheaf	**48.53 ± 4.63**	24.80 ± 11.88	**67.47 ± 12.36**	**72.80 ± 8.84**	30.40 ± 5.18
	OmniSheafTransformer	**51.20 ± 2.61**	**88.80 ± 2.99**	**77.33 ± 21.31**	–	–
$I_o = 30$	**OmniPhaseSheaf**	78.13 ± 1.00	**98.13 ± 1.00**	**99.20 ± 0.65**	**99.20 ± 0.65**	**99.47 ± 0.75**
	OmniPhaseSheaf-T	**89.60 ± 2.85**	**99.47 ± 0.75**	**99.20 ± 0.65**	–	–
	OmniCopreSheaf	76.27 ± 1.51	**98.40 ± 1.31**	**98.40 ± 1.13**	**98.40 ± 0.65**	**98.93 ± 0.75**
	OmniCopreSheaf-T	**86.13 ± 4.90**	**99.20 ± 0.65**	**98.40 ± 1.13**	–	–
	OmniDeepSheaf	30.40 ± 1.96	26.40 ± 4.08	**34.93 ± 3.22**	**37.60 ± 3.46**	**54.40 ± 6.53**
	OmniSheafTransformer	**42.13 ± 6.34**	**52.27 ± 19.64**	**69.60 ± 13.50**	–	–
$I_o = 50$	**OmniPhaseSheaf**	86.67 ± 2.29	**98.40 ± 0.65**	**99.20 ± 0.65**	**99.20 ± 0.65**	**98.67 ± 1.00**
	OmniPhaseSheaf-T	**92.27 ± 1.00**	**99.47 ± 0.38**	**99.20 ± 1.13**	–	–
	OmniCopreSheaf	85.33 ± 3.35	**99.47 ± 0.75**	98.40 ± 1.73	**99.20 ± 0.65**	**98.93 ± 1.00**
	OmniCopreSheaf-T	**88.80 ± 1.13**	**99.47 ± 0.75**	**99.73 ± 0.38**	–	–
	OmniDeepSheaf	35.47 ± 2.29	22.13 ± 1.00	**37.33 ± 1.64**	**36.00 ± 5.69**	**38.93 ± 7.57**
	OmniSheafTransformer	**40.80 ± 4.57**	**66.13 ± 11.54**	**86.67 ± 5.89**	–	–

Table 74: Ablation on the number of convolutional layers L_{conv} (undirected hypergraphs). Test accuracy (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given L_{conv}, all other hyperparameters re-tuned. Colors mark the first, second, and third L_{conv} per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$L_{\text{conv}} = 1$	$L_{\text{conv}} = 2$	$L_{\text{conv}} = 3$	$L_{\text{conv}} = 4$	$L_{\text{conv}} = 5$
Cora	**OmniDeepSheaf**	**78.24 ± 1.29**	78.04 ± 1.64	**78.73 ± 1.15**	**78.58 ± 1.47**	76.81 ± 0.48
	OmniSheafTransformer	**78.78 ± 0.18**	**79.17 ± 1.28**	**78.68 ± 1.12**	–	–
Citeseer	**OmniDeepSheaf**	**74.72 ± 0.92**	**74.32 ± 0.89**	**74.64 ± 1.71**	74.11 ± 1.68	74.15 ± 0.90
	OmniSheafTransformer	**74.52 ± 0.65**	**72.99 ± 1.23**	**72.91 ± 0.99**	–	–
Pubmed	**OmniDeepSheaf**	**87.86 ± 0.13**	**87.61 ± 0.10**	**87.48 ± 0.14**	–	–
	OmniSheafTransformer	**87.48 ± 0.24**	**87.50 ± 0.38**	**87.82 ± 0.29**	–	–
Cora-CA	**OmniDeepSheaf**	**82.82 ± 0.97**	**82.27 ± 0.53**	**82.42 ± 1.87**	80.60 ± 0.86	79.27 ± 0.70
	OmniSheafTransformer	**83.01 ± 0.91**	**81.09 ± 0.96**	**82.03 ± 1.29**	–	–
DBLP-CA	**OmniDeepSheaf**	**91.64 ± 0.16**	**91.38 ± 0.14**	**91.51 ± 0.14**	–	–
	OmniSheafTransformer	**91.48 ± 0.15**	**91.62 ± 0.27**	**91.50 ± 0.25**	–	–
Senate	**OmniDeepSheaf**	77.46 ± 3.98	**80.28 ± 0.00**	**80.28 ± 1.99**	77.93 ± 0.66	**78.87 ± 1.15**
	OmniSheafTransformer	**78.87 ± 1.99**	**81.22 ± 1.33**	**77.00 ± 1.76**	–	–
House	**OmniDeepSheaf**	**77.81 ± 1.27**	**77.19 ± 0.29**	76.88 ± 1.54	**77.61 ± 1.68**	**77.61 ± 0.29**
	OmniSheafTransformer	**77.40 ± 1.10**	**76.37 ± 0.64**	**76.88 ± 1.54**	–	–
Congress	**OmniDeepSheaf**	**93.49 ± 0.76**	**94.73 ± 0.58**	–	–	–
	OmniSheafTransformer	**93.18 ± 1.08**	**95.04 ± 0.29**	–	–	–

Table 75: Ablation on the hidden width w (directed node classification). Test accuracy (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given w, all other hyperparameters re-tuned. Colors mark the first, second, and third w per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$w = 32$	$w = 64$	$w = 128$	$w = 256$	$w = 512$
Roman-empire	**OmniPhaseSheaf**	**87.82 ± 0.25**	**88.00 ± 0.06**	86.78 ± 0.21	–	–
	OmniPhaseSheaf-T	92.85 ± 0.30	93.08 ± 0.18	91.72 ± 0.41	–	–
	OmniCopreSheaf	89.46 ± 0.47	**92.61 ± 0.49**	**91.93 ± 0.73**	–	–
	OmniCopreSheaf-T	**93.15 ± 0.38**	**93.35 ± 0.27**	92.20 ± 0.28	–	–
	OmniDeepSheaf	77.88 ± 0.16	80.58 ± 1.49	**85.20 ± 0.78**	–	–
	OmniSheafTransformer	85.50 ± 0.46	**86.61 ± 0.24**	84.34 ± 0.97	–	–
Squirrel	**OmniPhaseSheaf**	**42.23 ± 0.16**	40.97 ± 0.62	**41.79 ± 0.16**	41.41 ± 0.73	39.55 ± 1.17
	OmniPhaseSheaf-T	41.26 ± 1.02	**45.19 ± 1.99**	43.65 ± 1.62	40.45 ± 1.17	–
	OmniCopreSheaf	41.72 ± 1.48	42.54 ± 1.94	**44.62 ± 1.36**	44.02 ± 1.26	**45.15 ± 2.11**
	OmniCopreSheaf-T	42.83 ± 0.92	43.64 ± 0.29	41.04 ± 2.15	**44.01 ± 1.26**	–
	OmniDeepSheaf	**41.04 ± 0.54**	39.93 ± 0.90	40.60 ± 1.11	**40.82 ± 0.87**	40.42 ± 3.15
	OmniSheafTransformer	41.25 ± 1.50	**44.46 ± 1.18**	**44.60 ± 1.17**	42.44 ± 1.87	–
email-EU	**OmniPhaseSheaf**	54.93 ± 3.31	51.55 ± 1.34	**62.48 ± 2.44**	**63.43 ± 7.09**	45.07 ± 5.70
	OmniPhaseSheaf-T	**85.56 ± 0.83**	**83.40 ± 1.51**	83.27 ± 1.06	79.08 ± 7.07	–
	OmniCopreSheaf	**57.89 ± 0.87**	**71.66 ± 0.99**	54.12 ± 9.96	56.28 ± 12.02	55.47 ± 7.69
	OmniCopreSheaf-T	**76.65 ± 2.25**	**70.45 ± 6.93**	64.51 ± 2.77	62.21 ± 2.50	–
	OmniDeepSheaf	62.62 ± 1.70	56.95 ± 3.34	62.62 ± 2.39	**64.10 ± 2.02**	54.39 ± 1.38
	OmniSheafTransformer	67.88 ± 2.50	**79.22 ± 1.66**	**80.43 ± 0.76**	47.10 ± 3.50	–
Telegram	**OmniPhaseSheaf**	**80.13 ± 2.40**	**80.77 ± 8.74**	71.79 ± 7.90	73.08 ± 0.00	76.28 ± 5.05
	OmniPhaseSheaf-T	80.77 ± 1.57	**82.69 ± 2.72**	**82.05 ± 5.05**	82.05 ± 5.51	–
	OmniCopreSheaf	**86.54 ± 2.72**	80.13 ± 6.35	**80.77 ± 4.15**	79.49 ± 0.91	77.56 ± 0.91
	OmniCopreSheaf-T	**83.97 ± 0.91**	**87.82 ± 4.80**	83.33 ± 5.94	81.41 ± 1.81	–
	OmniDeepSheaf	64.74 ± 2.40	63.46 ± 1.57	60.90 ± 5.05	**66.67 ± 4.80**	60.90 ± 3.27
	OmniSheafTransformer	**67.95 ± 3.95**	64.10 ± 2.40	58.97 ± 3.27	56.41 ± 7.90	–
Chameleon	**OmniPhaseSheaf**	**44.84 ± 3.50**	37.52 ± 1.27	**42.73 ± 3.68**	39.80 ± 1.18	37.54 ± 2.72
	OmniPhaseSheaf-T	**42.34 ± 2.14**	40.34 ± 3.13	40.46 ± 5.93	**41.50 ± 5.17**	–
	OmniCopreSheaf	40.92 ± 5.05	**44.99 ± 6.02**	41.79 ± 3.40	**43.07 ± 2.26**	42.97 ± 2.78
	OmniCopreSheaf-T	44.62 ± 1.81	40.84 ± 1.72	**45.60 ± 4.83**	**44.70 ± 3.45**	–
	OmniDeepSheaf	41.28 ± 3.31	**43.24 ± 2.99**	**41.96 ± 3.97**	41.81 ± 4.17	40.46 ± 2.57
	OmniSheafTransformer	40.91 ± 2.71	**42.71 ± 1.63**	**44.26 ± 1.14**	40.45 ± 3.10	–
email-Enron	**OmniPhaseSheaf**	60.36 ± 8.35	**66.67 ± 4.59**	**64.86 ± 2.21**	53.15 ± 5.10	56.76 ± 9.62
	OmniPhaseSheaf-T	**73.87 ± 5.55**	35.14 ± 16.66	64.86 ± 8.83	**65.77 ± 3.37**	–
	OmniCopreSheaf	**60.36 ± 4.59**	48.65 ± 2.21	53.15 ± 4.59	58.56 ± 7.75	55.86 ± 6.37
	OmniCopreSheaf-T	**61.26 ± 6.37**	54.05 ± 5.84	46.85 ± 2.55	50.45 ± 6.37	–
	OmniDeepSheaf	58.56 ± 14.36	**62.16 ± 8.83**	**62.16 ± 9.62**	46.85 ± 14.19	57.66 ± 10.19
	OmniSheafTransformer	60.36 ± 7.09	**67.57 ± 6.62**	**67.57 ± 7.96**	52.25 ± 6.74	–
Cornell	**OmniPhaseSheaf**	**76.58 ± 4.59**	74.77 ± 1.27	75.68 ± 2.21	**80.18 ± 2.55**	71.17 ± 6.74
	OmniPhaseSheaf-T	**73.87 ± 5.55**	**75.68 ± 4.41**	**75.68 ± 2.21**	72.07 ± 4.59	–
	OmniCopreSheaf	74.77 ± 1.27	71.17 ± 3.37	76.58 ± 1.27	**78.38 ± 2.21**	**77.48 ± 4.59**
	OmniCopreSheaf-T	**75.68 ± 4.41**	**75.68 ± 3.82**	**78.38 ± 5.84**	72.97 ± 0.00	–
	OmniDeepSheaf	76.58 ± 6.37	**77.48 ± 2.55**	**77.48 ± 2.55**	72.07 ± 3.37	**79.28 ± 4.59**
	OmniSheafTransformer	**78.38 ± 2.21**	**77.48 ± 2.55**	76.58 ± 1.27	76.58 ± 5.55	–
Wisconsin	**OmniPhaseSheaf**	**86.93 ± 4.89**	86.93 ± 5.62	**86.93 ± 4.89**	81.05 ± 6.06	**87.58 ± 4.03**
	OmniPhaseSheaf-T	**86.93 ± 3.33**	81.70 ± 4.03	81.70 ± 3.33	**83.01 ± 4.89**	–
	OmniCopreSheaf	**84.31 ± 4.24**	81.05 ± 4.89	81.70 ± 0.92	**84.31 ± 5.77**	**86.27 ± 2.77**
	OmniCopreSheaf-T	**84.97 ± 0.92**	81.70 ± 2.45	**83.66 ± 4.03**	81.05 ± 3.33	–
	OmniDeepSheaf	84.31 ± 4.80	82.35 ± 5.77	**84.97 ± 2.45**	**86.27 ± 5.77**	83.01 ± 4.03
	OmniSheafTransformer	**84.31 ± 4.24**	81.05 ± 6.47	83.01 ± 5.62	**83.66 ± 4.03**	–
Amazon-ratings	**OmniPhaseSheaf**	47.74 ± 0.45	**49.26 ± 0.34**	**49.21 ± 1.55**	–	–
	OmniPhaseSheaf-T	**49.17 ± 0.33**	**49.50 ± 0.04**	48.47 ± 0.28	–	–
	OmniCopreSheaf	43.91 ± 0.73	**46.24 ± 0.43**	**46.59 ± 1.21**	–	–
	OmniCopreSheaf-T	47.67 ± 0.53	**48.17 ± 0.47**	**48.63 ± 0.90**	–	–
	OmniDeepSheaf	47.87 ± 0.27	**48.32 ± 0.70**	**51.61 ± 0.73**	–	–
	OmniSheafTransformer	**46.51 ± 0.39**	46.33 ± 0.74	**48.59 ± 0.81**	–	–
Texas	**OmniPhaseSheaf**	**82.88 ± 6.74**	81.08 ± 2.21	**81.98 ± 1.27**	78.38 ± 2.21	80.18 ± 5.55
	OmniPhaseSheaf-T	**78.38 ± 2.21**	77.48 ± 2.55	73.87 ± 4.59	**81.08 ± 3.82**	–
	OmniCopreSheaf	74.77 ± 2.55	79.28 ± 7.09	75.68 ± 2.21	**81.98 ± 3.37**	**80.18 ± 3.37**
	OmniCopreSheaf-T	80.18 ± 3.37	79.28 ± 1.27	**83.78 ± 3.82**	**81.08 ± 4.41**	–
	OmniDeepSheaf	**84.68 ± 5.10**	81.98 ± 5.10	81.98 ± 5.10	**83.78 ± 5.84**	81.98 ± 3.37
	OmniSheafTransformer	**81.98 ± 3.37**	**83.78 ± 4.41**	81.08 ± 5.84	80.18 ± 4.59	–
Citeseer	**OmniPhaseSheaf**	75.97 ± 0.78	**76.41 ± 0.77**	75.08 ± 0.54	76.41 ± 1.05	**77.54 ± 0.69**
	OmniPhaseSheaf-T	75.72 ± 1.03	**76.89 ± 0.63**	**76.81 ± 0.45**	**76.81 ± 0.34**	–
	OmniCopreSheaf	**76.25 ± 0.94**	75.48 ± 1.54	**76.97 ± 1.67**	76.21 ± 1.39	74.36 ± 2.22
	OmniCopreSheaf-T	75.68 ± 1.40	**76.97 ± 0.77**	**76.93 ± 0.52**	**75.93 ± 0.73**	–
	OmniDeepSheaf	**76.89 ± 1.28**	76.77 ± 1.93	76.41 ± 0.28	**77.17 ± 0.62**	**77.62 ± 1.38**
	OmniSheafTransformer	**77.38 ± 0.50**	**76.37 ± 1.22**	76.13 ± 1.25	**76.85 ± 0.28**	–
Cora	**OmniPhaseSheaf**	**85.82 ± 1.71**	84.05 ± 1.99	85.08 ± 1.28	**85.38 ± 1.15**	**85.23 ± 0.96**
	OmniPhaseSheaf-T	84.49 ± 0.36	**86.21 ± 1.50**	**86.56 ± 1.46**	**84.59 ± 0.78**	–
	OmniCopreSheaf	**85.97 ± 1.26**	85.08 ± 1.03	85.33 ± 1.87	**85.52 ± 1.15**	**86.56 ± 1.07**
	OmniCopreSheaf-T	**84.84 ± 0.94**	84.74 ± 1.32	**86.71 ± 0.63**	**85.33 ± 1.19**	–
	OmniDeepSheaf	**86.21 ± 0.97**	86.12 ± 0.55	**86.76 ± 0.54**	**86.95 ± 0.87**	86.16 ± 0.91
	OmniSheafTransformer	**86.31 ± 1.46**	**86.77 ± 0.87**	85.23 ± 0.84	**86.16 ± 1.95**	–

Table 76: Ablation on the hidden width w (hyperedge classification). Test macro-F1 (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given w, all other hyperparameters re-tuned. Colors mark the first, second, and third w per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$w = 32$	$w = 64$	$w = 128$	$w = 256$	$w = 512$
Molecular-1	**OmniPhaseSheaf**	**70.77 ± 0.51**	**72.58 ± 0.52**	**67.05 ± 5.06**	54.35 ± 7.96	–
	OmniPhaseSheaf-T	**90.71 ± 0.10**	**91.09 ± 0.32**	88.51 ± 0.22	**89.62 ± 0.18**	–
	OmniCopreSheaf	**84.78 ± 0.32**	**85.64 ± 0.16**	**86.49 ± 0.30**	84.58 ± 0.63	76.22 ± 5.10
	OmniCopreSheaf-T	**87.67 ± 0.42**	**90.29 ± 0.12**	**88.41 ± 0.66**	86.72 ± 0.16	–
	OmniDeepSheaf	75.48 ± 1.79	**78.19 ± 0.60**	74.66 ± 0.19	**77.83 ± 0.49**	**75.64 ± 1.55**
	OmniSheafTransformer	**81.47 ± 0.18**	**82.09 ± 0.39**	**81.72 ± 0.42**	80.70 ± 1.53	–
Molecular-2	**OmniPhaseSheaf**	**84.28 ± 1.16**	80.99 ± 2.00	83.31 ± 5.85	**85.82 ± 1.40**	**87.54 ± 1.99**
	OmniPhaseSheaf-T	**89.87 ± 1.65**	**89.40 ± 1.25**	89.08 ± 1.89	**90.46 ± 0.76**	–
	OmniCopreSheaf	85.63 ± 1.29	**87.48 ± 1.39**	**86.34 ± 0.86**	82.29 ± 2.71	**87.07 ± 2.56**
	OmniCopreSheaf-T	84.67 ± 2.04	**88.33 ± 2.54**	**84.68 ± 2.57**	**84.75 ± 0.95**	–
	OmniDeepSheaf	86.78 ± 2.11	83.91 ± 1.55	**88.45 ± 2.56**	**89.94 ± 1.64**	**88.91 ± 1.72**
	OmniSheafTransformer	**90.13 ± 3.06**	88.20 ± 1.99	**90.34 ± 2.17**	**90.02 ± 1.62**	–

Table 77: Ablation on the hidden width w (synthetic directed hypergraphs). Test accuracy (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given w, all other hyperparameters re-tuned. Colors mark the first, second, and third w per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$w = 32$	$w = 64$	$w = 128$	$w = 256$	$w = 512$
$I_o = 10$	**OmniPhaseSheaf**	**96.00 ± 2.85**	**96.27 ± 3.02**	**96.00 ± 2.36**	**96.80 ± 1.31**	**96.80 ± 0.65**
	OmniPhaseSheaf-T	**97.07 ± 0.75**	**98.13 ± 1.51**	**97.87 ± 0.75**	92.00 ± 3.92	–
	OmniCopreSheaf	97.07 ± 0.75	**97.60 ± 0.00**	**98.13 ± 1.00**	**97.60 ± 0.65**	**98.13 ± 1.51**
	OmniCopreSheaf-T	**98.40 ± 0.00**	**98.13 ± 1.00**	96.80 ± 0.65	**97.07 ± 2.29**	–
	OmniDeepSheaf	**61.87 ± 12.97**	**72.80 ± 8.84**	**36.80 ± 4.71**	30.40 ± 1.73	31.73 ± 4.35
	OmniSheafTransformer	**88.80 ± 2.99**	**61.33 ± 3.60**	**78.40 ± 5.58**	34.67 ± 4.81	–
$I_o = 30$	**OmniPhaseSheaf**	**98.13 ± 1.00**	**98.67 ± 0.75**	**99.47 ± 0.75**	**99.47 ± 0.38**	**98.67 ± 1.00**
	OmniPhaseSheaf-T	**98.67 ± 0.75**	**99.20 ± 0.65**	**99.47 ± 0.75**	98.13 ± 0.75	–
	OmniCopreSheaf	98.40 ± 1.13	**98.40 ± 1.31**	**99.73 ± 0.38**	**99.73 ± 0.38**	**99.47 ± 0.38**
	OmniCopreSheaf-T	**98.40 ± 1.13**	**99.47 ± 0.38**	**97.60 ± 2.85**	**98.40 ± 1.13**	–
	OmniDeepSheaf	**37.60 ± 3.46**	**54.40 ± 6.53**	**34.93 ± 3.22**	20.27 ± 0.75	21.33 ± 3.83
	OmniSheafTransformer	**63.47 ± 6.34**	**49.07 ± 31.77**	28.80 ± 4.28	**69.60 ± 13.50**	–
$I_o = 50$	**OmniPhaseSheaf**	**99.20 ± 0.65**	**99.20 ± 1.13**	**99.20 ± 0.65**	**98.67 ± 1.36**	**96.53 ± 0.75**
	OmniPhaseSheaf-T	**99.20 ± 1.13**	**99.47 ± 0.75**	94.13 ± 7.17	**99.73 ± 0.38**	–
	OmniCopreSheaf	**100.00 ± 0.00**	98.40 ± 1.73	**99.47 ± 0.75**	98.67 ± 1.00	**98.93 ± 1.00**
	OmniCopreSheaf-T	**99.47 ± 0.75**	**99.73 ± 0.38**	**99.20 ± 1.13**	98.67 ± 1.36	–
	OmniDeepSheaf	**35.47 ± 2.29**	**33.60 ± 2.99**	**38.93 ± 7.57**	26.93 ± 7.43	27.47 ± 7.98
	OmniSheafTransformer	**86.67 ± 5.89**	**25.07 ± 2.29**	**29.60 ± 5.58**	23.20 ± 5.58	–

Table 78: Ablation on the hidden width w (undirected hypergraphs). Test accuracy (%) $\pm$ standard deviation of the best configuration (selected on validation) with the given w, all other hyperparameters re-tuned. Colors mark the first, second, and third w per row; – denotes a value outside the search space of that dataset (Appendix D.2) or without a completed trial.

Dataset	Model	$w = 32$	$w = 64$	$w = 128$	$w = 256$	$w = 512$
Cora	**OmniDeepSheaf**	76.22 ± 1.82	**78.58 ± 1.47**	**78.53 ± 0.46**	78.04 ± 1.64	**78.73 ± 1.15**
	OmniSheafTransformer	**79.37 ± 1.55**	78.09 ± 0.25	**79.17 ± 1.54**	**79.17 ± 1.28**	–
Citeseer	**OmniDeepSheaf**	**74.48 ± 0.70**	73.67 ± 1.24	**74.64 ± 1.71**	**74.32 ± 0.89**	74.15 ± 0.90
	OmniSheafTransformer	**74.52 ± 0.65**	72.10 ± 1.59	**72.46 ± 0.68**	**72.18 ± 0.92**	–
Pubmed	**OmniDeepSheaf**	**87.81 ± 0.17**	**87.48 ± 0.14**	87.30 ± 0.49	**87.86 ± 0.13**	–
	OmniSheafTransformer	**87.82 ± 0.29**	87.48 ± 0.24	**87.61 ± 0.15**	**87.50 ± 0.38**	–
Cora-CA	**OmniDeepSheaf**	80.50 ± 0.84	**82.82 ± 0.97**	**83.26 ± 0.62**	**82.42 ± 1.87**	82.27 ± 0.53
	OmniSheafTransformer	**83.16 ± 1.03**	81.09 ± 0.96	**83.01 ± 0.91**	**83.06 ± 0.80**	–
DBLP-CA	**OmniDeepSheaf**	**91.56 ± 0.11**	**91.51 ± 0.14**	**91.64 ± 0.16**	–	–
	OmniSheafTransformer	**91.61 ± 0.22**	**91.41 ± 0.20**	**91.62 ± 0.27**	–	–
Senate	**OmniDeepSheaf**	**80.28 ± 1.99**	77.93 ± 2.89	**78.87 ± 1.15**	78.40 ± 2.39	**80.28 ± 2.30**
	OmniSheafTransformer	75.59 ± 3.70	**81.22 ± 1.33**	**78.87 ± 2.30**	**75.59 ± 3.51**	–
House	**OmniDeepSheaf**	77.19 ± 0.29	**77.61 ± 0.29**	**77.50 ± 1.48**	77.30 ± 0.96	**77.71 ± 1.77**
	OmniSheafTransformer	**77.50 ± 0.39**	**76.88 ± 1.54**	**77.40 ± 1.10**	**76.88 ± 1.02**	–
Congress	**OmniDeepSheaf**	**93.95 ± 1.14**	**94.73 ± 0.58**	–	–	–
	OmniSheafTransformer	**95.04 ± 0.29**	**95.19 ± 0.77**	–	–	–