

## Appendix for “High-Pass Matters: Theoretical Insights and Sheaflet-Based Design for Hypergraph Neural Networks”

Ming Li<sup>1</sup>, Yujie Fang<sup>2</sup>, Dongrui Shen<sup>3</sup>, Han Feng<sup>3</sup>, Xiaosheng Zhuang<sup>3</sup>, Kelin Xia<sup>4</sup>, Pietro Lio<sup>5</sup>

<sup>1</sup>Zhejiang Key Laboratory of Intelligent Education Technology and Application, Zhejiang Normal University, Jinhua, China

<sup>2</sup>School of Computer Science and Technology, Zhejiang Normal University, Jinhua, China

<sup>3</sup>Department of Mathematics, City University of Hong Kong, Hong Kong, China

<sup>4</sup>School of Physical & Mathematical Sciences, Nanyang Technological University, Singapore

<sup>5</sup>Department of Computer Science and Technology, University of Cambridge, UK

### A Further Details on Related Work

**Hypergraph Learning.** Hypergraph neural networks (HGNNs) have gained increasing attention for their ability to model higher-order relationships beyond pairwise interactions. One of the earliest methods, HGNN (Feng et al., 2019), extends graph convolutional networks to hypergraphs via spectral methods and effectively captures hyperedge-level dependencies. HGNN<sup>+</sup> (Gao et al., 2022) enhances this framework by introducing spatial hypergraph convolution and hyperedge group fusion to model multi-modal or multi-type interactions. HyperGCN (Yadati et al., 2019) adopts a different approach by transforming hyperedges into pairwise edges via mediators and applying graph convolutions, thereby simplifying hypergraph structures for computation. More recently, HyperND (Prokopchik, Benson, and Tudisco, 2022) proposes a nonlinear diffusion formulation for label propagation over hypergraphs, achieving both improved accuracy and theoretical convergence guarantees. ED-HNN (Wang et al., 2023) formulates hypergraph learning through equivariant message passing using a star expansion, and establishes universality by approximating any continuous equivariant hypergraph diffusion operator. HDS<sup>ODE</sup> (Yan et al., 2024) leverages neural ODEs to construct continuous-time hypergraph dynamic systems, enhancing model depth and controllability. Additionally, Transformer-based architectures have been introduced, such as HyperGT (Liu et al., 2024), which incorporates structural priors into Transformer backbones using hypergraph incidence-based positional encodings and regularizations. To address structural bottlenecks, SensHNN (Yadati, 2024) studies oversquashing in HGNNs and proposes new sensitivity-based benchmarks. HyperSheaflets (Li et al., 2025) is among the few models that utilize both low-pass and high-pass components through tight frame transforms, showing improved performance by mitigating oversmoothing, yet it lacks theoretical analysis on frequency behavior. Overall, while existing methods offer various architectural innovations, most either prioritize message passing or structural heuristics and rarely analyze the spectral behavior of frequency components in depth.

**Sheaf Theory for Graphs and Hypergraphs.** Sheaf theory has recently emerged as a powerful framework for generalizing graph-based learning via local data consistency. SheafNN (Hansen and Gebhart, 2020) replaces the classical graph Laplacian with the sheaf Laplacian (Hansen and Ghrist, 2019), leading to more expressive diffusion processes. Several extensions further improve this framework by integrating attention (Barbero et al., 2022), positional encodings (He, Bodnar, and Lio, 2023), and task-specific adaptations such as personalized federated learning (Nguyen et al., 2024) and recommendation (Purificato et al., 2023). SheafHyperGNN (Duta et al., 2024) introduces cellular sheaf constructions on hypergraphs and designs sheaf-guided diffusion, enabling more expressive modeling of higher-order relational structures. Meanwhile, recent work in algebraic topology extends the applicability of sheaf cohomology, such as the poset-based sheaf cohomology algorithm in (Ayzenberg et al., 2025). However, most existing sheaf-based methods operate on single-frequency diffusion and do not exploit multi-frequency filtering or spectral decomposition, leaving a gap in integrating sheaves with signal processing perspectives.

**Multiscale Graph Signal Processing.** Theoretical foundations for multiscale analysis on graphs trace back

to early works on wavelets and framelets. A universal multiscale construction is introduced in (Maggioni and Mhaskar, 2008), while spectral graph wavelets are formulated in (Hammond, Vandergheynst, and Gribonval, 2011) via Laplacian eigenbasis with polynomial approximations to improve efficiency. These ideas are brought into learning frameworks in (Zheng et al., 2021), which proposes framelet-based GNNs for localized, multiscale convolution and pooling. Further advances include QuasiSpectral (Yang et al., 2025), which directly designs spectral filtering functions for robustness and control, and permutation-equivariant framelet designs (Li et al., 2024), which extend framelets to structural GNNs. Despite these efforts, most prior works either focus on graphs or lack integration with sheaf theory and hypergraph structures.

**Position of Our Work.** Although there is growing interest in combining sheaf theory and framelet analysis, most prior work is limited to graphs. Chen et al. (2023) propose sheaflet constructions that fuse cellular sheaves with framelets for multiscale graph learning, but their method is restricted to pairwise graphs and lacks application to hypergraphs. Moreover, none of these methods offer theoretical insights into how low- and high-frequency components contribute to representation power. In contrast, our work introduces a unified spectral framework, HyperSheaflets, that integrates sheaves with framelet-based multiscale analysis on hypergraphs. We provide theoretical justification for the importance of high-pass components and demonstrate empirically that our design captures richer structural information. This not only fills a key gap in the literature but also opens new directions for principled, multi-frequency hypergraph learning with sheaf structures.

## B Notation Summary and Additional Proofs

Table 1: Summary of notations frequently used throughout the paper.

Symbol	Description
$\mathcal{H} = (\mathcal{V}, \mathcal{E})$	Hypergraph with node set $\mathcal{V}$ and hyperedge set $\mathcal{E}$
$N =  \mathcal{V} $	Number of nodes in the hypergraph
$M =  \mathcal{E} $	Number of hyperedges in the hypergraph
$\mathbf{X} \in \mathbb{R}^{N \times m}$	Input feature matrix for nodes
$\mathbf{H} \in \{0, 1\}^{N \times M}$	Hypergraph incidence matrix
$\delta_e$	Degree of hyperedge $e$ (i.e., number of nodes in $e$ )
$w_r(\cdot, t)$	the $r$ th order modulus of smoothness for $L^2(\mathcal{H})$ with scale $t > 0$
$\mathcal{F}$	Cellular sheaf associated with hypergraph $\mathcal{H}$
$\mathcal{F}(v)$	Vertex stalk: vector space associated with node $v$
$\mathcal{F}(e)$	Hyperedge stalk: vector space associated with hyperedge $e$
$\mathcal{F}_{v \leq e}$	Restriction map from node $v$ to hyperedge $e$
$d$	Dimension of the stalks (assumed the same across all nodes/edges)
$\mathcal{L}_{\mathcal{F}}$	Linear sheaf Laplacian operator
$x \in \mathbb{R}^{N \times d}$	Sheaf signal on nodes
$\lambda_\ell$	$\ell$ -th eigenvalue of $\mathcal{L}_{\mathcal{F}}$
$\mathbf{u}_\ell$	$\ell$ -th eigenvector of $\mathcal{L}_{\mathcal{F}}$
$\mathbf{U}$	Eigenvector matrix of $\mathcal{L}_{\mathcal{F}}$
$\mathbf{\Lambda}$	Diagonal matrix of eigenvalues
$\phi_{j,p}(v)$	Sheaflet scaling function at scale $j$ and location $p$
$\psi_{j,p}^r(v)$	Sheaflet wavelet function of channel $r$ at scale $j$ , location $p$
$\alpha, \beta^{(r)}$	Scaling and wavelet functions in the spectral domain
$\hat{\alpha}(\cdot), \hat{\beta}^{(r)}(\cdot)$	Fourier transforms of $\alpha$ and $\beta^{(r)}$
$\mathcal{T}_n$	Chebyshev polynomial of degree $n$
$n$	Number of high-pass channels in the filter bank
$\mathcal{W}_{0,J}$	Low-pass decomposition operator at coarsest scale $J$
$\mathcal{W}_{r,j}$	High-pass decomposition operator for channel $r$ at scale $j$
$V_0$	Sheaflet coefficients at low-pass scale (coarsest level)
$W_j^r$	Sheaflet coefficients for high-pass component $r$ at scale $j$
$\tilde{\mathbf{X}}^{(\ell)}$	Feature matrix at the $\ell$ -th layer of the HNN
$\Theta_{0,J}$	Learnable spectral filter coefficient (diagonal) for low-pass component
$\Theta_{r,j}$	Learnable spectral filter coefficient (diagonal) for high-pass component $r$ at scale $j$
$W_{0,J}$	Learnable transformation weight for low-pass frequency response
$W_{r,j}$	Learnable transformation weight for high-pass response (channel $r$ , scale $j$ )
$\sigma(\cdot)$	Nonlinear activation function, e.g., ReLU
$K$	Coarsest scale level in sheaflet construction
$J$	Finest scale level in sheaflet decomposition

**Theorem 3.4.** In the setting of Theorem 3.3, let  $s(z) = \log p(x, z)$  be the log-likelihood function, and define the low-pass and high-pass estimates:

$$\begin{aligned} z_L &= \mathcal{W}_{0,J}^\top \Theta_{0,J} \mathcal{W}_{0,J} x, \\ z_H &= \sum_{(r,j) \neq (0,J)} \mathcal{W}_{r,j}^\top \Theta_{r,j} \mathcal{W}_{r,j} x. \end{aligned}$$

Let  $\lambda_{\min}$  denote the smallest non-zero eigenvalue of the hypergraph Laplacian. Suppose the low-pass filter  $a_0$  satisfies  $\hat{a}_0(0) = 1$ ,  $a = \prod_{j=0}^{J-1} |\hat{a}_0(2^{-S+j} \lambda_{\min})|^2 < \frac{1}{2}$ . If the high-frequency components of  $x$  dominate in the sense that:

$$(1 - 2a) \left[ \sum_{\lambda_k \geq \lambda_{\min}} |\hat{x}_k|^2 \right]^{\frac{1}{2}} \geq \left[ \sum_{\lambda_k < \lambda_{\min}} |\hat{x}_k|^2 \right]^{\frac{1}{2}} + \sum_{(r,j) \in \Gamma} \|I - \Theta_{r,j}\|_2 \|\hat{x}\|_2 + \sqrt{2}\sigma\varepsilon,$$

where

$$\varepsilon = \max \left\{ \sum_{(r,j) \in \Gamma} [g(\mathcal{W}_{r,j} z_H; \gamma_{r,j}) - g(\mathcal{W}_{r,j} z_L; \gamma_{r,j})], 0 \right\}^{\frac{1}{2}}.$$

Then we have that  $s(z_L) \leq s(z_H)$ .

*Proof.* For each framelet decomposition operator  $\mathcal{W}_{r,j}$ , we denote the corresponding scaling functions in the spectral domain by  $\hat{h}_{r,j}(\xi)$ . That is,  $\widehat{\mathcal{W}_{r,j} x} = \hat{h}_{r,j}(\Lambda) \hat{x}$  and

$$\begin{aligned} \hat{h}_{r,1}(\xi) &= \hat{a}_r^*(2^{-S+j-1}\xi), \\ \hat{h}_{r,j}(\xi) &= \hat{a}_r^*(2^{-S+j-1}\xi) \prod_{l=0}^{j-2} \hat{a}_0^*(2^{-S+l}\xi) \quad (2 \leq j \leq J). \end{aligned}$$

Recall that

$$s(z) = -\frac{1}{2\sigma^2} \|x - z\|_2^2 - \sum_{(r,j) \in \Gamma} g(\mathcal{W}_{r,j} z; \gamma_{r,j}).$$

We will prove the theorem by showing that:

$$\frac{1}{2\sigma^2} \|x - z_L\|_2^2 + \sum_{(r,j) \in \Gamma} g(\mathcal{W}_{r,j} z_L; \gamma_{r,j}) \geq \frac{1}{2\sigma^2} \|x - z_H\|_2^2 + \sum_{(r,j) \in \Gamma} g(\mathcal{W}_{r,j} z_H; \gamma_{r,j}).$$

To begin with, we analyze the reconstruction error for the low-pass and high-pass estimates, respectively.

#### 1. Low-pass error

$$\begin{aligned} \|x - z_L\|_2 &\geq \|x - \mathcal{W}_{0,J}^\top \mathcal{W}_{0,J} x\|_2 - \|\mathcal{W}_{0,J}^\top (I - \text{diag}(\theta_{0,J})) \mathcal{W}_{0,J} x\|_2 \\ &= \left[ \sum_k |\hat{x}_k|^2 \left(1 - \hat{h}_{0,J}(\lambda_k)^2\right)^2 \right]^{\frac{1}{2}} - \left\| \hat{h}_{0,J}(\Lambda) U^\top (I - \Theta_{0,J}) U \hat{h}_{0,J}(\Lambda) \hat{x} \right\|_2 \\ &\geq \left(1 - \hat{h}_{0,J}(\lambda_{\min})^2\right) \left[ \sum_{\lambda_k \geq \lambda_{\min}} |\hat{x}_k|^2 \right]^{\frac{1}{2}} - \|I - \Theta_{0,J}\|_2 \|\hat{x}\|_2 \end{aligned}$$

## 2. High-pass error

$$\begin{aligned}
\|x - z_H\|_2 &\leq \left\| x - \sum_{(r,j) \neq (0,J)} \mathcal{W}_{r,j}^\top \mathcal{W}_{r,j} x \right\|_2 + \sum_{(r,j) \neq (0,J)} \|\mathcal{W}_{r,j}^\top (I - \Theta_{r,j}) \mathcal{W}_{r,j} x\|_2 \\
&= \left[ \sum_k \left( \hat{h}_{0,J}(\lambda_k) |\hat{x}_k| \right)^2 \right]^{\frac{1}{2}} + \sum_{(r,j) \neq (0,J)} \left\| U \hat{h}_{r,j}(\Lambda) U^\top (I - \Theta_{r,j}) U \hat{h}_{r,j}(\Lambda) \hat{x} \right\|_2 \\
&\leq \left[ \sum_{\lambda_k < \lambda_{\min}} |\hat{x}_k|^2 \right]^{\frac{1}{2}} + \hat{h}_{0,J}(\lambda_{\min})^2 \left[ \sum_{\lambda_k \geq \lambda_{\min}} |\hat{x}_k|^2 \right]^{\frac{1}{2}} + \sum_{(r,j) \neq (0,J)} \|I - \Theta_{r,j}\|_2 \|\hat{x}\|_2
\end{aligned}$$

Define  $a = \hat{h}_{0,J}(\lambda_{\min})^2$ . Recall that

$$(1 - 2a) \left[ \sum_{\lambda_k \geq \lambda_{\min}} |\hat{x}_k|^2 \right]^{\frac{1}{2}} \geq \left[ \sum_{\lambda_k < \lambda_{\min}} |\hat{x}_k|^2 \right]^{\frac{1}{2}} + \sum_{(r,j) \in \Gamma} \|I - \Theta_{r,j}\|_2 \|\hat{x}\|_2 + \sqrt{2}\sigma\varepsilon,$$

This leads to

$$\begin{aligned}
\frac{1}{2\sigma^2} \|x - z_L\|_2^2 &\geq \frac{1}{2\sigma^2} \|x - z_H\|_2^2 + \varepsilon^2 \\
&\geq \frac{1}{2\sigma^2} \|x - z_H\|_2^2 + \sum_{(r,j) \in \Gamma} g(\mathcal{W}_{r,j} z_H; \gamma_{r,j}) - \sum_{(r,j) \in \Gamma} g(\mathcal{W}_{r,j} z_L; \gamma_{r,j}),
\end{aligned}$$

which completes the proof.  $\square$

## C Additional Details on Experimental Studies

Table 2 summarizes the detailed statistics of the benchmark datasets used in our experimental studies.

Table 2: Statistics of the datasets

Datasets	Nodes	Edges	Classes	Features	$\mathcal{H}_{\text{node}}$	$\mathcal{H}_{\text{edge}}$
Cora	2708	1579	7	1433	0.640	0.746
Citeseer	3312	1709	6	3703	0.577	0.681
Pubmed	19717	7963	3	500	0.550	0.777
Cora-CA	2708	1072	7	1433	0.728	0.780
DBLP-CA	41302	22363	6	1425	0.856	0.866
Congress	1718	83105	2	100	0.659	0.651
Senate	282	315	2	100	0.479	0.464
House	1290	340	2	100	0.505	0.485
Actor	16255	10164	3	50	0.482	0.468
Amazon-ratings	22299	2090	5	111	0.481	0.368
Twitch-gamers	16812	2627	2	7	0.489	0.486
Pokec	14998	2406	2	65	0.495	0.453

Table 3 summarizes the hyperparameter settings used for the 12 benchmark datasets in our experiments.

To illustrate the structure of the analytically constructed multi-scale decomposition operators, we visualize the sheaflet transform matrices  $\mathcal{W}_{r,j}$  using heatmaps, as shown in Figure 1. The top row displays the low-pass operators  $\mathcal{W}_{0,j}$  from Level 1 to Level 3 (left to right), while the bottom row shows the corresponding high-pass operators  $\mathcal{W}_{1,j}$ . These matrices are derived from Chebyshev polynomial approximations of spectral filters at different scales. As observed, the low-pass operators exhibit strong diagonal dominance, indicating the

Table 3: Hyperparameter settings for the 12 benchmark datasets used in our experiments.

Dataset	Hyperparameter Setting	
<b>Cora</b>	Learning rate: 0.019 Weight decay: 1e-6 Hidden Size: 64 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: sigmoid Level: 1 Seed: 30
<b>Citeseer</b>	Learning rate: 0.003 Weight decay: 1e-5 Hidden Size: 64 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: tanh Level: 1 Seed: 0
<b>Pubmed</b>	Learning rate: 0.004 Weight decay: 3e-4 Hidden Size: 256 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: none Level: 1 Seed: 500
<b>Cora-CA</b>	Learning rate: 0.004 Weight decay: 1e-7 Hidden Size: 0 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: sigmoid Level: 1 Seed: 5000
<b>DBLP-CA</b>	Learning rate: 0.009 Weight decay: 3e-6 Hidden Size: 0 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: sigmoid Level: 1 Seed: 1000
<b>Congress</b>	Learning rate: 0.016 Weight decay: 1e-5 Hidden Size: 256 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: sigmoid Level: 1 Seed: 50
<b>Senate</b>	Learning rate: 0.02 Weight decay: 6e-6 Hidden Size: 512 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: tanh Level: 1 Seed: 2000
<b>House</b>	Learning rate: 0.009 Weight decay: 5e-2 Hidden Size: 512 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: sigmoid Level: 1 Seed: 3000
<b>Actor</b>	Learning rate: 0.002 Weight decay: 0 Hidden Size: 0 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: none Level: 1 Seed: 0
<b>Amazon-ratings</b>	Learning rate: 0.001 Weight decay: 0 Hidden Size: 0 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: none Level: 1 Seed: 0
<b>Twitch-gamers</b>	Learning rate: 0.005 Weight decay: 1e-5 Hidden Size: 512 Dropout ratio: 0.9	Normalization: sym_degree_norm Non-linearity: sigmoid Level: 1 Seed: 0
<b>Pokec</b>	Learning rate: 0.007 Weight decay: 9e-5 Hidden Size: 0 Dropout ratio: 0.9	Normalization: degree_norm Non-linearity: none Level: 1 Seed: 2000

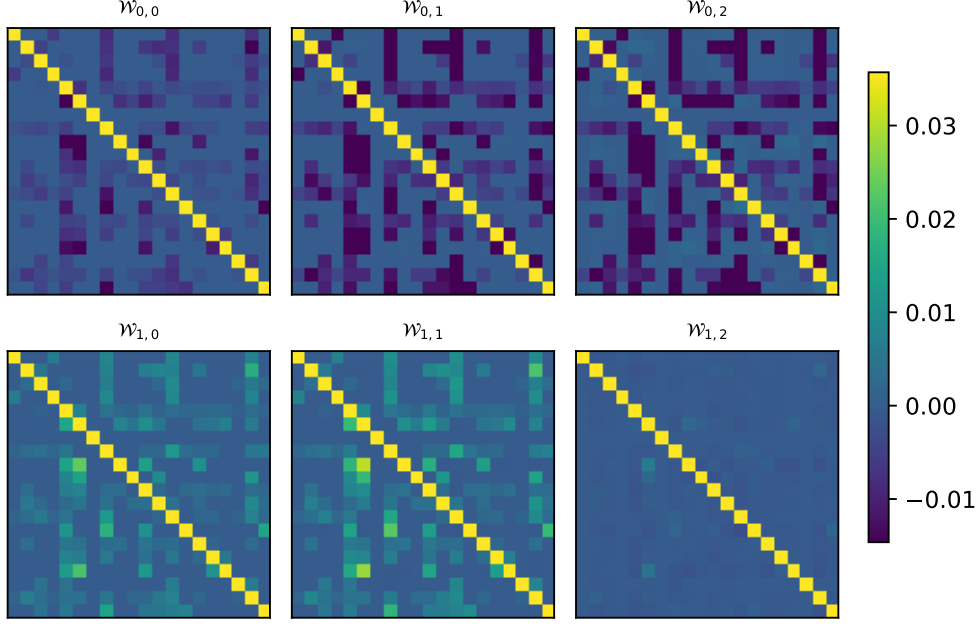


Figure 1: Framelet transform matrices  $W_{r,j}$  from Level = 1 (LHS) to Level = 3 (RHS)

smoothing effect over the hypergraph, whereas the high-pass operators capture more localized and oscillatory components, reflecting structural variations. This visualization provides insight into how HyperSheaflets perform multi-resolution filtering across frequency bands, consistent with the model’s theoretical formulation.

## D Computational Complexity Analysis

This section presents the training-time computational complexity of four representative hypergraph neural networks and the proposed HyperSheaflets model. Let  $N$  and  $M$  denote the numbers of nodes and hyperedges in the given hypergraph, respectively, and  $M'$  denote the number of edges in its clique expansion. The symbol  $\|\mathbf{H}\|_0$  represents the number of nonzero entries in the incidence matrix  $\mathbf{H}$ . We use  $T$  for the number of training epochs,  $L$  for the number of layers, and  $d$  for the feature dimension. In HyperSheaflets,  $n$  denotes the number of high-pass filters,  $J$  the number of scales, and  $K$  the maximal number of nonzero elements in the framelet transform matrices  $W_{r,j}$  (see Section 4 for details).

The parameters  $n$ ,  $J$ , and  $K$  are constants independent of the input hypergraph. In practice, both  $n$  and  $J$  take small values, while the sparsity of the constructed hypergraph framelets ensures that  $K$  is typically moderate and often comparable to or smaller than  $\|\mathbf{H}\|_0$ . Consequently, HyperSheaflets achieves competitive efficiency without introducing additional computational burden compared with existing methods. Its overall complexity is approximately on par with AllDeepSets (Chien et al., 2022) and ED-HNN (Wang et al., 2023).

Table 4: Summary of computational complexity for UniGCNII, Deep-HGNN, AllDeepSets, ED-HNN, and our proposed HyperSheaflets model.

Name	Computational Complexity
UniGCNII (Huang and Yang, 2021)	$\mathcal{O}\left(TL(N + M + \ \mathbf{H}\ _0)d + TLNd^2\right)$
Deep-HGCN (Chen et al., 2022)	$\mathcal{O}\left(TLM'd + TLNd^2\right)$
AllDeepSets (Chien et al., 2022)	$\mathcal{O}\left(TL\ \mathbf{H}\ _0d + TL(N + M)d^2\right)$
ED-HNN (Wang et al., 2023)	$\mathcal{O}\left(TL\ \mathbf{H}\ _0d + TL(N + M)d^2\right)$
HyperSheaflets	$\mathcal{O}\left(TL(nJ + 1)Kd + TL(N + M)d^2\right)$

## References

- Ayzenberg, A.; Gebhart, T.; Magai, G.; and Solomadin, G. 2025. Sheaf theory: from deep geometry to deep learning. *arXiv preprint arXiv:2502.15476*.
- Barbero, F.; Bodnar, C.; de Ocáriz Borde, H. S.; and Lio, P. 2022. Sheaf attention networks. In *NeurIPS 2022 Workshop on Symmetry and Geometry in Neural Representations*.
- Chen, G.; Zhang, J.; Xiao, X.; and Li, Y. 2022. Preventing over-smoothing for hypergraph neural networks. *arXiv preprint arXiv:2203.17159*.
- Chen, J.; Wang, Y.; Bodnar, C.; Ying, R.; Lio, P.; and Wang, Y. G. 2023. Dirichlet energy enhancement of graph neural networks by framelet augmentation. *arXiv preprint arXiv:2311.05767*.
- Chien, E.; Pan, C.; Peng, J.; and Milenkovic, O. 2022. You are AllSet: a multiset function framework for hypergraph neural networks. In *ICLR*.
- Duta, I.; Cassarà, G.; Silvestri, F.; and Liò, P. 2024. Sheaf hypergraph networks. In *NeurIPS*, 36: 12087–12099.
- Feng, Y.; You, H.; Zhang, Z.; Ji, R.; and Gao, Y. 2019. Hypergraph neural networks. In *AAAI*, 3558–3565.
- Gao, Y.; Feng, Y.; Ji, S.; and Ji, R. 2022. HGNN+: General hypergraph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3): 3181–3199.
- Hammond, D. K.; Vandergheynst, P.; and Gribonval, R. 2011. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2): 129–150.
- Hansen, J.; and Gebhart, T. 2020. Sheaf Neural Networks. In *TDA & Beyond*.
- Hansen, J.; and Ghrist, R. 2019. Toward a spectral theory of cellular sheaves. *Journal of Applied and Computational Topology*, 3(4): 315–358.
- He, Y.; Bodnar, C.; and Lio, P. 2023. Sheaf-based positional encodings for graph neural networks. In *NeurIPS 2023 Workshop on Symmetry and Geometry in Neural Representations*, volume 9.
- Huang, J.; and Yang, J. 2021. UniGNN: a unified framework for graph and hypergraph neural networks. In *IJCAI*, 2563–2569.
- Li, J.; Zheng, R.; Feng, H.; Li, M.; and Zhuang, X. 2024. Permutation equivariant graph framelets for heterophilous graph learning. *IEEE Transactions on Neural Networks and Learning Systems*, 35(9): 11634–11648.
- Li, M.; Fang, Y.; Wang, Y.; Feng, H.; Gu, Y.; Bai, L.; and Lio, P. 2025. Deep hypergraph neural networks with tight framelets. In *AAAI*, 18385–18392.
- Liu, Z.; Tang, B.; Ye, Z.; Dong, X.; Chen, S.; and Wang, Y. 2024. Hypergraph transformer for semi-supervised classification. In *ICASSP*, 7515–7519.
- Maggioni, M.; and Mhaskar, H. 2008. Diffusion polynomial frames on metric measure spaces. *Applied and Computational Harmonic Analysis*, 24(3): 329–353.
- Nguyen, B.; Sani, L.; Qiu, X.; Liò, P.; and Lane, N. D. 2024. Sheaf hypernetworks for personalized federated learning. *arXiv preprint arXiv:2405.20882*.
- Prokopchik, K.; Benson, A. R.; and Tudisco, F. 2022. Nonlinear feature diffusion on hypergraphs. In *ICML*, 17945–17958.
- Purificato, A.; Cassarà, G.; Liò, P.; and Silvestri, F. 2023. Sheaf neural networks for graph-based recommender systems. *arXiv preprint arXiv:2304.09097*.
- Wang, P.; Yang, S.; Liu, Y.; Wang, Z.; and Li, P. 2023. Equivariant hypergraph diffusion neural operators. In *ICLR*.
- Yadati, N. 2024. Oversquashing in Hypergraph Neural Networks. In *The Third Learning on Graphs Conference*.

Yadati, N.; Nimishakavi, M.; Yadav, P.; Nitin, V.; Louis, A.; and Talukdar, P. 2019. HyperGCN: A new method for training graph convolutional networks on hypergraphs. In *NeurIPS*, 1511–1522.

Yan, J.; Feng, Y.; Ying, S.; and Gao, Y. 2024. Hypergraph dynamic system. In *ICLR*.

Yang, M.; Shi, D.; Zheng, X.; Yin, J.; and Gao, J. 2025. Quasi-framelets: robust graph neural networks via adaptive framelet convolution. *International Journal of Machine Learning and Cybernetics*, 16(2): 755–770.

Zheng, X.; Zhou, B.; Gao, J.; Wang, Y. G.; Lió, P.; Li, M.; and Montúfar, G. 2021. How framelets enhance graph neural networks. In *ICML*, 12761–12771.