OMNISHEAF: UNIFYING SHEAF HYPERGRAPH NETWORKS VIA EFFICIENT UNIVERSAL APPROXIMATORS

Anonymous authors
Paper under double-blind review

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

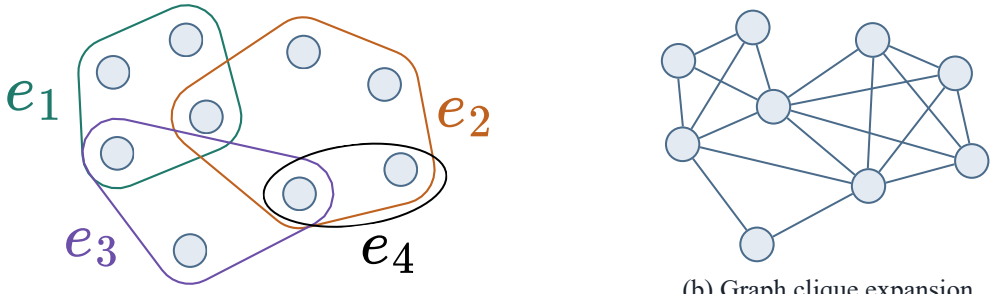
Hypergraphs extend graphs to higher-order relations and enable nodes to be connected via hyperedges, i.e., a set-type relation between multiple nodes. Hypergraphs have several applications in numerous fields such as computer networking (Liu et al., 2024), neuroscience (Guo et al., 2021) and radio allocation (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 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

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

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.

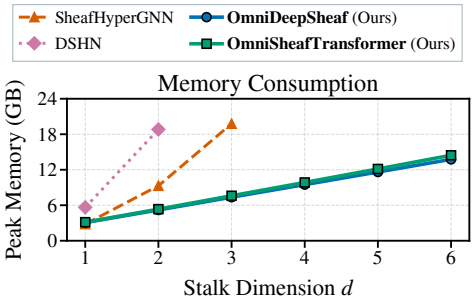


Figure 2: *Cost on Roman-empire as the stalk dimension d grows.* OS grows linearly in d in peak training memory, while DSHN and SheafHyperGNN grow quadratically and exhaust the 24 GB card at $d = 3$ and $d = 4$.

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

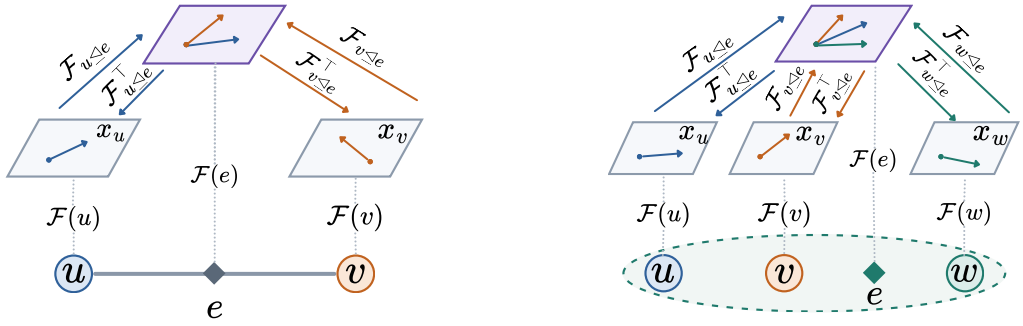
Graph Neural Networks. GNNs (Gori et al., 2005; Scarselli et al., 2009) are the most common framework for modeling pairwise interactions. They operate by propagating messages along edges (Kipf & Welling, 2017; Veličković et al., 2018). However they are fundamentally limited to binary connections and 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 $\tilde{\mathcal{F}}_{u \trianglelefteq e}^\top \tilde{\boldsymbol{H}}_e = \boldsymbol{Q}_u (\mathcal{F}_{u \trianglelefteq e}^\top \boldsymbol{H}_e)$. 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:*

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

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

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

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:

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

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.

Method	Directed Node Classification						Hyperedge Classification	
	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 $\pm$ 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	$83.10^{\dagger} \pm 1.99$	$78.33^{\dagger} \pm 1.26$	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	$81.69^{\dagger} \pm 2.30$	$78.43^{\dagger} \pm 1.17$	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 $7.1\times$ faster training and $3.6\times$ 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.

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.