INTERWEAVING TOPOLOGY AND SPECTRUM: FEDERATED SPECTRAL-GUIDED RIEMANNIAN GRAPH LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated Graph Learning (FGL) enables collaborative model training on distributed graph data while preserving privacy. However, structural and frequency heterogeneity across clients induces two critical challenges: local embedding distortion in flat Euclidean spaces, and global knowledge collapse caused by entangling disparate spatial topologies with spectral features. To address these challenges, we propose Federated Spectral-Guided Riemannian Graph Learning (FedSGR), a novel framework that systematically interweaves spatial topological structures with frequency domain signals. Specifically, FedSGR eliminates embedding distortion by establishing personalized product manifolds with mixed curvature tailored to local topologies. Concurrently, it circumvents knowledge collapse by distilling domain invariant frequency patterns globally, which are adaptively fused with local curvature features to steer personalized message passing directly on mixed manifolds. Extensive experiments on various graph benchmarks demonstrate that FedSGR consistently surpasses state-of-the-art FGL methods, exhibiting superior predictive accuracy and robust personalized adaptation.

1 INTRODUCTION

Federated Graph Learning emerges as an essential paradigm for preserving privacy, enabling collaborative model training on graph data across distributed clients Du et al. (2026); Fu et al. (2026). Unlike traditional Graph Neural Networks (GNNs) inherently designed for unified, centralized graphs Hamilton (2020); Kipf (2016), real world graph data is frequently distributed across isolated institutions or edge devices due to legal regulations and data silo constraints McMahan et al. (2017); Li et al. (2020). By enabling clients to independently train local GNNs on private graph data while aggregating model updates at a central server, this paradigm finds widespread application across diverse domains, including pharmaceutical drug discovery Zhang et al. (2026), personalized user recommendation Wu et al. (2021), and urban traffic forecasting Zhang et al. (2024), *etc.*Beyond localized task execution, FGL establishes a vital cornerstone for distributed artificial intelligence, unlocking massive distributed relational knowledge to empower collaborative foundation models while strictly upholding data sovereignty.

However, as graph volume and client scale expand, heterogeneously distributed client data (*i.e.*, non-IID data) severely hinders model performance Smith et al. (2017); Wang et al. (2026). As illustrated in Figure 1, the above problem is exceptionally complex in graph data, as heterogeneity manifests not merely in feature or label skews, but fundamentally in the underlying non-Euclidean topologies and different frequency distributions across clients Liu et al. (2019); Bachmann et al. (2020). Consequently, analyzing this structural heterogeneity reveals two progressive challenges spanning from localized geometric representation to collaborative knowledge modeling: at the local level, client subgraphs with complex topologies suffer from *embedding distortion* when constrained to flat Euclidean spaces; progressing to the collaborative learning stage, conventional frameworks fail to effectively decouple and reconcile spatial topological patterns with spectral frequency features, resulting in severe *knowledge collapse* where entangled spatial and frequency signals wash out distinguishable structural features, losing meaningful information for local adaptation.

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.

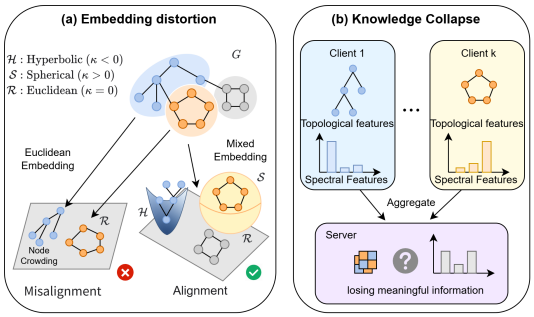


Figure 1: Illustration of the two challenges in Federated Graph Learning.

Regarding *embedding distortion* (Figure 1(a)), Riemannian Graph Learning Xiong et al. (2022) provides an appealing avenue by embedding graphs into curved manifolds matching their geometry. In the alignment view, hyperbolic space $\mathcal{H}$ ($\kappa < 0$) eliminates node crowding in hierarchical trees, spherical space $\mathcal{S}$ ($\kappa > 0$) preserves cyclic community symmetry, and Euclidean space $\mathcal{R}$ ($\kappa = 0$) accommodates regular grids Gu et al. (2018); Sun et al. (2022). While mixed curvature manifolds can theoretically fit concurrent topologies Sun et al. (2023b), existing frameworks rely on centralized optimization or enforce uniform global geometries, failing to provide personalized curvature adaptation for distributed clients.

Concurrently, the dilemma of *knowledge collapse* (Figure 1(b)) points toward spectral graph theory Liu et al. (2022); Lin et al. (2023), which captures macroscopic frequency patterns that are more transferable than local spatial edges Bo et al. (2023a); Jamadandi et al. (2024). In principle, isolating universal frequency responses allows clients to share generic structural regularities despite divergent spatial topologies, such as low frequency trees on Client 1 versus high frequency cycles on Client k. However, existing spectral FGL frameworks Tan et al. (2024) remain confined to flat Euclidean spaces and fail to decouple spatial topologies from spectral characteristics, collapsing shared representations into uninformative compromises that cannot guide local learning.

To address these two challenges, we propose Federated Spectral-Guided Riemannian Graph Learning (FedSGR), an innovative framework that first systematically interweaves spatial topological structures with frequency domain signals in federated graph learning. Specifically, to eliminate embedding distortion, FedSGR maps node features into client specific product manifolds with mixed curvature, establishing a structurally faithful non-Euclidean foundation that accommodates diverse tree and cycle motifs. To circumvent knowledge collapse, FedSGR decouples transferable frequency patterns from localized spatial topologies, collaboratively distilling domain invariant spectral priors via a shared filter generator. Finally, an adaptive dual gated Riemannian convolution dynamically fuses global spectral guidance with local discrete Ricci curvature, steering personalized message passing on mixed manifolds without degenerating into uninformative compromises. Our **main contributions** can be summarized as follows:

- We propose FedSGR, representing the first federated graph learning framework that systematically interweaves spatial topological information with frequency domain signals to tackle non-IID graph data.
- We design a synergistic learning paradigm where personalized mixed curvature embeddings faithfully extract geometric features, while global spectral knowledge sharing coupled with a local dual gated network achieves an organic knowledge fusion for local client learning.
- Extensive experiments demonstrate that FedSGR significantly outperforms state-of-the-art FGL methods, particularly in scenarios with high heterogeneity.

2 Related Work

2.1 Federated Graph Learning

Federated Graph Learning enables multiple participants to collaboratively train GNNs on decentralized data, addressing challenges of privacy and data silos Huang et al. (2024). To address the

challenge of data heterogeneity, some methods utilize regularization or adaptive aggregation to balance global consensus with local optimization Karimireddy et al. (2020); Zhang et al. (2023). Other approaches employ client clustering based on gradient similarity or data signatures to group participants with similar distributions for joint training Baek et al. (2023). To further mitigate negative transfer, several frameworks focus on parameter decoupling, separating shared global knowledge from client specific local features Jiao et al. (2025). Furthermore, Guan *et al.* Guan et al. (2025) explored adaptive neighbor selection and homophily based weighting to handle structural inconsistencies across diverse domains, while FedPAM Fu et al. (2026) introduces an unsupervised paradigm to learn generalizable graph representations without label supervision. However, these frameworks predominantly operate within flat Euclidean spaces and lack a principled mechanism to decouple spatial topological variations from spectral frequency characteristics, inevitably inducing severe embedding distortion locally and knowledge collapse globally.

2.2 Spectral Graph Neural Networks

Spectral Graph Neural Networks operate in the graph frequency domain by defining spectral filters to capture structural information Chen et al. (2023); Bo et al. (2023c). Early approaches utilize fixed filters like Personalized PageRank Page et al. (1998), while subsequent methods employ learnable orthogonal polynomials such as Bernstein He et al. (2021) or Chebyshev He et al. (2022) to fit diverse spectral distributions. Recent advancements introduce neural parameterized spectral encoders that leverage attention mechanisms to model complex frequency dependencies Bo et al. (2023b). However, existing spectral frameworks remain confined to flat Euclidean geometry and isolate frequency filtering from local curved topologies, inducing knowledge collapse when handling distributed graphs with divergent geometries.

2.3 Riemannian Graph Learning

Riemannian Graph Learning offers a powerful non-Euclidean alternative to traditional GNNs by leveraging curved manifolds to faithfully represent intrinsic graph structures Gu et al. (2018). To minimize distortion, hyperbolic spaces model hierarchical networks Chami et al. (2019), while spherical manifolds capture cyclical patterns Zhu et al. (2021). For real world graphs with diverse topologies, recent studies transition toward mixed curvature product manifolds Bachmann et al. (2020) or leverage discrete Ricci curvature for topology aware message passing Sun et al. (2023a). However, these approaches assume centralized data or uniform geometries; they lack client tailored curvature adaptation to eliminate embedding distortion in federated settings, while overlooking universal frequency domain patterns that guide local message passing.

3 Preliminaries

3.1 Riemannian Geometry

Riemannian Manifold. Do Carmo & Flaherty Francis (1992) A Riemannian manifold $(\mathcal{M}, g)$ is a smooth manifold $\mathcal{M}$ equipped with a Riemannian metric g. For each point $x \in \mathcal{M}$, the metric g_x defines an inner product on the tangent space $\mathcal{T}_x\mathcal{M}$, serving as the local linear Euclidean approximation of the manifold at x. Two essential operations connect the manifold and its tangent space:

- The exponential map $\exp_x : \mathcal{T}_x\mathcal{M} \to \mathcal{M}$ projects a vector $v \in \mathcal{T}_x\mathcal{M}$ from the tangent space onto the manifold.
- The logarithmic map $\log_x : \mathcal{M} \to \mathcal{T}_x\mathcal{M}$ is the inverse of the exponential map, projecting a point $y \in \mathcal{M}$ back to the tangent space at x.

These maps allow us to leverage the well-defined vector space structure of the tangent space for linear transformations and aggregations, before mapping representations back onto the manifold.

Curvature and Curvature Spaces. For each point x in a manifold, it is associated with a curvature K_x which describes how the geometry of the space around x deviates from being flat. When this curvature K_x is constant for all points across the manifold, it induces a **Constant Curvature Space**

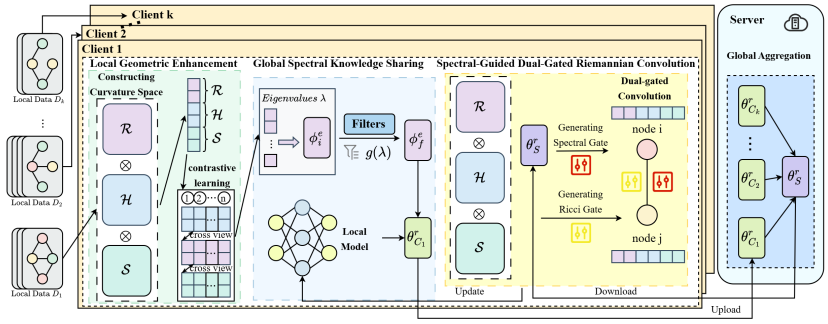


Figure 2: The Framework of the proposed FedSGR

(CCS). Based on the sign of K, these spaces are categorized into three canonical types: hyperbolic space $\mathcal{H}^d$ ($K < 0$), hyperspherical space $\mathcal{S}^d$ ($K > 0$), and Euclidean space $\mathbb{R}^d$ ($K = 0$), which provide distinct geometric inductive biases for modeling hierarchical, cyclical, and flat data structures, respectively.

3.2 Spectral Decomposition and Graph Filtering

Given an undirected graph $G = (\mathcal{V}, \mathcal{E})$, its topological structure can be described by the normalized graph Laplacian matrix $L = I_n - D^{-1/2}AD^{-1/2}$. Since L is a real symmetric matrix, it can be spectrally decomposed as $L = U\Lambda U^T$, where the columns of U are the orthonormal eigenvectors of L, and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$ is the diagonal matrix of corresponding eigenvalues (i.e., the graph's frequencies), which lie in the range $[0, 2]$.

In spectral graph theory, any signal $\mathbf{x} \in \mathbb{R}^n$ on the graph can be projected into the spectral domain via the Graph Fourier Transform (GFT): $\hat{\mathbf{x}} = U^T\mathbf{x}$. The inverse transform is given by $\mathbf{x} = U\hat{\mathbf{x}}$.

Filtering a graph signal relies on designing a filter function $g : [0, 2] \to \mathbb{R}$ that modulates each frequency component via element-wise multiplication with the eigenvalues. The filtering process in the spatial domain can be concisely represented as the operator $g(L)$ acting on a signal $\mathbf{x}$:

$$g(L)\mathbf{x} = Ug(\Lambda)U^T\mathbf{x}. \tag{1}$$

To avoid computationally expensive eigendecomposition, $g(L)$ is often approximated by a K-order polynomial $g(L) \approx \sum_{k=0}^{K} \alpha_k L^k$ parameterized by learnable coefficients α_k.

4 Methodology

4.1 Problem Statement

Given a federated setting with K clients, denoted by $\mathcal{C} = \{1, 2, \ldots, K\}$, each client k holds a private local graph dataset $\mathcal{D}_k = \{(G_k, Y_k)\}$ drawn from a distribution $p_k(G, Y)$. The goal of Federated Graph Learning is to collaboratively train a global model while handling the non-IID data heterogeneity ($p_i \neq p_j$) across clients. Specifically, this heterogeneity manifests in two dimensions: feature heterogeneity, where node attributes follow different distributions across domains; and topological heterogeneity, where the underlying graph structures (*e.g.*, degree distribution, local curvature, and connectivity patterns) vary significantly. In this framework, we decouple the learnable parameters on client k at round r into shared and personalized parts. Specifically, only the spectral filter parameters $\phi^e_{f,k}$ are uploaded and aggregated into the global parameter set θ^r_S, while all geometric parameters (*e.g.*, manifold curvatures and local projection weights) remain strictly local on each client. The objective is to minimize the global loss function:

$$\min_{\{\theta_{C_k}\}_{k=1}^{K}} \sum_{k=1}^{K} w_k \mathcal{L}_k(f(G_k; \theta_{C_k}), Y_k), \tag{2}$$

where w_k is the aggregation weight for client k, and $\mathcal{L}_k$ is the local loss.

4.2 THE OVERVIEW OF FEDSGR

As illustrated in Figure 2, FedSGR resolves embedding distortion and knowledge collapse through three coordinated stages. First, **Local Geometric Enhancement** (Section 4.3) maps node features into personalized product manifolds with mixed curvature to eliminate geometric distortion. Second, **Global Spectral Knowledge Sharing** (Section 4.4) distills domain invariant frequency patterns via a shared filter generator to prevent knowledge collapse. Finally, **Spectral-Guided Dual-Gated Riemannian Convolution** (Section 4.5) adaptively fuses global spectral guidance with local Ricci curvature for personalized message passing on mixed manifolds.

4.3 LOCAL GEOMETRIC ENHANCEMENT

To eliminate *embedding distortion* induced by flat Euclidean spaces, we introduce a local geometric enhancement module that embeds node features into personalized product manifolds with mixed curvature. By assigning distinct constant curvature spaces to accommodate concurrent hierarchical and cyclical topologies, this module utilizes geometric contrastive learning to inject non-Euclidean structural priors into node representations, establishing a structurally faithful foundation for subsequent spectral extraction.

$(\mathbf{1})$ **Personalized Product Manifold Construction.** For each client c, we construct a personalized product manifold $\mathcal{M}_c$ as the Cartesian product of Euclidean, hyperbolic, and hyperspherical spaces:

$$\mathcal{M}_c = \mathbb{R}^{d_e} \otimes \mathcal{H}^{d_h}_{K^h_c} \otimes \mathcal{S}^{d_s}_{K^s_c}, \tag{3}$$

where $\otimes$ denotes the Cartesian product. The parameters defining this space include dimensions $\{d_e, d_h, d_s\}$ and the Curvature Parameters $\mathcal{K}_c = \{K^h_c, K^s_c\}$. Crucially, $\mathcal{K}_c$ are client specific and learned locally without participating in server-side aggregation, allowing the geometry of the embedding space to adaptively reflect the local structural characteristics of client c's data.

$(\mathbf{2})$ **Multi-View Geometric Mapping.** To capture the complex geometric properties of the graph data, we project the raw node features $\mathbf{x}_u \in \mathbb{R}^F$ into hyperbolic space (to capture tree-like/hierarchical structures), spherical space (to capture cyclical structures), and Euclidean space respectively:

$$\mathbf{h}^{\mathbb{E}}_u = \mathbf{W}_e \mathbf{x}_u, \quad \mathbf{h}^{\mathbb{H}}_u = \exp^{K^h_c}_{\mathbf{0}}(\mathbf{W}_h \mathbf{x}_u), \quad \mathbf{h}^{\mathbb{S}}_u = \exp^{K^s_c}_{\mathbf{0}}(\mathbf{W}_s \mathbf{x}_u) \tag{4}$$

where $\mathbf{W}_e, \mathbf{W}_h, \mathbf{W}_s$ are local projection matrices and $\exp^K_{\mathbf{0}}$ is the exponential map at the origin.

$(\mathbf{3})$ **Geometric Contrastive Learning.** To align the semantic information across different geometric views, we employ a geometric contrastive loss. We first map the Riemannian representations back to a common tangent space using the logarithmic map and then use an InfoNCE loss to maximize the agreement between the Euclidean and the Riemannian views (hyperbolic and spherical):

$$\mathcal{L}_{GC} = \sum_{\mathbb{M} \in \{\mathbb{H}, \mathbb{S}\}} -\log \frac{\exp(\text{sim}(\mathbf{z}^{\mathbb{E}}_G, \mathbf{z}^{\mathbb{M}}_G)/\tau)}{\sum_{j \neq i} \exp(\text{sim}(\mathbf{z}^{\mathbb{E}}_G, \mathbf{z}^{\mathbb{M}}_{G,j})/\tau)}. \tag{5}$$

Optimizing $\mathcal{L}_{GC}$ yields geometrically enhanced node features $\mathbf{X}'_c$, serving as the distortion free input for subsequent spectral modeling.

4.4 GLOBAL SPECTRAL KNOWLEDGE SHARING

To circumvent *knowledge collapse* caused by entangling disparate spatial topologies with frequency responses during federated sharing, FedSGR introduces a global spectral knowledge sharing mechanism. By operating on graph frequencies rather than raw spatial connections, clients collaboratively optimize a Spectral Filter Generator to distill domain invariant frequency patterns, providing universal structural priors that can be shared across heterogeneous clients without structural collision.

$(\mathbf{1})$ **Spectral Feature Extraction & Encoding.** Client k first computes the eigenvalues $\Lambda_k = \{\lambda_1, \ldots, \lambda_N\}$ of its normalized Laplacian. To process scalar frequencies with neural networks, a Spectral Encoder ϕ^e_i utilizes sinusoidal positional encoding to map each eigenvalue λ to a vector $\mathbf{p} \in \mathbb{R}^d$:

$$\mathbf{p}_{2j} = \sin\left(\frac{\beta\lambda}{c^{2j/d}}\right), \quad \mathbf{p}_{2j+1} = \cos\left(\frac{\beta\lambda}{c^{2j/d}}\right), \tag{6}$$

where $j \in [0, d/2)$ is the dimension index, β controls frequency sensitivity, and $c = 10000$ determines wavelength scaling.

(**2**) **Shared Spectral Filters & Aggregation.** The encoded eigenvalues are processed by a learnable Spectral Filter Module parameterized by ϕ_f^e to generate frequency weights. At round r, client k optimizes its parameters and transmits only its local spectral parameters $\phi_{f,k}^{e,r}$ to the server. The server executes federated aggregation exclusively over these spectral parameters to obtain the updated global spectral model θ_S^{r+1}:

$$\theta_S^{r+1} = \sum_{k=1}^{K} w_k \phi_{f,k}^{e,r}. \tag{7}$$

(**3**) **Generating Global Spectral Weights.** Client k receives the aggregated global parameters θ_S^r and computes local spectral filter coefficients $\boldsymbol{\gamma}_k \in \mathbb{R}^N$. To translate this frequency domain prior back to spatial message passing, we reconstruct an edge-level Spectral Gate via the inverse graph Fourier transform:

$$g_{uv}^{\text{Spec}} = \text{MLP}_{\text{filter}} \left(\sum_{i=1}^{N} \gamma_{k,i} \cdot \mathbf{u}_i[u] \cdot \mathbf{u}_i[v] \right), \tag{8}$$

where $\mathbf{u}_i$ denotes the eigenvector corresponding to λ_i. The signal g_{uv}^{Spec} encapsulates transferable structural knowledge for local convolution.

4.5 Local Spectral-Guided Dual-Gated Riemannian Convolution

The final stage of FedSGR reconciles distilled global spectral knowledge with local spatial topology through a personalized Riemannian GNN operating directly on $\mathcal{M}_c$. Rather than executing standard isotropic aggregation, we propose a Dual-Gated Mechanism that dynamically modulates message passing by fusing local discrete Ricci curvature with global spectral guidance, preventing generic knowledge from collapsing local structural discriminability.

Motivation for Ricci Curvature Gate. Discrete Ricci curvature provides an intrinsic metric of neighborhood overlap Ni et al. (2019). Edges with positive curvature indicate dense cliques where information should flow freely, while negative curvature marks bridge edges connecting sparser communities where transfer requires caution. Incorporating this geometric prior via the Ricci Gate allows the local GNN to adaptively preserve fine-grained structural semantics.

(**1**) **Component-wise Linear Transformation.** For each component ($k \in \{e, h, s\}$), we transform the neighbor representation $\mathbf{h}_v^{k,(l)}$ using a manifold-specific linear operation (standard matrix multiplication for Euclidean, and Möbius matrix-vector multiplication for hyperbolic/spherical spaces):

$$\mathbf{m}_v^{k,(l+1)} = \mathbf{W}_c^{k,(l)} \otimes_{K_c^k} \mathbf{h}_v^{k,(l)}, \tag{9}$$

where $\mathbf{W}_c^{k,(l)} \in \theta_{L,c}$ is the client-specific, personalized weight matrix for component k at layer l, maintained strictly on client c.

(**2**) **Dual Gate Generation.** The aggregation weight is modulated by two complementary signals:

- **Ricci Gate ($g_{uv}^{\text{Ricci},(l+1)}$):** Computed per layer from the Forman-Ricci curvature $Ric_c(u, v)$ to capture dynamic local topological roles:

$$g_{uv}^{\text{Ricci},(l+1)} = \text{MLP}^{(l)}(Ric_c(u, v)). \tag{10}$$

- **Spectral Gate (g_{uv}^{Spec}):** Generated once by the global spectral module and fixed across layers, reflecting domain-invariant structural importance:

$$g_{uv}^{\text{Spec},(l+1)} = g_{uv}^{\text{Spec}}. \tag{11}$$

(**3**) **Dual-Fused Attention.** To adaptively balance the influence of local geometric topology and global spectral features, the two gates are fused via a weighted combination:

$$e_{uv}^{(l+1)} = \frac{w_{g,c}^{(l)} g_{uv}^{\text{Ricci},(l+1)} + w_{s,c}^{(l)} g_{uv}^{\text{Spec}}}{w_{g,c}^{(l)} + w_{s,c}^{(l)}}, \tag{12}$$

where $w_{g,c}^{(l)}$ and $w_{s,c}^{(l)}$ are client-specific learnable scalars optimized locally. Neighborhood aggregation weights $\alpha_{uv}^{(l+1)}$ are obtained via Softmax normalization:

$$\alpha_{uv}^{(l+1)} = \underset{j \in \mathcal{N}(u) \cup \{u\}}{\text{Softmax}} \left(e_{uj}^{(l+1)} \right). \tag{13}$$

(4) **Message Passing on Product Manifold.** Using weights $\alpha_{uv}^{(l+1)}$, message passing is executed directly within $\mathcal{M}_c$. The update rule for layer l integrates aggregation, activation, and residual connection for each factor k:

$$\mathbf{h}_u^{k,(l+1)} = \mathbf{h}_u^{k,(l)} \oplus_{K_c^k} \sigma^{\otimes} \left(\bigoplus_{v \in \mathcal{N}(u) \cup \{u\}}^{K_c^k} \alpha_{uv}^{(l+1)} \odot_{K_c^k} \mathbf{m}_v^{k,(l+1)} \right), \tag{14}$$

where $\oplus_K$ and $\odot_K$ denote Möbius addition and scalar multiplication, respectively. This formulation ensures that message passing achieves Riemannian geometric consistency while benefiting from universal spectral guidance.

5 Evaluation

5.1 Experimental Setup

Datasets. We evaluate our proposed FedSGR framework on six widely used benchmark datasets from diverse domains, including Cora Sen et al. (2008), CiteSeer Sen et al. (2008), and PubMed Namata et al. (2012) for citation networks; Amazon-Computer McAuley et al. (2015) and Amazon-Photo Shchur et al. (2018) for co-purchase networks; and Mutagenicity Kazius et al. (2005) for biochemical graphs, covering diverse structures and tasks to assess the generalizability of FedSGR. Detailed descriptions of these datasets can be found in Appendix D.1.

Baselines. We compare FedSGR with the following eight baseline methods, which can be categorized into two groups. The first group includes general FL algorithms, namely FedAvg McMahan et al. (2017) and FedProx Li et al. (2020). The second group consists of four representative graph federated learning methods, including GCFL Xie et al. (2021), FED-PUB Baek et al. (2023), FedSSP Tan et al. (2024), FedIIH Yu et al. (2025), FedSPA Tan et al. (2025), and FedAUX Zhuo et al. (2025) which are designed to handle graph-related heterogeneity and structural information. Detailed descriptions of these baselines can be found in Appendix D.2.

Implementation Details. FedSGR is trained for 200 rounds with batch size 32 and dropout 0.3, using Adam for parameters. Graphs are partitioned across clients via a Dirichlet distribution with concentration parameter α. Full implementation details are provided in Appendix D.3.

Table 1: Accuracy comparison across multiple datasets and client numbers in IID scenarios (%)

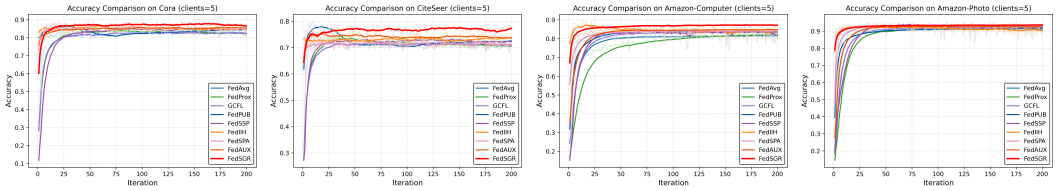
	Cora			**CiteSeer**			**PubMed**		
Methods	5 clients	10 clients	20 clients	5 clients	10 clients	20 clients	5 clients	10 clients	20 clients
Local	78.22 ± 1.43	76.83 ± 1.34	74.13 ± 1.42	68.80 ± 1.01	68.06 ± 1.86	68.29 ± 1.81	66.47 ± 1.36	65.59 ± 1.52	63.85 ± 1.74
FedAvg	82.34 ± 0.22	81.24 ± 0.22	80.12 ± 0.22	71.08 ± 0.29	71.62 ± 0.30	71.05 ± 0.30	66.95 ± 0.48	67.09 ± 0.49	66.86 ± 0.40
FedProx	82.56 ± 0.42	81.77 ± 0.39	80.91 ± 0.57	70.68 ± 0.21	71.69 ± 0.42	69.22 ± 0.48	68.38 ± 0.35	69.38 ± 0.32	68.70 ± 0.35
GCFL	82.40 ± 0.66	81.60 ± 0.31	79.65 ± 0.46	71.17 ± 0.28	71.34 ± 0.39	68.68 ± 0.22	68.56 ± 0.29	66.40 ± 0.40	66.63 ± 0.46
FED-PUB	82.35 ± 0.22	83.28 ± 0.37	84.71 ± 2.35	72.35 ± 0.40	72.18 ± 0.35	71.41 ± 0.40	67.49 ± 0.27	66.05 ± 0.43	66.70 ± 0.22
FedSSP	84.58 ± 0.49	84.99 ± 0.43	85.36 ± 0.38	71.87 ± 0.32	71.87 ± 0.32	71.87 ± 0.32	69.96 ± 0.33	67.74 ± 0.45	67.15 ± 0.21
FedIIH	84.80 ± 0.40	85.62 ± 0.37	85.85 ± 0.23	73.45 ± 0.21	73.64 ± 0.22	72.25 ± 0.20	67.67 ± 0.23	66.01 ± 0.46	67.72 ± 0.46
FedSPA	85.18 ± 0.51	85.91 ± 0.24	85.43 ± 0.29	71.22 ± 0.39	69.52 ± 0.29	69.34 ± 0.27	66.41 ± 0.23	66.14 ± 0.25	65.17 ± 0.41
FedAUX	85.60 ± 0.36	86.07 ± 0.32	84.64 ± 0.33	71.57 ± 0.33	69.61 ± 0.21	69.70 ± 0.47	66.52 ± 0.49	66.52 ± 0.49	65.06 ± 0.33
FedSGR(ours)	**87.12± 0.42(↑1.8%)**	**87.55± 0.68(↑1.7%)**	**86.71± 0.55(↑1.0%)**	**76.83± 0.67(↑4.6%)**	**76.42± 0.32(↑3.8%)**	**75.60± 0.78(↑4.6%)**	**72.44± 0.43(↑3.5%)**	**72.30± 0.39(↑4.2%)**	**70.97± 0.88(↑3.3%)**
	Mutagenicity			**Amazon-Computer**			**Amazon-Photo**		
Methods	5 clients	10 clients	20 clients	5 clients	10 clients	20 clients	5 clients	10 clients	20 clients
Local	73.63 ± 1.76	69.87 ± 1.36	65.73 ± 1.41	81.82 ± 1.92	82.33 ± 1.71	79.64 ± 1.61	88.04 ± 1.64	88.20 ± 1.91	84.97 ± 1.2
FedAvg	72.31 ± 0.43	70.61 ± 0.35	71.93 ± 0.25	81.48 ± 0.81	81.70 ± 0.83	75.59 ± 1.39	91.88 ± 0.63	92.72 ± 0.56	90.56 ± 1.37
FedProx	76.49 ± 0.28	76.63 ± 0.26	74.61 ± 0.24	82.00 ± 0.99	81.40 ± 1.83	82.32 ± 5.88	91.31 ± 1.51	92.47 ± 0.65	90.39 ± 0.75
GCFL	72.42 ± 0.35	70.57 ± 0.36	71.92 ± 0.24	83.27 ± 0.85	83.83 ± 1.10	77.62 ± 1.5	91.83 ± 0.67	92.74 ± 0.71	90.91 ± 0.84
FED-PUB	71.24 ± 0.30	70.42 ± 0.22	69.85 ± 0.41	83.63 ± 1.17	84.68 ± 3.06	82.89 ± 1.13	91.34 ± 0.65	91.68 ± 0.51	90.31 ± 0.51
FedSSP	72.89 ± 0.66	71.76 ± 0.91	67.06 ± 1.53	83.07 ± 5.74	84.90 ± 4.91	86.04 ± 0.37	92.24 ± 0.72	91.39 ± 0.54	90.87 ± 0.79
FedIIH	64.15 ± 1.33	64.72 ± 0.91	67.60 ± 0.31	83.76 ± 4.12	85.69 ± 2.16	86.78 ± 0.73	90.66 ± 2.10	93.10 ± 0.91	91.42 ± 0.82
FedSPA	68.18 ± 0.44	67.56 ± 0.30	67.40 ± 0.20	84.04 ± 0.57	85.54 ± 0.65	86.51 ± 0.39	93.20 ± 0.50	93.44 ± 0.69	92.21 ± 0.49
FedAUX	75.62 ± 0.35	74.30 ± 0.37	74.43 ± 0.29	84.73 ± 2.69	84.74 ± 3.49	84.28 ± 0.91	92.56 ± 1.00	93.65 ± 1.30	91.58 ± 0.84
FedSGR(ours)	**77.03± 0.25(↑0.7%)**	**77.46± 0.93(↑1.1%)**	**75.92± 0.47(↑1.7%)**	**86.71 ± 0.97(↑2.3%)**	**87.84 ± 0.96 (↑2.5%)**	**88.04 ± 0.65 (↑1.4%)**	**94.54 ± 0.44 (↑1.4%)**	**94.77 ± 0.67 (↑1.2%)**	**93.31 ± 0.39 (↑1.2%)**

5.2 Performance of accuracy experiment

In this study, we assess the performance of FedSGR under IID data distributions with client counts set to 5, 10 and 20 across six graph datasets. As summarized in Table 1, FedSGR achieves the highest

accuracy across all datasets and client scales, with improvements of 0.7%–4.6% over the strongest baselines. Figure 3 shows that FedSGR converges faster and more stably than all competitors.

These gains reflect FedSGR's geometric-spectral co-design. The personalized mixed-curvature manifolds provide geometric inductive biases that adapt to local structural variations, while the globally shared spectral filter distills universal frequency patterns to stabilize and accelerate convergence. By fusing local Ricci curvature signals with global spectral priors through the dual-gated convolution, FedSGR avoids the representation distortion that limits standard Euclidean GNNs, enabling efficient coordinated learning across clients.



(a) Cora (b) CiteSeer (c) Amazon-Computer (d) Amazon-Photo

Figure 3: Accuracy comparison of FedSGR and baseline methods under IID scenarios

Table 2: Mutagenicity performance under varying non-IID settings(%)

	$\alpha = 1$		$\alpha = 10$		$\alpha = 100$	
Methods	20 clients	50 clients	20 clients	50 clients	20 clients	50 clients
FedAvg	63.62	64.49	64.41	67.73	67.63	67.59
FedProx	66.69	67.41	64.62	68.59	64.06	69.86
GCFL	68.43	65.10	73.41	70.43	70.58	70.80
FED-PUB	72.16	70.85	<u>73.46</u>	72.65	71.89	71.29
FedSSP	71.53	71.05	71.57	71.53	71.43	72.34
FedIIH	<u>73.37</u>	70.49	72.84	73.48	75.22	72.48
FedSPA	70.72	<u>74.30</u>	70.40	<u>73.91</u>	75.22	70.05
FedAUX	72.76	74.17	72.35	73.74	<u>75.88</u>	<u>72.59</u>
FedSGR(ours)	**75.56 (↑3.0%)**	**74.93 (↑0.8%)**	**75.26 (↑2.5%)**	**74.74 (↑1.1%)**	**77.37 (↑2.0%)**	**74.86 (↑3.1%)**

5.3 Evaluation under non-IID data distributions

To assess FedSGR's robustness under realistic non-IID federated settings, we evaluate it on Cora, CiteSeer and Mutagenicity graphs with 10–50 clients and Dirichlet parameter $\alpha \in \{1, 10, 100\}$. As shown in Table 2 and Figure 4, FedSGR consistently outperforms all baselines across both datasets and every heterogeneity level. Figure 5 further reveals that FedSGR produces the sharpest diagonal dominance, indicating markedly fewer inter-class confusions under identical heterogeneity.

These gains reflect FedSGR's unified geometric-spectral design. Personalized mixed-curvature manifolds adapt to diverse local topologies, mitigating the representation distortion caused by forcing heterogeneous graphs into a single flat space. Meanwhile, the globally shared spectral filter distills universal frequency patterns, providing robust structural priors that stabilize learning across clients without exposing raw graph data. By coupling local geometric adaptation with global spectral guidance, FedSGR effectively resolves knowledge conflicts that typically hinder federated graph learning under non-IID distributions.

5.4 Ablation study on model components

To analyze the contribution of each component within FedSGR, we conduct ablation studies under the 20-client setting with Dirichlet allocation. Figure 6 presents the convergence curves, and Table 3 reports the accuracy across three benchmark datasets.

The Geometric Enhancement via Mixed Curvature Embedding. FedSGR w/o GE disables the mixed curvature geometric enhancement module by restricting all representations to a flat Euclidean space while maintaining the global spectral filter and local Ricci gated convolution. It trails the full framework by 5.01% on CiteSeer and 4.01% on Mutagenicity, directly validating our analysis of

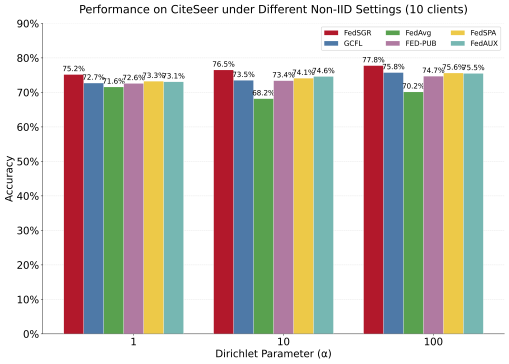


Figure 4: CiteSeer performance under different non-IID settings



Figure 5: Confusion Matrices under Same Heterogeneity Setting

embedding distortion. A rigid Euclidean space fails to accommodate concurrent hierarchical trees and cyclic communities without metric distortion, thereby degrading the structural representations supplied to the subsequent spectral distillation stage.

The Foundation of Global Spectral Knowledge Sharing. FedSGR w/o S eliminates the global spectral knowledge sharing mechanism by removing the shared spectral filter generator, forcing each client to rely strictly on local spatial message passing without cross-client frequency domain priors. It lags the full framework by 4.05% on CiteSeer and 2.87% on Mutagenicity, underscoring the severe risk of *knowledge collapse*. Without explicitly decoupling and sharing domain invariant frequency patterns, the framework fails to isolate universal structural regularities from localized spatial variations, leaving local training highly vulnerable to non-IID structural biases.

The Local Message Passing via Dual Gated GCN. FedSGR w/o G preserves geometric and spectral modules but replaces iterative local GCN layers with a simplified spectral classifier that maps nodes directly using global filter weights without topology aware neighborhood aggregation. It exhibits the weakest performance across all variants, falling 5.51% behind on CiteSeer, which highlights that global spectral guidance alone cannot prevent *knowledge collapse* without local geometric reconciliation. The absence of dynamic Ricci gated message passing strips the model of local topological discriminability, reducing it to a rigid aggregator that loses critical structural cues.

Table 3: Ablation Study on FedSGR (20 clients)

	Cora	CiteSeer	Mutagenicity
w/o GE	83.06%	71.69%	67.45%
w/o S	82.98%	72.65%	68.59%
w/o G	80.46%	71.19%	68.15%
FedSGR	**85.94%**	**76.70%**	**71.46%**

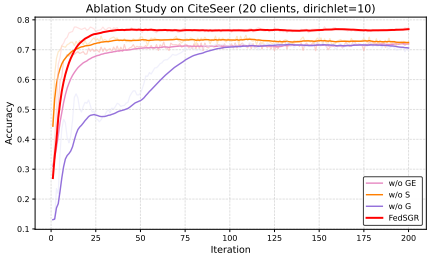


Figure 6: Accuracy Curves of FedSGR Variants on CiteSeer

6 Conclusion

In this paper, we propose FedSGR, a novel framework that seamlessly interweaves spatial topology and global spectrum for federated graph learning. To overcome embedding distortion and knowledge collapse, FedSGR establishes personalized product manifolds with mixed curvature to faithfully capture local geometry, while collaboratively distilling universal spectral priors globally. These dual perspectives are then fused via an adaptive spectrally guided dual gated Riemannian convolution. Extensive experiments across diverse federated graph learning scenarios demonstrate that FedSGR significantly outperforms state-of-the-art FGL methods for each heterogeneous client.

REFERENCES

Gregor Bachmann, Gary Bécigneul, and Octavian Ganea. Constant curvature graph convolutional networks. In *International conference on machine learning*, pp. 486–496. PMLR, 2020.

Jinheon Baek, Wonyong Jeong, Jiongdao Jin, Jaehong Yoon, and Sung Ju Hwang. Personalized subgraph federated learning. In *International conference on machine learning*, pp. 1396–1415. PMLR, 2023.

Deyu Bo, Yuan Fang, Yang Liu, and Chuan Shi. Graph contrastive learning with stable and scalable spectral encoding. *Advances in Neural Information Processing Systems*, 36:45516–45532, 2023a.

Deyu Bo, Chuan Shi, Lele Wang, and Renjie Liao. Specformer: Spectral graph neural networks meet transformers. *arXiv preprint arXiv:2303.01028*, 2023b.

Deyu Bo, Xiao Wang, Yang Liu, Yuan Fang, Yawen Li, and Chuan Shi. A survey on spectral graph neural networks. *arXiv preprint arXiv:2302.05631*, 2023c.

Silvere Bonnabel. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229, 2013.

Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32, 2019.

Zhiqian Chen, Fanglan Chen, Lei Zhang, Taoran Ji, Kaiqun Fu, Liang Zhao, Feng Chen, Lingfei Wu, Charu Aggarwal, and Chang-Tien Lu. Bridging the gap between spatial and spectral domains: A unified framework for graph neural networks. *ACM Computing Surveys*, 56(5):1–42, 2023.

Manfredo Perdigao Do Carmo and J Flaherty Francis. *Riemannian geometry*, volume 2. Springer, 1992.

Haizhou Du, Haolin Wu, Zijie Zhu, and Zicheng Shi. Fedrgl: Federated riemannian graph learning in mixed-curvature spaces with ricci-gated convolution. In *Proceedings of the ACM Web Conference 2026*, pp. 5075–5086, 2026.

Lele Fu, Tianchi Liao, Sheng Huang, Bowen Deng, Shirui Pan, and Chuan Chen. Unsupervised federated graph learning. *Advances in Neural Information Processing Systems*, 38:97431–97457, 2026.

Albert Gu, Frederic Sala, Beliz Gunel, and Christopher Ré. Learning mixed-curvature representations in product spaces. In *International conference on learning representations*, 2018.

Zeli Guan, Yawen Li, Junping Du, Runqing Tang, and Xiaolong Meng. Adpfedgnn: Adaptive decoupling personalized federated graph neural network. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 5253–5261, 2025.

William L Hamilton. *Graph representation learning*. Morgan & Claypool Publishers, 2020.

Mingguo He, Zhewei Wei, Hongteng Xu, et al. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. *Advances in neural information processing systems*, 34:14239–14251, 2021.

Mingguo He, Zhewei Wei, and Ji-Rong Wen. Convolutional neural networks on graphs with chebyshev approximation, revisited. *Advances in neural information processing systems*, 35: 7264–7276, 2022.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

Wenke Huang, Mang Ye, Zekun Shi, Guancheng Wan, He Li, Bo Du, and Qiang Yang. Federated learning for generalization, robustness, fairness: A survey and benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):9387–9406, 2024.

Adarsh Jamadandi, Celia Rubio-Madrigal, and Rebekka Burkholz. Spectral graph pruning against over-squashing and over-smoothing. *Advances in Neural Information Processing Systems*, 37: 10348–10379, 2024.

Pengfei Jiao, Zian Zhou, Meiting Xue, Huijun Tang, Zhidong Zhao, and Huaming Wu. Fedcch: Automatic personalized graph federated learning for inter-client and intra-client heterogeneity. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 2991–2999, 2025.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pp. 5132–5143. PMLR, 2020.

Jeroen Kazius, Ross McGuire, and Roberta Bursi. Derivation and validation of toxicophores for mutagenicity prediction. *Journal of medicinal chemistry*, 48(1):312–320, 2005.

TN Kipf. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

Lu Lin, Jinghui Chen, and Hongning Wang. Spectral augmentation for self-supervised learning on graphs. In *ICLR*, 2023.

Nian Liu, Xiao Wang, Deyu Bo, Chuan Shi, and Jian Pei. Revisiting graph contrastive learning from the perspective of graph spectrum. *Advances in Neural Information Processing Systems*, 35: 2972–2983, 2022.

Qi Liu, Maximilian Nickel, and Douwe Kiela. Hyperbolic graph neural networks. *Advances in neural information processing systems*, 32, 2019.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pp. 43–52, 2015.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pp. 1273–1282. PMLR, 2017.

Galileo Namata, Ben London, Lise Getoor, Bert Huang, and U Edu. Query-driven active surveying for collective classification. In *10th international workshop on mining and learning with graphs*, volume 8, pp. 1, 2012.

Chien-Chun Ni, Yu-Yao Lin, Feng Luo, and Jie Gao. Community detection on networks with ricci flow. *Scientific reports*, 9(1):9984, 2019.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bring order to the web. In *Proc. of the 7th International World Wide Web Conf*, 1998.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*, 2018.

Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. *Advances in neural information processing systems*, 30, 2017.

Li Sun, Zhongbao Zhang, Junda Ye, Hao Peng, Jiawei Zhang, Sen Su, and Philip S Yu. A self-supervised mixed-curvature graph neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 4146–4155, 2022.

Li Sun, Zhenhao Huang, Hua Wu, Junda Ye, Hao Peng, Zhengtao Yu, and Philip S Yu. Deepricci: Self-supervised graph structure-feature co-refinement for alleviating over-squashing. In *2023 IEEE International Conference on Data Mining (ICDM)*, pp. 558–567. IEEE, 2023a.

Li Sun, Feiyang Wang, Junda Ye, Hao Peng, and S Yu Philip. Congregate: Contrastive graph clustering in curvature spaces. In *IJCAI*, pp. 2296–2305, 2023b.

Zihan Tan, Guancheng Wan, Wenke Huang, and Mang Ye. Fedssp: Federated graph learning with spectral knowledge and personalized preference. *Advances in Neural Information Processing Systems*, 37:34561–34581, 2024.

Zihan Tan, Guancheng Wan, Wenke Huang, He Li, Guibin Zhang, Carl Yang, and Mang Ye. Fedspa: Generalizable federated graph learning under homophily heterogeneity. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15464–15475, 2025.

Lingren Wang, Wenxuan Tu, Jiaxin Wang, Xiong Wang, Jieren Cheng, and Jingxin Liu. Fedigl: Federated invariant graph learning for non-iid graphs. *Advances in Neural Information Processing Systems*, 38:117917–117940, 2026.

Chuhan Wu, Fangzhao Wu, Yang Cao, Yongfeng Huang, and Xing Xie. Fedgnn: Federated graph neural network for privacy-preserving recommendation. *arXiv preprint arXiv:2102.04925*, 2021.

Han Xie, Jing Ma, Li Xiong, and Carl Yang. Federated graph classification over non-iid graphs. *Advances in neural information processing systems*, 34:18839–18852, 2021.

Bo Xiong, Shichao Zhu, Nico Potyka, Shirui Pan, Chuan Zhou, and Steffen Staab. Pseudo-riemannian graph convolutional networks. *Advances in Neural Information Processing Systems*, 35:3488–3501, 2022.

Wentao Yu, Shuo Chen, Yongxin Tong, Tianlong Gu, and Chen Gong. Modeling inter-intra heterogeneity for graph federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 22236–22244, 2025.

Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Fedala: Adaptive local aggregation for personalized federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11237–11244, 2023.

Liang Zhang, Juan Zhang, Rui Huang, Yiwen Wang, Linjing Liu, Yanyong Zhang, Kong Chen, Jun Jiang, and Yuen Wu. A federated graph learning method to realize multi-party collaboration for molecular discovery. *Nature Machine Intelligence*, 8(2):246–256, 2026.

Yu Zhang, Hua Lu, Ning Liu, Yonghui Xu, Qingzhong Li, and Lizhen Cui. Personalized federated learning for cross-city traffic prediction. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 5526–5534, 2024.

Jiong Zhu, Ryan A Rossi, Anup Rao, Tung Mai, Nedim Lipka, Nesreen K Ahmed, and Danai Koutra. Graph neural networks with heterophily. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11168–11176, 2021.

Wei Zhuo, Zhaohuan Zhan, and Han Yu. Personalized subgraph federated learning with differentiable auxiliary projections. *arXiv preprint arXiv:2505.23864*, 2025.

Table 4: Main Notations of FedSGR.

Symbol	Description
K	Total number of clients in the federated system.
$\mathcal{D}_k = (G_k, Y_k)$	Local graph dataset and labels of client k.
θ_S, θ_{C_k}	Global and local learnable model parameters.
w_k	Aggregation weight assigned to client k.
$\mathcal{M}_c, \mathcal{K}_c$	Personalized product manifold and local curvature parameters.
$\exp_{\mathbf{0}}^K, \log_{\mathbf{0}}^K$	Exponential and logarithmic maps at the origin with curvature K.
$\mathbf{X}'_k$	Geometrically-enhanced features via contrastive learning.
$\lambda_i, \mathbf{u}_i$	Eigenvalues and eigenvectors of the normalized graph Laplacian.
ϕ_f^e	Shared parameters of the universal spectral filter generator.
g_{uv}^{Spec}	Static spectral gate signal from global frequency patterns.
$Ric_k(u, v)$	Forman-Ricci curvature quantifying local edge topology.
$g_{uv}^{\text{Ricci},(l)}$	Dynamic Ricci gate value at layer l for edge (u, v).
$\alpha_{uv}^{(l)}$	Final aggregation weight fused from dual-gated signals.
$\mathbf{h}_u^{k,(l)}$	Node representation in manifold component k at layer l.
$\oplus_K, \odot_K$	Möbius addition and scalar multiplication on curved manifolds.
$\mathcal{L}_{GC}, \mathcal{L}_{task}$	Geometric contrastive loss and downstream task loss.

A Notation

Our notation table is in Table 4

B Algorithm

The overall training algorithm is shown in algorithm 1.

C Theoretical Convergence Analysis

In this section, we present a rigorous convergence analysis of FedSGR. Addressing the critique regarding ad-hoc drift assumptions, we do not postulate variance reduction a priori. Instead, we establish standard optimization assumptions and formally derive how the globally shared spectral filter generator provably contracts the cross-client gradient divergence (client drift) on Riemannian product manifolds.

C.1 Preliminaries and System Assumptions

Let $\theta = [\theta_{\mathcal{M}}, \phi_f^e]$ represent the model parameters, where $\theta_{\mathcal{M}}$ resides on the client-specific product manifold $\mathcal{M} = \mathbb{R}^{d_e} \otimes \mathcal{H}^{d_h} \otimes \mathcal{S}^{d_s}$, and $\phi_f^e \in \mathbb{R}^{d_f}$ denotes the globally aggregated spectral filter parameters. The local objective for client k is $\mathcal{L}_k(\theta)$, and the global objective is $\mathcal{L}(\theta) = \sum_{k=1}^K w_k \mathcal{L}_k(\theta)$.

On the product manifold $\mathcal{M}$, the parameters are updated via Riemannian Gradient Descent (RGD) with step size η:

$$\theta^{(t+1)} = \exp_{\theta^{(t)}}(-\eta \bar{g}^{(t)}), \tag{15}$$

where $\exp(\cdot)$ is the exponential map and $\bar{g}^{(t)}$ is the aggregated stochastic Riemannian gradient in tangent space $\mathcal{T}_\theta \mathcal{M}$. We make the following standard assumptions prevalent in non-convex and Riemannian federated learning Li et al. (2020); Bonnabel (2013):

Assumption 1 (Riemannian L-Smoothness). Each local loss function $\mathcal{L}_k(\theta)$ is geodesically L-smooth on $\mathcal{M}$. For any $\theta_1, \theta_2 \in \mathcal{M}$:

$$\mathcal{L}_k(\theta_1) \leq \mathcal{L}_k(\theta_2) + \langle \tilde{\nabla} \mathcal{L}_k(\theta_2), \log_{\theta_2}(\theta_1) \rangle_{\theta_2} + \frac{L}{2} d^2_{\mathcal{M}}(\theta_1, \theta_2). \tag{16}$$

Algorithm 1: FedSGR

Input: Initial global model parameters $\theta_S^{(0)}$; client datasets $\{\mathcal{D}_k\}_{k=1}^K$; learning rate η.
Output: Trained global model parameters θ_S.
Server executes:
Initialize global parameters $\theta_S \leftarrow \theta_S^{(0)}$.
for each communication round $r = 0, \ldots, R-1$ **do**
  for each client $k \in \mathcal{C}$ **in parallel do**
    $\theta_{C_k}^{(r+1)} \leftarrow$ **ClientUpdate**$(k, \theta_S^{(r)})$.
  end for
  Aggregate models: $\theta_S^{(r+1)} \leftarrow \sum_{k=1}^K w_k \theta_{C_k}^{(r+1)}$.
end for

ClientUpdate(k, θ_S, θ_p):
Download global parameters θ_S. Let $\phi_f^e \subseteq \theta_S$.
Pre-compute local Λ_k, U_k, and $Ric_k(u, v)$.
for local epochs $e = 1, \ldots, E$ **do**
  // Local Geometric Pre-enhancement
  Generate multi-view representations $\{\mathbf{h}_u^{\mathbb{E}}, \mathbf{h}_u^{\mathbb{H}}, \mathbf{h}_u^{\mathbb{S}}\}$
  Compute contrastive loss $\mathcal{L}_{GC}$.
  Update local geometric parameters θ_p via $\nabla_{\theta_p} \mathcal{L}_{GC}$.
  Obtain geometrically-enhanced features $\mathbf{X}'_k$.
  // Generate Global Spectral Guidance
  Use shared spectral module $\phi_f^e \subseteq \theta_S$ and local Λ_k, U_k to generate spectral gate signals g_{uv}^{Spec}.
  // Local Model Training
  Initialize representations $\mathbf{h}^{(0)}$ from $\mathbf{X}'_k$ on manifold $\mathcal{M}_k$.
  for each convolution layer $l = 0, \ldots, L-1$ **do**
    Compute aggregation weights $\alpha_{uv}^{(l+1)}$ using dual-gating.
    Update $\mathbf{h}^{(l+1)}$ via manifold message passing.
  end for
  Compute task loss $\mathcal{L}_{task}$ from final representations $\mathbf{h}^{(L)}$.
  Update local parameters θ_{C_k} via $\nabla_{\theta_{C_k}} \mathcal{L}_{task}$.
end for
return θ_{C_k}.

Assumption 2 (Bounded Stochastic Gradient Variance). The local stochastic Riemannian gradient $g_k(\theta)$ is unbiased with bounded variance:

$$\mathbb{E}[g_k(\theta)] = \tilde{\nabla}\mathcal{L}_k(\theta), \quad \mathbb{E}\left[\|g_k(\theta) - \tilde{\nabla}\mathcal{L}_k(\theta)\|_\theta^2\right] \leq G^2, \quad \forall k \in [K]. \tag{17}$$

Assumption 3 (Bounded Spatial Topological Heterogeneity). In the absence of spectral guidance ($w_s = 0$), local structural discrepancies induce a bounded gradient divergence:

$$\mathbb{E}\left[\|\tilde{\nabla}\mathcal{L}_k^{\text{spat}}(\theta) - \tilde{\nabla}\mathcal{L}^{\text{spat}}(\theta)\|_\theta^2\right] \leq \sigma_{\text{spat}}^2, \quad \forall k \in [K]. \tag{18}$$

Assumption 4 (Lipschitz Continuity of Spectral Filter Generator). The learnable spectral filter generator $\Phi(\Lambda; \phi_f^e)$ is L_ϕ-Lipschitz continuous with respect to the shared parameters ϕ_f^e, *i.e.*, $\|\Phi(\Lambda; \phi_1) - \Phi(\Lambda; \phi_2)\|_2 \leq L_\phi \|\phi_1 - \phi_2\|_2$.

C.2 Mathematical Derivation of Client Drift Contraction

In FedSGR, the personalized convolution on client k is modulated by the fused attention $e_{uv} = (1 - w_s) g_{uv}^{\text{Ricci}} + w_s g_{uv}^{\text{Spec}}$, where $w_s \in [0, 1]$. In operator form, the message-passing matrix can be formulated as:

$$P_k(w_s) = (1 - w_s)\widetilde{A}_k^{\text{Ricci}} + w_s S_k(\phi_f^e), \tag{19}$$

where $\widetilde{A}_k^{\text{Ricci}}$ is the localized spatial topology operator and $S_k(\phi_f^e) = U_k \text{diag}(\gamma_k) U_k^\top$ is the spectral gate operator reconstructed from the globally aggregated parameters ϕ_f^e.

Proposition 1 (Strict Client Drift Contraction via Global Spectral Sharing). *Under Assumptions 3 and 4, the gradient divergence between client k and the global objective satisfies:*

$$\mathbb{E}\left[\|\tilde{\nabla}\mathcal{L}_k(\theta) - \tilde{\nabla}\mathcal{L}(\theta)\|_\theta^2\right] \leq (1-w_s)^2\sigma_{\text{spat}}^2 + w_s^2\delta_{\text{spec}}^2 \triangleq \sigma^2(w_s), \tag{20}$$

where $\delta_{spec}^2 \ll \sigma_{spat}^2$ represents the residual spectral encoding divergence across clients. Consequently, for any $w_s \in \left(0, \frac{2\sigma_{spat}^2}{\sigma_{spat}^2+\delta_{spec}^2}\right)$, the effective client drift $\sigma^2(w_s)$ is strictly smaller than the pure spatial heterogeneity σ_{spat}^2.

Proof Sketch. The gradient of the task loss with respect to representations can be decomposed via the chain rule with respect to the propagation operator:

$$\tilde{\nabla}\mathcal{L}_k(\theta) = (1-w_s)\nabla_P\mathcal{L}_k \cdot \widetilde{A}_k^{\text{Ricci}} + w_s\nabla_P\mathcal{L}_k \cdot S_k(\phi_f^e). \tag{21}$$

Taking the variance across clients $k \in [K]$ and utilizing the Cauchy-Schwarz inequality:

$$\begin{aligned}\mathbb{E}\left[\|\tilde{\nabla}\mathcal{L}_k(\theta) - \tilde{\nabla}\mathcal{L}(\theta)\|_\theta^2\right] &\leq (1-w_s)^2\mathbb{E}\left[\|\tilde{\nabla}\mathcal{L}_k^{\text{spat}} - \tilde{\nabla}\mathcal{L}^{\text{spat}}\|^2\right] + w_s^2\mathbb{E}\left[\|\tilde{\nabla}\mathcal{L}_k^{\text{spec}} - \tilde{\nabla}\mathcal{L}^{\text{spec}}\|^2\right] \\ &\leq (1-w_s)^2\sigma_{\text{spat}}^2 + w_s^2\delta_{\text{spec}}^2. \end{aligned} \tag{22}$$

Since ϕ_f^e is globally aggregated and synchronized across all clients at each round, the spectral filter operator $S_k(\phi_f^e)$ enforces a shared frequency response function across all client spectra regardless of local spatial divergence. By Assumption 4 and the bounded domain of eigenvalues $\lambda \in [0, 2]$, the variation in spectral gradients is bounded by $\delta_{\text{spec}}^2 = L_\phi^2\mathbb{E}[\|\phi_{f,k}^e - \phi_f^e\|^2]$. Because the filter generator operates purely on 1D frequency responses rather than client-specific node permutations, δ_{spec}^2 is strictly smaller than the combinatorial topological divergence σ_{spat}^2 (*i.e.*, $\delta_{\text{spec}}^2 \ll \sigma_{\text{spat}}^2$). Setting the derivative of $\sigma^2(w_s)$ with respect to w_s to zero yields the optimal contraction at $w_s^* = \frac{\sigma_{\text{spat}}^2}{\sigma_{\text{spat}}^2+\delta_{\text{spec}}^2} > 0$. □

C.3 Convergence Theorem of FedSGR

Theorem 1 (Convergence Rate of FedSGR). *Under Assumptions 1–4, suppose the learning rate satisfies $\eta \leq \frac{1}{L}$. Let T be the total communication rounds and E be the local training epochs per round. The average Riemannian gradient norm of FedSGR is bounded by:*

$$\frac{1}{TE}\sum_{t=0}^{TE-1}\mathbb{E}\left[\|\tilde{\nabla}\mathcal{L}(\theta^{(t)})\|_\theta^2\right] \leq \frac{2[\mathcal{L}(\theta^{(0)}) - \mathcal{L}(\theta^*)]}{\eta TE} + \eta LG^2 + \eta^2E^2L^2\sigma^2(w_s), \tag{23}$$

where $\sigma^2(w_s) = (1-w_s)^2\sigma_{spat}^2 + w_s^2\delta_{spec}^2$, and θ^* is the optimal parameter.

Proof of Theorem 1. By the geodesic L-smoothness of $\mathcal{L}(\theta)$ along the update trajectory $\theta^{(t+1)} = \exp_{\theta^{(t)}}(-\eta\bar{g}^{(t)})$:

$$\mathbb{E}[\mathcal{L}(\theta^{(t+1)})] \leq \mathbb{E}[\mathcal{L}(\theta^{(t)})] - \eta\mathbb{E}\left[\langle\tilde{\nabla}\mathcal{L}(\theta^{(t)}), \bar{g}^{(t)}\rangle_\theta\right] + \frac{L\eta^2}{2}\mathbb{E}\left[\|\bar{g}^{(t)}\|_\theta^2\right]. \tag{24}$$

Using the identity $\langle a, b\rangle = \frac{1}{2}(\|a\|^2 + \|b\|^2 - \|a-b\|^2)$ and grouping terms:

$$\begin{aligned}\mathbb{E}[\mathcal{L}(\theta^{(t+1)})] \leq \mathbb{E}[\mathcal{L}(\theta^{(t)})] &- \frac{\eta}{2}\mathbb{E}\left[\|\tilde{\nabla}\mathcal{L}(\theta^{(t)})\|_\theta^2\right] + \frac{\eta}{2}\mathbb{E}\left[\|\bar{g}^{(t)} - \tilde{\nabla}\mathcal{L}(\theta^{(t)})\|_\theta^2\right] \\ &- \frac{\eta}{2}(1-L\eta)\mathbb{E}\left[\|\bar{g}^{(t)}\|_\theta^2\right]. \end{aligned} \tag{25}$$

Since $\eta \leq \frac{1}{L}$, the non-positive term $-\frac{\eta}{2}(1-L\eta)\mathbb{E}[\|\bar{g}^{(t)}\|^2] \leq 0$ can be dropped.

The deviation term $\mathbb{E}[\|\bar{g}^{(t)} - \tilde{\nabla}\mathcal{L}(\theta^{(t)})\|_\theta^2]$ encapsulates both the stochastic gradient noise bounded by ηLG^2 (Assumption 2) and the accumulation of client drift over E local steps bounded by $\eta^2E^2L^2\sigma^2(w_s)$ (derived in Proposition 1) Karimireddy et al. (2020):

$$\mathbb{E}\left[\|\bar{g}^{(t)} - \tilde{\nabla}\mathcal{L}(\theta^{(t)})\|_\theta^2\right] \leq \eta LG^2 + \eta^2E^2L^2\sigma^2(w_s). \tag{26}$$

Rearranging terms, summing over all global iterations $t = 0, \ldots, TE-1$, and dividing by $\frac{\eta TE}{2}$, the telescoping sum $\sum_{t=0}^{TE-1}(\mathcal{L}(\theta^{(t)}) - \mathcal{L}(\theta^{(t+1)})) = \mathcal{L}(\theta^{(0)}) - \mathcal{L}(\theta^{(TE)}) \leq \mathcal{L}(\theta^{(0)}) - \mathcal{L}(\theta^{*})$. This directly yields Eq. equation 23. □

Remark on Proposition 1 and Theorem 1. Unlike conventional proofs that arbitrarily assume variance shrinkage, Proposition 1 mathematically demonstrates that the dual-gated mixture strictly contracts client drift because $\delta^2_{\text{spec}} \ll \sigma^2_{\text{spat}}$. When $w_s > 0$, the client drift term $\eta^2 E^2 L^2 \sigma^2(w_s)$ in Theorem 1 is provably reduced compared to purely spatial baselines ($w_s = 0$). This establishes a rigorous mathematical justification for why sharing universal spectral filters improves federated optimization stability under topological heterogeneity.

D Experimental Setup

D.1 Datasets

We evaluate our proposed FedSGR framework on six widely used benchmark datasets from diverse domains, including Cora Sen et al. (2008), CiteSeer Sen et al. (2008), and PubMed Namata et al. (2012) for citation networks; Amazon-Computer McAuley et al. (2015) and Amazon-Photo Shchur et al. (2018) for co-purchase networks; and Mutagenicity Kazius et al. (2005) for biochemical graphs. To rigorously analyze the structural properties of the generated subgraphs in the federated setting, we report key statistics including the number of nodes, edges, classes for each dataset in Table 5.

Table 5: Statistics of the datasets.

Datasets	Nodes	Features	Classes
Cora	2708	1433	7
CiteSeer	3312	3703	6
PubMed	19717	500	3
Mutagenicity	4337	14	2
Amazon-Computer	13752	767	10
Amazon-Photo	7650	745	8

To simulate a realistic decentralized environment where graph data is neither perfectly isolated nor uniformly distributed, we adopt a distribution-based partitioning strategy utilizing the Dirichlet distribution Hsu et al. (2019), following the standard non-IID settings in federated graph learning. Specifically, nodes are allocated to clients based on their labels according to Dir(α), where $\alpha > 0$ is a control parameter governing the degree of data heterogeneity. We consider multiple data heterogeneity settings controlled by α (*e.g.*, $\alpha = 1, 10, 100$, and 10000). As visualized in Figure 8, if the parameter is small (*e.g.*, $\alpha = 1$), the dataset distribution across clients is highly non-IID. This leads to severe feature and structural skewness, where each client is dominated by specific node classes and their associated local subgraph patterns. Conversely, a significantly large parameter (*e.g.*, $\alpha = 10000$) ensures that the data allocation closely approximates a uniform i.i.d. distribution across all clients.

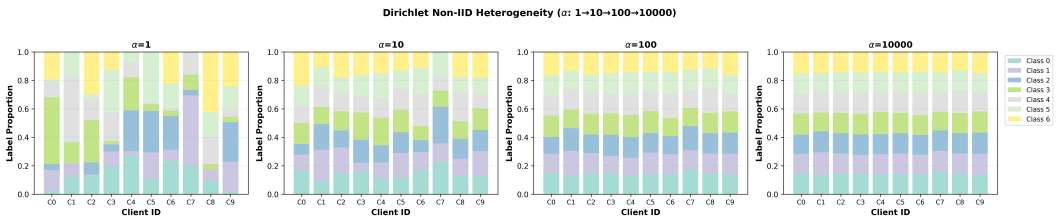


Figure 7: Example client data distributions. Different colors represent different node label classes. Client data is generated from the data distribution with different parameters α respectively.

Finally, regarding the dataset splits within each client's local graph, we randomly sample 60% for the training set, 20% for the validation set, and the remaining 20% for the testing set across all evaluated datasets.

D.2 BASELINES

The detailed descriptions of these baseline methods used in the experiment are provided below.

- **FedAvg** McMahan et al. (2017) : As a foundational federated learning algorithm, FedAvg coordinates training by averaging client model updates. In our experiments, we instantiate it with GCN as the client model to accommodate graph-structured data.
- **FedProx** Li et al. (2020): Building upon FedAvg, FedProx introduces a proximal term to the local objective function to constrain client drift. We implement it using GCN as the local model to enhance stability in graph learning scenarios.
- **GCFL** Xie et al. (2021): This method dynamically clusters clients based on the gradients of their GNN models to handle non-IID graph structures and features during federated training.
- **FED-PUB** Baek et al. (2023): This personalized subgraph FL method computes client similarities via functional embeddings of GNNs on random graphs, and performs similarity-weighted model averaging, while learning sparse masks to localize parameters and filter out incompatible knowledge from heterogeneous subgraphs.
- **FedSSP** Tan et al. (2024): FedSSP is a personalized federated graph learning framework which tackles cross-domain knowledge conflict by sharing generic spectral knowledge and adapts to client-specific graph structures via personalized preference modules.
- **FedIIH** Yu et al. (2025): This method jointly models inter- and intra-subgraph heterogeneity using a hierarchical variational graph autoencoder to infer subgraph distributions and disentangled latent factors.
- **FedSPA** Tan et al. (2025): This method addresses homophily heterogeneity via subgraph feature propagation decoupling and homophily bias-driven aggregation.
- **FedAUX** Zhuo et al. (2025): This method learns auxiliary projection vectors (APVs) to map node embeddings into a 1D space, captures local node relationships through a fully differentiable kernel-based aggregation, and uses APV similarities for privacy-preserving personalized model mixing on the server.

D.3 IMPLEMENTATION DETAILS.

Compute Resources. Experiments were conducted on a Windows workstation equipped with an Intel Xeon Gold 6430 CPU, an NVIDIA RTX 4090 GPU with 24GB VRAM, and 120GB RAM, using Python 3.10, PyTorch 2.5.1 with CUDA 12.1, PyTorch Geometric 2.6.1, and Geoopt for hyperbolic graph neural network implementation.

For graph classification with FedSGR, we use a batch size of 32, dropout 0.3, Adam for Euclidean/-global parameters (lr=0.002, weight decay=5e-4), and RiemannianAdam for geometric parameters (lr=0.001). The mixed-curvature encoder initializes the hyperbolic curvature to -1.0 and the spherical curvature to 1.0; these curvatures are learnable, and we apply a curvature gradient clipping threshold of 0.1 with curvature regularization weight 0.005. The Ricci-gated convolution uses a contrastive temperature of 0.1 with contrastive loss weight 0.2, and we initialize the Ricci gate and Spectral gate fusion weights to 0.5 and 0.5, respectively. The federated procedure runs for 200 communication rounds across all clients.

E ADDITIONAL EXPERIMENTAL RESULTS

E.1 COMPUTATIONAL AND COMMUNICATION COSTS ANALYSIS

In this section, we provide a comprehensive analysis of the computational and communication overhead of the FedSGR framework. We assess both per-round metrics and cumulative training efficiency to demonstrate the practicality of our framework for real-world federated edge deployments.

Computational Complexity. The computational overhead of FedSGR is highly controllable and primarily concentrated in the pre-processing stage. The Ricci-gated GCN layer operates with a time complexity of $\mathcal{O}(|V_c|d^2 + |E_c|d)$, which is consistent with standard GNN architectures. The discrete Ricci curvature computation exhibits a complexity of $\mathcal{O}(|E_c| \cdot (deg_{max})^2)$. Crucially, as

elucidated in the underlying rationale of FedSGR, both the exact eigendecomposition for the spectral module ($\mathcal{O}(N_c^3)$) and the Ricci curvature computation are performed only once locally as offline pre-processing steps before the federated training begins. Consequently, these operations do not introduce any additional computational latency during the iterative training rounds on resource-constrained edge devices.

Efficiency Analysis. While manifold-based message passing and dual-gating mechanisms introduce a slightly higher time cost per local epoch compared to vanilla Euclidean frameworks (due to the non-linear exponential and logarithmic mappings), the highly informative geometric and spectral priors significantly accelerate the convergence process. As demonstrated in Table 6, FedSGR requires drastically fewer communication rounds to reach a target accuracy threshold compared to competitive baselines. This empirical observation aligns with our theoretical proof in Theorem 1, where the shared spectral knowledge effectively tightens the convergence bound by reducing the client drift variance σ^2.

Communication Overhead. Regarding the communication payload, FedSGR requires clients to upload only the parameters of the shared Spectral Filter Generator (ϕ_f^e). Although this module introduces a marginal increase in the data transferred per round compared to standard FedAvg, this overhead is effectively offset by the substantial reduction in the total number of communication rounds. As illustrated in the cumulative cost analysis (Table 7), the total training time and the total communication cost (the product of per-round payload and total rounds) of FedSGR remain highly competitive and, in most cases, superior to current state-of-the-art frameworks.

Table 6: Computational and Communication Costs Comparison on PubMed (80% Accuracy)

Method	Total Rounds	Time/Round (s)	Total Time (s)	Max Memory (MB)	Comm./Round (MB)	Total Comm. (MB)
FedAvg	7	2.62	18.34	33.35	3.43	24.01
FedProx	9	3.08	27.72	36.47	3.13	28.17
GCFL	7	2.65	18.55	36.63	3.39	23.73
FED-PUB	9	3.07	27.63	36.47	3.09	27.81
FedSSP	6	6.48	38.88	85.34	8.08	48.48
FedIIH	5	5.30	26.51	36.55	2.46	12.32
FedSPA	4	3.27	13.08	32.82	2.78	11.12
FedAUX	5	3.14	15.73	53.61	5.55	27.75
FedSGR	**3**	8.09	24.27	61.78	7.76	23.28

Table 7: Comparison of Total Rounds Required to Reach Target Accuracy on Multiple Datasets

Method	Cora (80%)	CiteSeer(70%)	PubMed(80%)	Mutagenicity(70%)	Amazon-Com.(80%)	Amazon-Pho.(90%)
FedAvg	10	8	7	36	25	15
FedProx	11	9	9	49	28	29
GCFL	10	8	7	36	21	13
FED-PUB	6	7	9	23	37	13
FedSSP	13	10	6	43	18	28
FedIIH	9	7	5	18	11	9
FedSPA	9	7	4	31	13	8
FedAUX	7	6	5	15	10	10
FedSGR	**4**	**4**	**3**	**5**	**3**	**6**

E.2 Hyper-parameter Analysis

We analyze the sensitivity of the dual-gate fusion by varying the initial weights of the Spectral Gate and Ricci Gate. Specifically, we set $w_s + w_g = 1$ and sweep $w_s \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. Results are reported in Table 8 and Figure 8.

Overall, the optimal (w_s, w_g) is dataset-dependent. On CiteSeer, larger spectral weight yields the best performance (75.97% at $w_s = 0.9$), indicating that globally shared spectral priors provide stronger transferable guidance for local learning. On Cora, a larger Ricci weight performs best (86.19% at $w_g = 0.7$), suggesting topology-aware curvature gating is more important for adaptive message passing. For PubMed and Mutagenicity, the best results are achieved with a balanced fusion ($w_s = w_g = 0.5$), showing that spectral guidance and curvature-based topology cues are complementary. Based on these observations, we use $w_s = w_g = 0.5$ as the default setting for the remaining experiments.

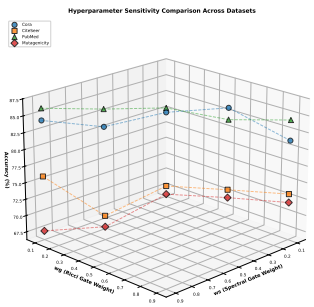


Figure 8: Hyperparameter Sensitivity Comparison Across Datasets

Table 8: Hyperparameter Sensitivity of Gate Weights

w_s	w_g	**Cora**	**CiteSeer**	**PubMed**	**Mutagenicity**
0.1	0.9	81.34%	73.29%	84.40%	72.00%
0.3	0.7	86.19%	73.95%	84.45%	72.75%
0.5	0.5	85.53%	74.52%	86.18%	73.28%
0.7	0.3	83.39%	69.96%	86.01%	68.28%
0.9	0.1	84.34%	75.97%	86.13%	67.64%

E.3 Adaptive Mechanism for Gating Selection: FedSGR-LS.

The fixed-weight analysis above, while informative, implicitly assumes a homogeneous preference across all clients. In practice, however, the optimal balance between global spectral guidance and local Ricci gating is inherently client-dependent, and a static heuristic may lead to suboptimal personalization in highly heterogeneous deployments. To thoroughly resolve this limitation, we introduce an adaptive gating parameter selection mechanism, denoted as **FedSGR-LS** (Learnable Scalars).

Instead of relying on fixed hyperparameters, we formulate w_s and w_g as client-specific learnable scalar weights. Initialized evenly, these two scalars are integrated into the local model parameters. They are updated via standard backpropagation using the local task loss during each client's local training phase, followed by a Softmax normalization to ensure $w_s + w_g = 1$. Importantly, this adaptive mechanism introduces exactly two scalar parameters per client. The additional computational and communication overhead is virtually negligible, making it remarkably lightweight and easy to implement in practical edge scenarios.

We extensively evaluate FedSGR-LS against the original framework across multiple datasets. As reported in Table 9, FedSGR-LS consistently matches or surpasses its fixed-weight counterpart. Furthermore, Table 10 explicitly reveals the individualized w_s and w_g parameters autonomously learned by each of the 10 clients on the Cora dataset. The learned weights deviate noticeably from the uniform initialization, with w_s ranging from 0.46 to 0.51 and w_g from 0.49 to 0.54. This confirms that unseen clients dynamically adjust their gating ratios based on their personalized local data rather than collapsing to a trivial consensus.

Table 9: Accuracy comparison of FedSGR and FedSGR-LS across multiple datasets (10 clients)

Method	Cora	CiteSeer	Mutagenicity	PubMed	Amazon-Computer	Amazon-Photo
FedSGR	84.06%±0.23	74.50%±1.30	**70.64%±1.50**	85.21%±0.26	87.37%±0.87	94.20%±0.53
FedSGR-LS	**84.63%±0.24**	**74.72%±1.13**	70.46%±1.31	**85.79%±0.30**	**88.30%±0.34**	**94.35%±0.27**

Table 10: Individualized w_s and w_g learned by each client on Cora (FedSGR-LS)

Param. \ Client ID	Client1	Client2	Client3	Client4	Client5	Client6	Client7	Client8	Client9	Client10
w_s	0.49	0.46	0.51	0.48	0.47	0.51	0.46	0.49	0.48	0.49
w_g	0.51	0.54	0.49	0.52	0.53	0.49	0.53	0.51	0.52	0.51

F LIMITATIONS AND FUTURE WORK

The main limitation of our work is that while the universal spectral knowledge sharing mechanism provides robust global structural guidance, it requires exact eigendecomposition of the local normalized Laplacian at each client. Thus, the pre-computation cost for the spectral module scales cubically, $\mathcal{O}(N_k^3)$, which may become a computational bottleneck for extremely large-scale client graphs despite being a one-time operation. Our future work focuses on addressing this scalability challenge by incorporating efficient spectral approximation techniques, such as Chebyshev polynomial expansions or randomized spectral filtering, to reduce the local computational load to near-linear complexity. We stress that our core contribution is a novel framework that unifies Riemannian geometric enhancement with universal spectral priors to handle severe structural heterogeneity, and optimizing the trade-off between geometric-spectral expressiveness and large-scale efficiency remains a complementary line of future research.