

# Sheaf Graph Neural Networks via PAC-Bayes Spectral Optimization

Yoonhyuk Choi<sup>1</sup>, Jiho Choi<sup>2</sup>, Taewook Ko<sup>3</sup>  
JongWook Kim<sup>4\*</sup>, Chong-Kwon Kim<sup>5\*</sup>

<sup>1</sup>Sookmyung Women’s University, Seoul, Republic of Korea

<sup>2</sup>Korea Advanced Institute of Science and Technology (KAIST), Seoul, Republic of Korea

<sup>3</sup>Samsung Electronics, Seoul, Republic of Korea

<sup>4</sup>Sangmyung University, Seoul, Republic of Korea

<sup>5</sup>Korea Institute of Energy Technology (KENTECH), Naju, Republic of Korea

chldbsgur123@sookmyung.ac.kr, jihochoi1993@gmail.com, taewook.ko@snu.ac.kr, jkim@smu.ac.kr, ckim@kentech.ac.kr

## Abstract

Over-smoothing in Graph Neural Networks (GNNs) causes collapse in distinct node features, particularly on heterophilic graphs where adjacent nodes often have dissimilar labels. Although sheaf neural networks partially mitigate this problem, they typically rely on static or heavily parameterized sheaf structures that hinder generalization and scalability. Existing sheaf-based models either predefine restriction maps or introduce excessive complexity, yet fail to provide rigorous stability guarantees. In this paper, we introduce a novel scheme called SGPC (Sheaf GNNs with PAC-Bayes Calibration), a unified architecture that combines cellular-sheaf message passing with several mechanisms, including optimal transport-based lifting, variance-reduced diffusion, and PAC-Bayes spectral regularization for robust semi-supervised node classification. We establish performance bounds theoretically and demonstrate that end-to-end training in linear computational complexity can achieve the resulting bound-aware objective. Experiments on nine homophilic and heterophilic benchmarks show that SGPC outperforms state-of-the-art spectral and sheaf-based GNNs while providing certified confidence intervals on unseen nodes.

## Introduction

The explosive growth of graph-structured data across social (Fan et al. 2019), biological (Zhang et al. 2021), and industrial domains (Chen et al. 2021) has established Graph Neural Networks (GNNs) as a cornerstone of modern machine learning. Classic message passing GNNs (Kipf and Welling 2016; Velickovic et al. 2017; Defferrard, Bresson, and Vandergheynst 2016) aggregate neighbor signals under an implicit homophily assumption, where adjacent nodes tend to share labels or attributes. The resulting low-pass filters perform Laplacian smoothing (Li et al. 2022), which is effective on homophilic graphs but provably degrades under heterophily or adversarial structure (Pei et al. 2020; Zhu et al. 2020). While recent spatial remedies like edge re-weighting (Choi et al. 2025), subgraph sampling (Bo et al. 2021), and attention mechanisms (Brody, Alon, and Yahav 2021) yield empirical gains, they treat edges as scalar weights and largely ignore uncertainty.

\*Corresponding authors.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

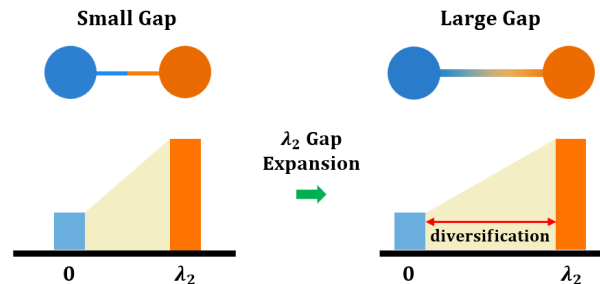


Figure 1: In the small-gap regime (left), two nodes are connected by a weak edge, so the Laplacian spectrum shows only a narrow separation between the first eigenvalue (0) and the second one ( $\lambda_2$ ). After  $\lambda_2$ -gap expansion (right), the edge becomes strong and smoothly color-graded with a wide spectrum, illustrating the enlarged spectral gap

Cellular sheaf theory reinterprets an edge as a linear restriction between local feature spaces, inducing a sheaf Laplacian whose spectrum captures edge directionality and class dispersion (Hansen and Gebhart 2020; Bodnar et al. 2022). Previous studies showed that matrix edge representation can suppress over-smoothing while respecting feature anisotropy. However, existing sheaf GNNs suffer from several limitations: they (i) fix restriction maps via simple gating mechanisms, (ii) lack heterophily-aware uncertainty calibration, and (iii) offer no generalization guarantees beyond empirical test accuracy (Zaghen et al. 2024). These gaps hinder widespread adoption across various domains, highlighting the need for calibrated risk and theoretical robustness.

As one solution, PAC-Bayes analysis offers a principled route by linking expected risk to posterior-prior compression and data-dependent margins (Zhou et al. 2018; Letarte et al. 2019). Yet current PAC-Bayes bounds for GNNs remain loose because they ignore the spectrum of the underlying operator (e.g., the sheaf or graph Laplacian), which governs diffusion depth and representational expressivity (Xu et al. 2018). A tighter, spectrum-aware bound can directly translate spectral engineering into certified risk reduction.

Motivated by these gaps, we propose Sheaf Graph Neural Networks with PAC-Bayes Calibration (SGPC), a unified

framework that learns sheaf restrictions, calibrates their uncertainty, and optimizes the spectral gap under a provably tight PAC-Bayes bound. As illustrated in Figure 1, spectral gap optimization diversifies sheaves and separates informative frequency components from near-null modes. This can enhance the model’s ability to discriminate labels across classes and further tighten the generalization bound. Our contribution can be summarized as follows:

- We propose SGPC, a fully differentiable architecture that learns sheaf restriction maps via a Wasserstein-Entropic Lift, which optimizes the sheaf Laplacian spectral gap.
- We derive theoretical guarantees for cellular-sheaf GNNs, including convergence, spectral gap increase with risk control, and generalization bound.
- Extensive experiments on nine benchmarks demonstrate that SGPC outperforms state-of-the-art GNNs and prior sheaf models, while providing PAC-Bayes calibrated uncertainty estimates and provably tighter risk bounds.

The codes and full proofs are provided in the following

**link** — <https://github.com/ChoiYoonHyuk/SGPC>

## Related Work

**Heterophilic Graph Neural Networks.** Spectral formulations such as GCN (Kipf and Welling 2016), ChebNet (Defferrard, Bresson, and Vandergheynst 2016), and spatial attention models like GAT (Velickovic et al. 2017) rely on low-pass filters, which perform well under high homophily. Subsequent works improved depth and scalability but largely retained the homophily prior. Early solutions attempted to decouple ego and neighbor features (Zhu et al. 2020) or to sample distant yet similar nodes (Pei et al. 2020). Recent surveys catalog more than 50 heterophilily-oriented architectures (Luan et al. 2024). Notable trends include edge reweighting (Choi et al. 2025), causal discovery for message routing (Wang et al. 2025), and adaptive frequency mixing (Choi and Kim 2025). Despite this progress, many methods overlook signed structures or treat all disassortative edges uniformly, underscoring the need for a sheaf-aware neural network capable of handling heterophily.

**Sheaf Theory and Spectral Optimization.** Neural Sheaf Diffusion (NSD) (Hansen and Gebhart 2020) advances graph representation learning by endowing graphs with non-trivial cellular sheaves. On the empirical side, sheaf Laplacian-based GNN models consistently outperform baseline GCNs on signed and heterophilic benchmarks, demonstrating clear gains in classification accuracy (Barbero et al. 2022). More recent efforts have extended NSD to handle nonlinear sheaf Laplacians that capture complex interactions (Zaghen 2024). Recent works have begun complementing sheaf-based diffusion with spectral optimization techniques (Hansen and Ghrist 2019). For example, (Bodnar et al. 2022) demonstrates that the spectral gap is tightly linked to path-dependent transport maps, optimizing this via path alignment. Others incorporate directional bias through a directed cellular sheaf, deriving a directed Laplacian to improve task-specific bias (Duta et al. 2023).

**Summary & Gap.** Although existing methods effectively model heterophilic structures, they often face limitations in scalability, spectral control, or generality. In contrast, we (i) employ a joint diffusion model (Caralt et al. 2024) that learns restriction maps and features concurrently, reducing parameter count while preserving inductive bias; (ii) incorporate a spectral gap regularization term during training, ensuring better control over diffusion stability and linear separability; (iii) learn asymmetric sheaf maps under spectral constraints, enabling task-adaptive directionality while bounding eigenvalue distributions.

## Preliminaries

Let  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, H)$  denote an attribute graph with  $n = |\mathcal{V}|$  nodes and  $m = |\mathcal{E}|$  edges. The node feature matrix  $H \in \mathbb{R}^{n \times d_0}$  encodes  $d_0$ -dimensional input vectors for each node. The adjacency matrix  $A \in \{0, 1\}^{n \times n}$  captures the edge structure of  $\mathcal{G}$ , and  $D$  is a diagonal degree matrix with entries  $d_{ii} = \sum_{j=1}^n A_{ij}$ . Each node is associated with a one-hot label vector in  $Y \in \mathbb{R}^{n \times C}$ , where  $C$  is the number of classes. To quantify class consistency along edges, we define the global edge homophily ratio  $\mathcal{G}_h$  as follows:

$$\mathcal{G}_h := \frac{1}{|\mathcal{E}|} \sum_{(i,j) \in \mathcal{E}} \mathbb{I}(Y_i = Y_j), \quad (1)$$

where  $\mathbb{I}(\cdot)$  is the indicator function. Given a labeled subset  $\mathcal{V}_L \subset \mathcal{V}$ , the task of semi-supervised node classification is to infer the class labels of the remaining unlabeled nodes  $\mathcal{V}_U := \mathcal{V} \setminus \mathcal{V}_L$ . The following section introduces the cellular sheaf and geometric foundations.

**Graphs and Cellular Sheaves.** In graph  $\mathcal{G}$ , each vertex  $i$  carries an input feature  $h_i \in \mathbb{R}^{d_0}$  and a label  $y_i \in \{1, \dots, C\}$ . Throughout the paper, we fix  $d_0$  as the input feature dimension and  $d$  as the sheaf-fibre dimension. A cellular sheaf  $\mathcal{F}$  over  $\mathcal{G}$  assigns a fibre  $\mathcal{F}(v_i) = \mathbb{R}^d$  to every vertex and  $\mathcal{F}(e_{ij}) = \mathbb{R}^d$  to every edge, together with linear restriction maps  $R_{ij} : \mathcal{F}(v_i) \rightarrow \mathcal{F}(v_j)$  and  $R_{ji}$  in the opposite direction. The sheaf incidence matrix is the block matrix  $B \in \mathbb{R}^{md \times nd}$  whose  $(e_{ij}, v_i)$  block is  $R_{ij}$  and  $-R_{ji}$  for  $(e_{ij}, v_j)$ .

**Optimal Transport and Wasserstein Geometry.** Node features are interpreted as empirical probability measures in the 2-Wasserstein space  $(\mathcal{P}_2(\mathbb{R}^{d_0}), W_2)$ . Here,  $\mathcal{P}_2(\mathbb{R}^{d_0})$  denotes the set of probability measures on  $\mathbb{R}^{d_0}$  with finite second moments, and  $W_2$  is the associated 2-Wasserstein distance. Given node features  $h_i, h_j \in \mathbb{R}^{d_0}$ , we define  $\mu = h_i / \|h_i\|_1$  and  $\nu = h_j / \|h_j\|_1$  as empirical measures in  $\mathcal{P}_2(\mathbb{R}^{d_0})$  by  $\ell_1$ -normalization. Let  $e_p \in \mathbb{R}^{d_0}$  denote the  $p$ -th canonical basis vector, i.e.  $(e_p)_\ell = \mathbf{1}\{\ell = p\}$ . Then, the canonical-basis cost is given by:

$$C_{\text{feat}}[p, q] = \|e_p - e_q\|_2^2. \quad (2)$$

Consequently, the entropic optimal transport (OT) problem can be defined as follows:

$$P_\star = \arg \min_{P \in \Pi(\mu, \nu)} \langle P, C_{\text{feat}} \rangle + \varepsilon \mathcal{H}(P), \quad (3)$$

where  $\mathcal{H}(P)$  denotes the Shannon entropy  $\mathcal{H}(P) = -\sum_{ij} P_{ij} \log P_{ij}$ , encouraging smoother couplings.

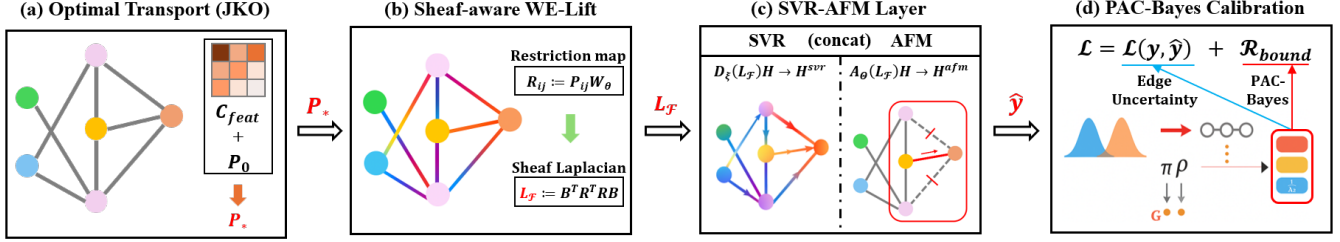


Figure 2: (a) A Jordan-Kinderlehrer-Otto (JKO) step refines the initial Sinkhorn plan  $P_0$  under the feature-cost matrix  $C_{\text{feat}}$ , producing a globally stable coupling  $P_*$ ; (b) The coupling  $P_*$  is turned into restriction maps  $R_{ij}$ , which in turn define the sheaf Laplacian  $L_{\mathcal{F}}$ ; (c) Stochastic variance-reduced diffusion  $D_{\xi}$  with adaptive frequency mixing  $A_{\theta}$  yields node-level predictions  $\hat{y}$ ; (d) A  $\beta$ -Dirichlet posterior calibrates edge uncertainty, while an optimizer enlarges the spectral gap  $\lambda_2$

## Methodology

We introduce Sheaf GNNs with PAC-Bayes Calibration (SGPC), which learns graph-structured sheaf parameters while providing a PAC-Bayes generalization bound for cellular-sheaf GNNs as follows:

1. **OT with Sheaf-aware WE Lift** refines Sinkhorn optimal transport via a single-step Wasserstein gradient flow, producing globally stable restriction maps.
2. **SVR-AFM Layer** combines (i) Stochastic Variance-Reduced (SVR) diffusion for global denoising and (ii) an Adaptive Frequency Mixing (AFM) branch.
3. **PAC-Bayes Spectral Optimization** jointly (i) calibrates edge-level uncertainty through a  $\beta$ -Dirichlet model with explicit heterophily penalties and (ii) tightens the PAC-Bayes bound by optimizing the sheaf Laplacian spectral gap  $\lambda_2$  under a perturbation constraint.

### OT with Sheaf-aware WE Lift

As shown in Figure 2-(a), we generalize the classic Sinkhorn coupling (Eq. 3) by evolving the transport plan  $P_t$  (step  $t$ ) inside the 2-Wasserstein metric space:

$$\partial_t P_t = -\nabla_{W_2} \left[ \langle P_t, C_{\text{feat}} \rangle + \varepsilon \mathcal{H}(P_t) \right], \quad (4)$$

where  $C_{\text{feat}}$  (Eq. 2) is the pairwise feature cost and  $\mathcal{H}(\cdot)$  is Shannon entropy. Starting from  $P_0 = P_{\text{Sinkhorn}}$ , a Jordan-Kinderlehrer-Otto (JKO) step refines the transport plan towards a globally KL-stable configuration as below:

$$P_* = \arg \min_P \left\{ \frac{1}{2} W_2^2(P, P_0) + \langle P, C_{\text{feat}} \rangle + \varepsilon \mathcal{H}(P) \right\}, \quad (5)$$

As in Figure 2-(b), the restriction maps  $R_{ij}$  are generated using the refined transport plan  $P_*$  as follows:

$$R_{ij} := P_{*,ij} W_{\theta}, \quad W_{\theta} \in \mathbb{R}^{d_0 \times d_0}. \quad (6)$$

Although  $P_*$  is obtained via OT, it is still updated end-to-end by the node-classification loss, making  $R_{ij}$  task-adaptive and differentiable. These maps populate the block-diagonal tensor and enter the sheaf Laplacian  $L_{\mathcal{F}}$  in Eq. 7.

### SVR-AFM Layer

The collection  $\{R_{ij}\}$  endows the graph with a cellular-sheaf structure whose co-boundary matrix is  $B \in \mathbb{R}^{|E| \times |V|}$  (edge-to-node relationship). Let  $R := \text{diag}(R_{ij})$  and define the sheaf Laplacian as below:

$$L_{\mathcal{F}} := (B \otimes I_{d_0})^{\top} R^{\top} R (B \otimes I_{d_0}) \quad (7)$$

When each  $R_{ij}$  collapses to a scalar weight, Eq. 7 reduces to the standard graph Laplacian. Given node features  $H \in \mathbb{R}^{n \times d_0}$ , diffusion hyperparameters  $\xi = (\Delta t)$ , and frequency mixing weights  $\Theta$ , we design the SVR-AFM pipeline below.

**Stochastic variance-reduced (SVR) diffusion.** As shown in the left side of Figure 2-(c), we introduce the diffusion process using the sheaf Laplacian  $L_{\mathcal{F}}$  (Eq. 7) below:

$$H^{\text{svr}} = \mathcal{D}_{\xi}(L_{\mathcal{F}})H \approx (I + \Delta t L_{\mathcal{F}})^{-1}H, \quad (8)$$

where a few SVR-preconditioned conjugate-gradient iterations approximate the inverse. Further details on solving this iteration are provided in Eq. 23.

**Adaptive Frequency Mixing (AFM) branch.** Let  $Q$  denote the maximum polynomial order and  $T_q(\cdot)$  the  $q$ -th Chebyshev polynomial. As in the right side of the Figure 2-(c), we utilize learnable frequency coefficients below:

$$\alpha_q = \frac{\exp(\gamma_q)}{\sum_{p=0}^Q \exp(\gamma_p)}, \quad q = 0, \dots, Q, \quad (9)$$

where  $\gamma_q \in \mathbb{R}$  are free parameters. The AFM representation is then obtained as follows:

$$H^{\text{afm}} = \mathcal{A}_{\Theta}(L_{\mathcal{F}})H = \sum_{q=0}^Q \alpha_q T_q(\tilde{L})H, \quad (10)$$

with  $\tilde{L} = I - D^{-1/2} L_{\mathcal{F}} D^{-1/2}$  the symmetrically normalized sheaf Laplacian. The term  $q = 0$  corresponds to the identity operator (no filtering). Higher-order terms  $q \geq 1$  serve as polynomial bases that can approximate both low- and high-pass behaviors depending on the learned coefficients  $\{\alpha_q\}$ . Consequently, on heterophilous graphs, the model tends to place more weight on combinations whose spectral response is larger at high eigenvalues (i.e., high-frequency components).

**Branch fusion.** We first concatenate the outputs from the sheaf-based diffusion branch and the AFM branch, and feed the result to a lightweight projector with two layers. Then we apply an MLP or GAT (Velickovic et al. 2017) to mix channels and propagate along the graph:

$$H' = F_{\text{mix}}([H^{\text{svr}} || H^{\text{afm}}]). \quad (11)$$

Regarding  $F_{\text{mix}}$ , we employ a GAT module for homophilic datasets and an MLP for heterophilic ones. Given the fused representations  $H'$  in Eq. 11, the class probability of each node can be inferred as follows:

$$\hat{y} = \text{softmax}(WH'). \quad (12)$$

**Computational cost** is introduced in Appendix A.

### PAC-Bayes Calibration

The stochastic restriction maps  $\{R_{ij}\}$  yielded by the WE Lift render every edge uncertain. We convert this uncertainty into a data-dependent posterior and then actively enlarge the sheaf spectral gap, obtaining a PAC-Bayes bound.

**$\beta$ -Dirichlet prior.** For each edge, we model the message agreement rate  $\kappa_{ij} \in [0, 1]$  with the following prior:

$$\kappa_{ij} \sim \text{Beta}(\alpha_{ij}, \beta_{ij}), \quad \pi = \prod_{(i,j) \in \mathcal{E}} \text{Beta}(\alpha_{ij}, \beta_{ij}). \quad (13)$$

**Posterior update.** A three-step fixed-point solver updates the posterior for each edge  $(i, j)$ . Starting from the prior, the solver iteratively (i) computes pseudo-counts of agreement based on SVR-AFM output, (ii) calibrates them through a class-coupling matrix, and (iii) normalizes the posterior to avoid over-confidence. This process starts with fixed-point updates as follows:

$$\bar{\kappa}_{ij} := \mathbb{E}_{\rho}[\kappa_{ij}] = \frac{\bar{\alpha}_{ij}}{\bar{\alpha}_{ij} + \bar{\beta}_{ij}}, \quad \rho = \prod_{(i,j) \in \mathcal{E}} \text{Beta}(\bar{\alpha}_{ij}, \bar{\beta}_{ij}). \quad (14)$$

**Empirical risk optimization.** The retrieved posterior mean  $\bar{\kappa}_{ij} := \mathbb{E}_{\rho}[\kappa_{ij}]$  serves as an edge uncertainty weight in calibrating empirical risk as below:

$$f(\hat{y}_i; \bar{\kappa}_{ij}) = \bar{\kappa}_{ij} \cdot \hat{y}_i + (1 - \bar{\kappa}_{ij}) \cdot y^{\text{prior}}, \quad (15)$$

where the  $y^{\text{prior}}$  is a pre-defined class distribution (e.g., uniform). Given the class probability  $\hat{y}$  (Eq. 12), the calibrated empirical loss is given by:

$$\mathcal{L}(y, \hat{y}) = \frac{1}{|\mathcal{V}_L|} \sum_{i \in \mathcal{V}_L} C(y_i, f(\hat{y}_i; \bar{\kappa}_{ij})) \quad (16)$$

**KL-divergence.** To regularize the posterior complexity and control generalization, we define the divergence between the prior  $\pi$  (Eq. 13) and posterior  $\rho$  (Eq. 14):

$$\mathcal{L}_{\text{KL}} = \sqrt{\frac{\text{KL}(\rho || \pi) + \log(2/\delta)}{2n}} \quad (17)$$

where  $n := |\mathcal{V}_L|$  is the number of samples and the  $\log(2/\delta)$  is confidence adjustment in PAC-Bayes bound.

---

### Algorithm 1: SGPC: One Training Epoch

---

**Require:** Graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , node features  $H$ , labels  $Y$ , parameters  $\theta$ , hyperparameters  $(\lambda_{\text{KL}}, \lambda_{\text{spec}})$

**Ensure:** Updated  $\theta$

(1) **OT**  $\rightarrow$  **WE Lift**  $\rightarrow$  **Sheaf Laplacian**

1:  $P_0 \leftarrow \text{SINKHORN}(H, \varepsilon)$

2:  $P_* \leftarrow \text{JKO}(P_0, C_{\text{feat}}, \varepsilon)$

3:  $R_{ij} \leftarrow P_{*,ij} W_{\theta}$

4:  $L_{\mathcal{F}} \leftarrow (B \otimes I_{d_0})^{\top} \text{diag}(R_{ij})^{\top} \text{diag}(R_{ij})(B \otimes I_{d_0})$

(2) **SVR-AFM Forward Pass**

5:  $H^{\text{svr}} \leftarrow \text{SVR}(H, L_{\mathcal{F}})$

6:  $H^{\text{afm}} \leftarrow \text{AFM}(H, L_{\mathcal{F}})$

7:  $H' \leftarrow F_{\text{mix}}([H^{\text{svr}} || H^{\text{afm}}])$

8:  $\hat{y} \leftarrow \text{softmax}(WH')$

(3)  **$\beta$ -Dirichlet Posterior Update**

9: **for all**  $(i, j) \in \mathcal{E}$  **parallel do**

10:  $(\bar{\alpha}_{ij}, \bar{\beta}_{ij}) \leftarrow \text{FIXEDPOINT}(\alpha_{ij}, \beta_{ij})$

11:  $\bar{\kappa}_{ij} \leftarrow \bar{\alpha}_{ij} / (\bar{\alpha}_{ij} + \bar{\beta}_{ij})$

(4) **PAC-Bayes Calibration & Parameter Update**

12:  $\mathcal{L}(y, \hat{y}) \leftarrow \frac{1}{|\mathcal{V}_L|} \sum_{i \in \mathcal{V}_L} C(y_i, f(\hat{y}_i; \bar{\kappa}_i))$

13:  $\mathcal{L}_{\text{KL}} \leftarrow \sqrt{\frac{\text{KL}(\rho || \pi) + \log(2/\delta)}{2|\mathcal{V}_L|}}$

14:  $\mathcal{L}_{\text{spec}} \leftarrow c_{\text{het}} / \lambda_2(L_{\mathcal{F}})$

15:  $\mathcal{R}_{\text{bound}} \leftarrow \lambda_{\text{KL}} \mathcal{L}_{\text{KL}} + \lambda_{\text{spec}} \mathcal{L}_{\text{spec}}$

16:  $\mathcal{L} \leftarrow \mathcal{L}(y, \hat{y}) + \mathcal{R}_{\text{bound}}$

17:  $g \leftarrow \nabla_{L_{\mathcal{F}}} \lambda_2(L_{\mathcal{F}})$

18:  $L_{\mathcal{F}} \leftarrow L_{\mathcal{F}} + \eta g$

19:  $\{R_{ij}\} \leftarrow \text{REASSEMBLEFROM}(L_{\mathcal{F}}, B)$

---

**Spectral-gap optimization.** To isolate the heterophily effect that will appear in the bound, let  $\mathcal{E}_{cc'} = \{(i, j) : y_i = c, y_j = c'\}$ . The posterior class-coupling matrix and its Frobenius norm are given by:

$$\Pi_{cc'} = \frac{1}{|\mathcal{E}_{cc'}|} \sum_{(i,j) \in \mathcal{E}_{cc'}} \bar{\kappa}_{ij}, \quad c_{\text{het}} = \|\Pi\|_F. \quad (18)$$

Let  $\lambda_2$  stand for the spectral gap. For a stable diffusion, we introduce to enlarge  $\lambda_2$  as below:

$$\lambda_2(L_{\mathcal{F}}) = \min_{v \perp \mathbf{1}} \frac{v^{\top} L_{\mathcal{F}} v}{v^{\top} v}. \quad (19)$$

Combining the heterophily penalty  $c_{\text{het}}$  (Eq. 18) and the spectral gap  $\lambda_2$  (Eq. 19) jointly characterizes the stability of sheaf diffusion in heterophilic regimes as below:

$$\mathcal{L}_{\text{spec}} = \frac{c_{\text{het}}}{\lambda_2(L_{\mathcal{F}})} \quad (20)$$

which penalizes excessive heterophily relative to the diffusion capacity of the sheaf Laplacian.

**Overall loss function.** Given the calibrated cross-entropy  $\mathcal{L}(y, \hat{y})$  (Eq. 16), KL divergence  $\mathcal{L}_{\text{KL}}$  (Eq. 17), and spectral gap  $\mathcal{L}_{\text{spec}}$  (Eq. 20), we can define the total loss as below:

$$\mathcal{L} = \mathcal{L}(y, \hat{y}) + \underbrace{\lambda_{\text{KL}} \mathcal{L}_{\text{KL}} + \lambda_{\text{spec}} \mathcal{L}_{\text{spec}}}_{\mathcal{R}_{\text{bound}}} \quad (21)$$

**Theorem 1** (PAC-Bayes Sheaf Generalization Bound). *For any  $\delta > 0$ , our model  $f$  meets the following inequality over the data distribution  $\mathcal{D}$  with probability at least  $1 - \delta$ :*

$$\mathcal{L}_{\mathcal{D}}(f) \leq \mathcal{L}(y, \hat{y}) + \mathcal{R}_{\text{bound}}. \quad (22)$$

**Proof.** Please see Appendix B.

### Theoretical Analysis

We analyze SGPC along four axes: (i) solver convergence, (ii) spectral-gap monotonicity, (iii) risk-variance contraction, and (iv) PAC-Bayes generalization. Throughout, we set  $L_t := B^\top R_t^\top R_t B$  (the sheaf Laplacian at epoch  $t$ ) and write  $\lambda_{\max,t} := \lambda_{\max}(L_t)$  and  $\lambda_{2,t} := \lambda_2(L_t)$ .

### Convergence Analysis

To solve Eq. 8, we need to approximate the linear system at every SGPC layer as below:

$$(I + \Delta t L_t)h = b, \quad (23)$$

where  $b$  denotes the input node features from the previous layer. Running Conjugate Gradient (CG) directly on the dense sheaf Laplacian  $L_t$  would cost  $O(|E|\sqrt{\kappa(L_t)})$  per solve ( $\kappa(L_t) := \lambda_{\max}(L_t)/\lambda_{\min}(L_t)$ ) and gives no iteration bound that is uniform throughout training. To guarantee epoch-wise scalability, we replace  $L_t$  by a leverage-score spectral sparsifier  $\tilde{L}_t$  containing only  $O(|V|\log|V|/\varepsilon^2)$  non-zeros, and show that CG on the shifted system  $(I + \Delta t \tilde{L}_t)h = b$  converges in  $O(\log(1/\epsilon_{\text{CG}}))$  iterations independently of  $|V|$  and of the epoch index  $t$ .

**Theorem 2** (CG convergence with sparsifier). *Let  $\tilde{L}_t$  be a  $(1 \pm \varepsilon)$  spectral sparsifier of the sheaf Laplacian  $L_t$ , obtained via leverage-score sampling as,*

$$\lambda_2(L_t) \geq \gamma \quad \text{and} \quad \lambda_{\max}(L_t) \leq \Lambda \quad (24)$$

with a time step  $\Delta t \leq 1/\Lambda$ . Then, for any right-hand side  $b$  and initial residual  $r_0$ , CG applied to  $(I + \Delta t \tilde{L}_t)h = b$  achieves a residual  $\|r_k\|_2 \leq \epsilon_{\text{CG}}$  (error bound) at most  $k_{\max}$  iterations:

$$k_{\max} \leq \left\lceil \sqrt{\kappa(I + \Delta t \tilde{L}_t)} \log \frac{\|r_0\|_2}{\epsilon_{\text{CG}}} \right\rceil = O(\log(1/\epsilon_{\text{CG}})). \quad (25)$$

The above inequality holds because

$$\kappa(I + \Delta t \tilde{L}_t) = \frac{1 + \Delta t \lambda_{\max}(\tilde{L}_t)}{1 + \Delta t \lambda_2(\tilde{L}_t)} \leq \frac{1 + (1 + \varepsilon)\Delta t \Lambda}{1 + (1 - \varepsilon)\Delta t \gamma}. \quad (26)$$

Since  $\kappa(I + \Delta t \tilde{L}_t) \leq 2 + \varepsilon = O(1)$ , we can infer that the iteration bound is uniform in  $|V|$ ,  $|E|$ , and the epoch  $t$ .

**Proof.** Please see Appendix C.

### Monotone $\lambda_2$ Growth

The PAC-Bayes bound in Eq. 22 decays with the inverse spectral gap  $\lambda_2(L_t)^{-1}$ . Thus, the optimizer  $\mathcal{U}_\phi$  is designed to monotonically enlarge  $\lambda_2$ . Any epoch that would shrink the gap would loosen the bounds and weaken the robustness guarantees. The next theorem certifies that, under a standard Wolfe back-tracking line search, every epoch leaves the gap unchanged by a positive amount.

**Theorem 3** (Wolfe-controlled gap ascent). *Let  $v_t$  be the normalized eigenvector corresponding to  $\lambda_2(L_t)$ . At epoch  $t$ , the optimizer performs the gradient ascent step,*

$$L_{t+1} = L_t + \eta_t g_t, \quad (27)$$

where  $g_t := \nabla_L(v_t^\top L_t v_t) = v_t v_t^\top$ . The step size  $\eta_t \in (0, 1]$  is chosen by a Wolfe line search with constant  $c_w \in (0, 1)$ . Then, the following inequality holds

$$\lambda_2(L_{t+1}) - \lambda_2(L_t) \geq \frac{c_w \eta_t}{2} \geq \frac{c_w}{4}. \quad (28)$$

Consequently, the sequence  $\{\lambda_2(L_t)\}_{t \geq 0}$  is strictly non-decreasing and grows by at least  $c_w/4$  once the initial full step  $\eta_t = 1$  survives the first case.

**Proof.** Please see Appendix D.

### Risk-Variance Contraction

The PAC-Bayes bound in Eq. 22 splits into three terms: empirical risk, a KL-divergence, and a spectral penalty. Because the KL term tightens the most when every edge posterior becomes sharper, we first quantify how fast the variance of each edge posterior shrinks.

**Lemma 1** (Variance reduction). *Let  $(\alpha_{ij}, \beta_{ij}) \geq 1$  be updated once per epoch by the fixed-point rule, and let  $n_{\text{tot}}(i, j)$  be the cumulative number of diffusion messages sent across edge  $(i, j)$ . Assuming that  $\gamma_{ij} := \alpha_{ij} + \beta_{ij}$ ,*

$$\text{Var}[\theta_{ij} | \mathcal{D}] \leq \frac{\gamma_{ij}}{(\gamma_{ij} + n_{\text{tot}})^2} \left(1 - \frac{1}{\gamma_{ij} + n_{\text{tot}} + 1}\right). \quad (29)$$

In particular, if  $n_{\text{tot}} \geq 5$  and  $\alpha_{ij} + \beta_{ij} \leq 10$  (less informative prior), the posterior variance is at most 60% of its initial value (i.e., has contracted by at least 40%).

**Proof.** Please see Appendix E.

**Theorem 4** (Risk-Variance Contraction). *For epoch  $t$ , let us define a loss function  $\mathcal{B}_t$  as below:*

$$\mathcal{B}_t := \underbrace{\mathcal{L}_t}_{\text{empirical risk}} + \underbrace{\sqrt{\frac{\text{KL}(\rho_t \| \pi) + \log(2/\delta)}{2n}}}_{\text{KL term}} + \underbrace{\frac{c_{\text{het}}}{\lambda_2(L_t)}}_{\text{spectral penalty}}. \quad (30)$$

Given the stochastic gradient descent with steps  $\eta_t \leq \eta_{\max}$ , and the Rayleigh-quotient ascent with Wolfe constant  $c_w$  in Eq. 28, there exists a constant  $\kappa = \kappa(\eta_{\max}, c_w) \in (0, 1)$  such that

$$\mathcal{B}_{t+1} \leq (1 - \kappa)\mathcal{B}_t, \quad \forall t \geq 0 \quad (31)$$

Thus, the PAC-Bayes bound contracts geometrically.

**Proof.** Please see Appendix E.

### Generalization Bound

The PAC-Bayes result of Theorem 1 is posterior-averaged. To convert it into a high-probability statement for the single predictor returned after training, we quantify how stable SGPC is concerning its initial parameters. The key driver of stability will be the cumulative spectral-gap gain  $\Delta_G := \sum_{t=0}^{T-1} (\lambda_2(L_{t+1}) - \lambda_2(L_t)) = \lambda_2(L_T) - \lambda_2(L_0)$ .

Dataset	Cora	Citeseer	Pubmed	Actor	Chameleon	Squirrel	Cornell	Texas	Wisconsin
$\mathcal{G}_h$ (Eq. 1)	0.81	0.74	0.80	0.22	0.23	0.22	0.11	0.06	0.16
GCN (Kipf and Welling 2016)	81.3 $\pm$ 0.74	71.1 $\pm$ 0.63	79.4 $\pm$ 0.44	20.4 $\pm$ 0.40	49.8 $\pm$ 0.59	31.0 $\pm$ 0.71	60.2 $\pm$ 0.96	68.3 $\pm$ 1.15	57.7 $\pm$ 0.97
GAT (Velickovic et al. 2017)	82.5 $\pm$ 0.52	72.0 $\pm$ 0.76	79.8 $\pm$ 0.45	21.7 $\pm$ 0.36	49.3 $\pm$ 0.84	31.1 $\pm$ 0.94	63.4 $\pm$ 1.02	70.2 $\pm$ 1.19	59.4 $\pm$ 1.10
GCNII (Chen et al. 2020)	82.1 $\pm$ 0.70	71.4 $\pm$ 1.29	79.3 $\pm$ 0.51	25.1 $\pm$ 1.22	49.1 $\pm$ 0.77	30.7 $\pm$ 0.91	<b>79.7</b> $\pm$ 1.51	<b>82.6</b> $\pm$ 1.68	<b>75.3</b> $\pm$ 1.51
H <sub>2</sub> GCN (Zhu et al. 2020)	80.2 $\pm$ 0.46	71.9 $\pm$ 0.80	78.9 $\pm$ 0.31	24.8 $\pm$ 1.16	48.0 $\pm$ 0.83	31.3 $\pm$ 0.75	78.3 $\pm$ 1.45	79.0 $\pm$ 1.56	73.3 $\pm$ 1.45
Geom-GCN (Pei et al. 2020)	82.2 $\pm$ 0.40	71.8 $\pm$ 0.55	79.0 $\pm$ 0.33	24.6 $\pm$ 0.41	51.5 $\pm$ 0.64	<b>32.6</b> $\pm$ 0.78	75.9 $\pm$ 1.48	70.0 $\pm$ 1.62	73.3 $\pm$ 1.53
GPRGNN (Chien et al. 2020)	81.9 $\pm$ 0.57	71.7 $\pm$ 0.84	79.5 $\pm$ 0.56	24.1 $\pm$ 0.88	50.7 $\pm$ 0.80	30.5 $\pm$ 0.63	74.0 $\pm$ 1.72	72.8 $\pm$ 1.49	74.3 $\pm$ 1.49
GloGNN (Li et al. 2022)	<b>82.8</b> $\pm$ 0.40	<b>72.5</b> $\pm$ 0.53	<b>80.2</b> $\pm$ 0.28	<b>25.9</b> $\pm$ 0.72	48.8 $\pm$ 0.69	31.1 $\pm$ 0.81	70.8 $\pm$ 1.38	74.9 $\pm$ 1.51	70.9 $\pm$ 1.40
Auto-HeG (Zheng et al. 2023)	82.5 $\pm$ 1.07	71.6 $\pm$ 1.42	80.0 $\pm$ 0.24	25.4 $\pm$ 0.99	49.2 $\pm$ 1.38	31.8 $\pm$ 1.12	77.2 $\pm$ 1.24	<b>80.6</b> $\pm$ 2.06	<b>75.6</b> $\pm$ 1.83
NSD (Bodnar et al. 2022)	81.6 $\pm$ 0.39	71.4 $\pm$ 0.28	78.8 $\pm$ 0.11	22.6 $\pm$ 2.70	49.4 $\pm$ 1.44	31.3 $\pm$ 1.21	68.0 $\pm$ 3.13	63.4 $\pm$ 2.74	67.3 $\pm$ 2.88
SheafAN (Barbero et al. 2022)	81.9 $\pm$ 0.43	71.5 $\pm$ 0.30	78.9 $\pm$ 0.09	23.0 $\pm$ 1.08	49.8 $\pm$ 0.45	31.4 $\pm$ 0.84	70.1 $\pm$ 2.57	77.4 $\pm$ 3.25	69.7 $\pm$ 2.95
JacobiConv (Wang and Zhang 2022)	82.7 $\pm$ 0.70	<b>73.0</b> $\pm$ 0.76	79.5 $\pm$ 0.42	25.3 $\pm$ 1.05	52.0 $\pm$ 1.11	32.4 $\pm$ 0.74	<b>79.5</b> $\pm$ 1.34	74.6 $\pm$ 1.52	72.0 $\pm$ 1.26
SheafHyper (Duta et al. 2023)	82.3 $\pm$ 0.45	71.7 $\pm$ 0.30	79.0 $\pm$ 0.06	23.4 $\pm$ 1.12	49.9 $\pm$ 0.45	31.6 $\pm$ 0.40	73.5 $\pm$ 3.24	78.9 $\pm$ 2.78	71.9 $\pm$ 2.97
NLSD (Zaghen et al. 2024)	82.0 $\pm$ 0.39	72.3 $\pm$ 0.41	78.9 $\pm$ 0.05	22.2 $\pm$ 2.24	51.4 $\pm$ 0.97	31.2 $\pm$ 0.75	66.2 $\pm$ 2.26	73.6 $\pm$ 2.58	72.9 $\pm$ 2.86
SimCalib (Tang et al. 2024)	82.7 $\pm$ 0.41	71.4 $\pm$ 0.63	78.9 $\pm$ 0.11	23.0 $\pm$ 3.08	<b>53.1</b> $\pm$ 0.62	<b>33.0</b> $\pm$ 0.90	69.4 $\pm$ 3.17	71.4 $\pm$ 2.74	69.1 $\pm$ 2.90
PCNet (Li, Pan, and Kang 2024)	<b>83.3</b> $\pm$ 0.77	72.2 $\pm$ 1.21	<b>80.1</b> $\pm$ 0.26	<b>25.7</b> $\pm$ 0.86	<b>52.5</b> $\pm$ 1.70	31.7 $\pm$ 0.57	77.2 $\pm$ 1.30	74.7 $\pm$ 1.43	71.5 $\pm$ 1.33
<b>SGPC (ours)</b>	<b>83.0</b> $\pm$ 0.55	<b>72.6</b> $\pm$ 0.21	<b>79.9</b> $\pm$ 0.06	<b>38.1</b> $\pm$ 0.52	<b>53.3</b> $\pm$ 1.29	<b>36.0</b> $\pm$ 0.30	<b>81.0</b> $\pm$ 2.33	<b>83.2</b> $\pm$ 1.82	<b>81.1</b> $\pm$ 2.60

Table 1. (RQ1) Node classification performance (%) on nine benchmarks, where we highlight the top-3 values in each column. The upper methods are focused on message passing, whereas the lower blocks leverage sheaf diffusion and spectral optimization

**Lemma 2** (Algorithmic stability bound). *Assume the time-step satisfies  $\Delta t < 1/\lambda_{\max}$  and let  $\epsilon_{CG}$  be the residual tolerance used in every CG solve. Then, the SGPC encoder after  $T$  epochs  $f_T$  obeys the following inequality:*

$$\|f_T - f_0\|_2 \leq \sqrt{\frac{\lambda_{\max}}{\lambda_2(L_0)}} \exp\left(-\frac{\Delta t \Delta_G}{2}\right) + \epsilon_{CG} T. \quad (32)$$

If  $\Delta_G$  grows linearly in  $T$  (as guaranteed by Theorem 3), the first term decays exponentially fast, while the CG term can be made negligible by choosing  $\epsilon_{CG} = O(T^{-2})$ .

**Proof.** Please see Appendix F.

**Theorem 5** (PAC-Bayes population risk). *Combine Lemma 2 with Theorems 1 (PAC-Bayes) and 4 (risk-variance contraction). Choosing  $\epsilon_{CG} T \leq \exp(-\frac{\Delta t \Delta_G}{2})$ , the following inequality holds with probability at least  $1 - \delta$ :*

$$\mathcal{L}_{\mathcal{D}}(f) \leq \mathcal{L} + \sqrt{\frac{2 \exp(-\frac{\Delta t \Delta_G}{2})}{|\mathcal{V}_L|}} + O\left(\sqrt{\frac{\log(1/\delta)}{|\mathcal{V}_L|}}\right). \quad (33)$$

Therefore, the generalization gap shrinks exponentially in the cumulative gap gain  $\Delta_G$ .

**Proof.** Please see Appendix F.

## Experiments

We performed comprehensive experiments to support our theoretical analysis, focusing on the following research questions (RQs). For empirical evaluation, we adopted node classification, a widely studied task in graph mining.

- **RQ1:** Does SGPC enhance node classification accuracy in graph neural networks?
- **RQ2:** How does PAC-Bayes calibrated spectral optimization affect generalization, and does each module (SVR vs AFM) contribute to the overall performance?
- **RQ3:** How do the hyperparameters  $\lambda_{KL}$  and  $\lambda_{\text{spec}}$  (Eq. 21) affect the overall performance of the model?
- **RQ4:** Do the proposed strategies alleviate over-smoothing when stacking deeper layers?

## Implementation

All models are implemented using PyTorch Geometric with the Adam optimizer (learning rate =  $1 \times 10^{-3}$ ) and a weight decay of  $5 \times 10^{-4}$ . The hyperparameters include a diffusion step  $\Delta t = (0.02, 0.5)$  for homophilic, heterophilic datasets. We set inner gradient steps as  $K = 5$  per epoch during spectral optimization while enforcing the perturbation constraint. The  $\beta$ -Dirichlet calibration adopts  $a_0 = b_0 = 1.0$ . Following (Kipf and Welling 2016), 20 labeled nodes per class are randomly selected for training, with the remaining nodes split into validation and test sets.

**Datasets and Baselines.** Please see Appendix G.

## (RQ1) Main Results

Table 1 illustrates the performance across nine benchmarks. The GCN and GAT still provide solid baselines on homophilic networks (e.g., Cora), yet their performance drops sharply on other datasets. Depth-controlled variants such as GCNII and H<sub>2</sub>GCN partially mitigate over-smoothing, but deeper stacks reveal sensitivity to overfitting on smaller graphs (WebKB). Models that explicitly re-weight messages or blend global feedback: Geom-GCN, GPRGNN, GloGNN, and Auto-HeG outperform their classical counterparts on the moderately heterophilic (Cornell and Wisconsin) datasets. Nonetheless, their gains are less consistent in highly heterophilic settings, where variance across random splits remains high.

Methods grounded in sheaf diffusion or spectral filtering generally excel whenever long-range, cross-community signals dominate. In particular, JacobiConv and SimCalib secure leading positions on Chameleon and Squirrel thanks to spectrum realignment and calibration. However, their ranking fluctuates on the WebKB-style graphs, indicating limited robustness under severe data scarcity or topological sparsity. Our method (SGPC) ranks first on six datasets and second on the remaining three, producing the most balanced performance profile of all contenders. While deeper message passing and spectral diffusion each alleviate specific weaknesses of classical GNNs, they often trade stability on one

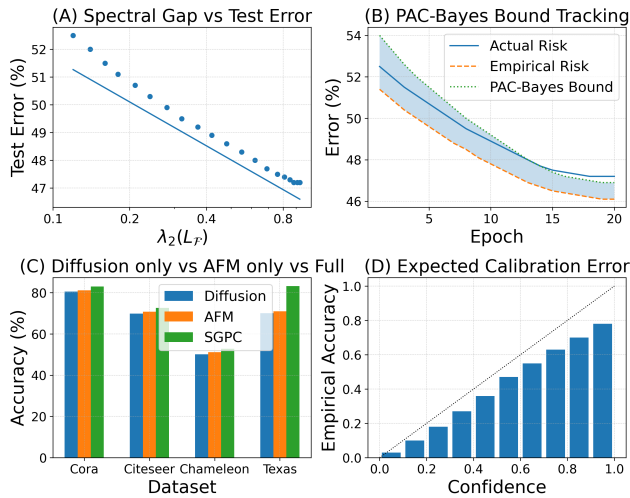


Figure 3: (RQ2) Ablation and generalization study

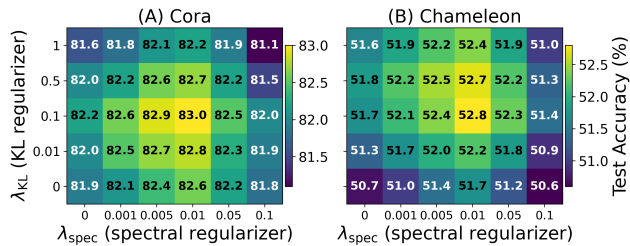


Figure 4: (RQ3) Hyperparameter analysis on loss function

graph family for gains on another. SGPC reconciles these objectives, delivering consistently high accuracy and variance control without losing scalability.

### (RQ2) Ablation Study

Figure 3 summarizes our ablation and generalization study. Panel (a) now reports results on the heterophilic Chameleon dataset: enlarging the spectral gap  $\lambda_2(L_{\mathcal{F}})$  (dots) produces an almost linear decrease in test error, empirically supporting our gap-optimization objective. Figure (b) tracks training on the same split; as epochs proceed, the PAC-Bayes bound (green) tightens monotonically and the true test risk converges toward it. Panel (c) disentangles the two architectural blocks, SVR diffusion and AFM. Diffusion dominates homophilic graphs (Cora, Citeseer), whereas AFM contributes more to heterophilic graphs (Chameleon, Texas). Their combination consistently outperforms the stronger single-branch baseline by about 2%. Finally, Figure (d) shows a reliability diagram on Chameleon whose bars remain close to the diagonal, yielding an Expected Calibration Error of  $\approx 9.3\%$ . This indicates that the  $\beta$ -Dirichlet posterior still provides reasonably well-calibrated predictions. Overall, the four panels confirm that (i) maximizing the spectral gap directly reduces error, (ii) the PAC-Bayes bound is tight in practice, (iii) SVR and AFM are complementary, and (iv) output probabilities remain well calibrated.

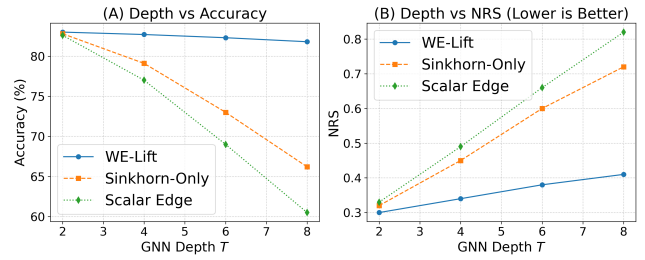


Figure 5: (RQ4) Over-smoothing analysis on Cora dataset, where the metrics are Accuracy ( $\uparrow$ ) and NRS ( $\downarrow$ )

### (RQ3) Hyperparameter Sensitivity

We investigate how the KL-divergence  $\lambda_{KL}$  and the spectral regularizer weights  $\lambda_{spec}$  in Eq. 21 affect node classification accuracy. As shown in Fig. 4, overly small  $\lambda_{KL}$  (e.g.,  $\lambda_{KL} \leq 0.01$ ) degrades performance, whereas a moderate value (approximately  $\lambda_{KL} = 0.1$ ) yields the best results by stabilizing posterior complexity without over-penalizing model capacity. Accuracy also exhibits an inverted U trend concerning  $\lambda_{spec}$ , which peaks at mid-range settings and drops when the spectral penalty is too weak to shape the operator or too strong to over-smooth informative components. These trends confirm that balanced KL control and well-tempered spectral regularization jointly maximize performance under our PAC-Bayes calibration objective.

### (RQ4) Alleviating Over-smoothing

To quantify depth-induced over-smoothing, we incrementally stacked 8 SVR-AFM layers and trained each model for 200 epochs with early stopping (patience=30) on the Cora dataset. Figure 5(a) shows that WE Lift maintains an accuracy of 81.8% (a drop of only 1.2%), whereas the Sinkhorn-only and scalar-edge variants (w/o sheaf Laplacian) experience sharp declines of approximately 16.6% and 22.1%, respectively. A similar trend is observed in Figure 5(b), where we use Neighborhood Representation Similarity (NRS), defined as the average pairwise cosine similarity of node embeddings. We observe that NRS increases mildly for WE Lift but exceeds 0.7 for the baselines at  $T = 8$ . These results demonstrate that a single Wasserstein-Entropic lift step is sufficient to preserve feature diversity and mitigate over-smoothing, enabling much deeper stacks of sheaf-based GNNs without sacrificing performance.

## Conclusion

We introduce Sheaf GNNs with PAC-Bayes Calibration (SGPC), a framework integrating restriction maps, variance-reduced diffusion with adaptive mixing, and PAC-Bayes calibrated spectral optimization. Our theoretical analysis establishes the first PAC-Bayes generalization bound for cellular-sheaf GNNs, explicitly linking heterophily penalties and the sheaf spectral gap. Extensive experiments on both homophilic and heterophilic benchmarks demonstrate that SGPC consistently outperforms classical and state-of-the-art GNNs while providing calibrated uncertainty estimates.

## Acknowledgments

This research was supported by Sookmyung Women’s University Research Grants (1-2503-2027), by the KENTECH Research Grant (202200019A), and by the Institute of Information & Communications Technology Planning & Evaluation (IITP)-Innovative Human Resource Development for Local Intellectualization program grant funded by the Korea government (MSIT) (IITP-2025-RS-2022-00156287).

## References

- Barbero, F.; Bodnar, C.; de Ocariz Borde, H. S.; and Lio, P. 2022. Sheaf attention networks. In *NeurIPS 2022 Workshop on Symmetry and Geometry in Neural Representations*.
- Bo, D.; Wang, X.; Shi, C.; and Shen, H. 2021. Beyond low-frequency information in graph convolutional networks. *arXiv preprint arXiv:2101.00797*.
- Bodnar, C.; Di Giovanni, F.; Chamberlain, B. P.; Liò, P.; and Bronstein, M. M. 2022. Neural sheaf diffusion: A topological perspective on heterophily and oversmoothing in gnns. *arXiv preprint arXiv:2202.04579*.
- Brody, S.; Alon, U.; and Yahav, E. 2021. How attentive are graph attention networks? *arXiv preprint arXiv:2105.14491*.
- Caralt, F. H.; Gil, G. B.; Duta, I.; Liò, P.; and Cot, E. A. 2024. Joint diffusion processes as an inductive bias in sheaf neural networks. *arXiv preprint arXiv:2407.20597*.
- Chen, D.; Liu, R.; Hu, Q.; and Ding, S. X. 2021. Interaction-aware graph neural networks for fault diagnosis of complex industrial processes. *IEEE Transactions on neural networks and learning systems*, 34(9): 6015–6028.
- Chen, M.; Wei, Z.; Huang, Z.; Ding, B.; and Li, Y. 2020. Simple and deep graph convolutional networks. In *International Conference on Machine Learning*, 1725–1735. PMLR.
- Chien, E.; Peng, J.; Li, P.; and Milenkovic, O. 2020. Adaptive universal generalized pagerank graph neural network. *arXiv preprint arXiv:2006.07988*.
- Choi, Y.; and Kim, C.-K. 2025. SpecSphere: Dual-Pass Spectral-Spatial Graph Neural Networks with Certified Robustness. *arXiv preprint arXiv:2505.08320*.
- Choi, Y.; Ko, T.; Choi, J.; and Kim, C.-K. 2025. Beyond Binary: Improving Signed Message Passing in Graph Neural Networks for Multi-Class Graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29.
- Duta, I.; Cassarà, G.; Silvestri, F.; and Liò, P. 2023. Sheaf hypergraph networks. *Advances in Neural Information Processing Systems*, 36: 12087–12099.
- Fan, W.; Ma, Y.; Li, Q.; He, Y.; Zhao, E.; Tang, J.; and Yin, D. 2019. Graph neural networks for social recommendation. In *The world wide web conference*, 417–426.
- Hansen, J.; and Gebhart, T. 2020. Sheaf neural networks. *arXiv preprint arXiv:2012.06333*.
- Hansen, J.; and Ghrist, R. 2019. Toward a spectral theory of cellular sheaves. *Journal of Applied and Computational Topology*, 3(4): 315–358.
- Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- Letarte, G.; Germain, P.; Guedj, B.; and Laviolette, F. 2019. Dichotomize and generalize: PAC-Bayesian binary activated deep neural networks. *Advances in Neural Information Processing Systems*, 32.
- Li, B.; Pan, E.; and Kang, Z. 2024. Pc-conv: Unifying homophily and heterophily with two-fold filtering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 13437–13445.
- Li, X.; Zhu, R.; Cheng, Y.; Shan, C.; Luo, S.; Li, D.; and Qian, W. 2022. Finding Global Homophily in Graph Neural Networks When Meeting Heterophily. *arXiv preprint arXiv:2205.07308*.
- Luan, S.; Hua, C.; Lu, Q.; Ma, L.; Wu, L.; Wang, X.; Xu, M.; Chang, X.-W.; Precup, D.; Ying, R.; et al. 2024. The heterophilic graph learning handbook: Benchmarks, models, theoretical analysis, applications and challenges. *arXiv preprint arXiv:2407.09618*.
- Pei, H.; Wei, B.; Chang, K. C.-C.; Lei, Y.; and Yang, B. 2020. Geom-gcn: Geometric graph convolutional networks. *arXiv preprint arXiv:2002.05287*.
- Tang, B.; Wu, Z.; Wu, X.; Huang, Q.; Chen, J.; Lei, S.; and Meng, H. 2024. Simcalib: Graph neural network calibration based on similarity between nodes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 15267–15275.
- Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *stat*, 1050: 20.
- Wang, B.; Li, J.; Chang, H.; Zhang, K.; and Tsung, F. 2025. Heterophilic Graph Neural Networks Optimization with Causal Message-passing. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, 829–837.
- Wang, X.; and Zhang, M. 2022. How powerful are spectral graph neural networks. In *International Conference on Machine Learning*, 23341–23362. PMLR.
- Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2018. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*.
- Zaghen, O. 2024. Nonlinear sheaf diffusion in graph neural networks. *arXiv preprint arXiv:2403.00337*.
- Zaghen, O.; Longa, A.; Azzolin, S.; Telyatnikov, L.; Passerini, A.; et al. 2024. Sheaf diffusion goes nonlinear: Enhancing gnns with adaptive sheaf laplacians. *PROCEEDINGS OF MACHINE LEARNING RESEARCH*, 251.
- Zhang, X.-M.; Liang, L.; Liu, L.; and Tang, M.-J. 2021. Graph neural networks and their current applications in bioinformatics. *Frontiers in genetics*, 12: 690049.

Zheng, X.; Zhang, M.; Chen, C.; Zhang, Q.; Zhou, C.; and Pan, S. 2023. Auto-heg: Automated graph neural network on heterophilic graphs. *arXiv preprint arXiv:2302.12357*.

Zhou, W.; Veitch, V.; Austern, M.; Adams, R. P.; and Orbanz, P. 2018. Non-vacuous generalization bounds at the imagenet scale: a PAC-bayesian compression approach. *arXiv preprint arXiv:1804.05862*.

Zhu, J.; Yan, Y.; Zhao, L.; Heimann, M.; Akoglu, L.; and Koutra, D. 2020. Beyond homophily in graph neural networks: Current limitations and effective designs. *Advances in Neural Information Processing Systems*, 33: 7793–7804.