FedSPA : Generalizable Federated Graph Learning under Homophily Heterogeneity

Zihan Tan[1*] Guancheng Wan[1*] Wenke Huang[1*] He Li[1] Guibin Zhang[3]
Carl Yang[4] Mang Ye[1,2†]
[1] National Engineering Research Center for Multimedia Software, Institute of Artificial Intelligence, Hubei Key Laboratory of Multimedia and Network Communication Engineering, School of Computer Science, Wuhan University, Wuhan, China.
[2] Taikang Center for Life and Medical Sciences, Wuhan University, Wuhan, China
[3] National University of Singapore, Singapore
[4] Department of Computer Science, Emory University, Atlanta, GA, USA
{zihantan,guanchengwan,wenkehuang,heli,yemang}@whu.edu.cn,
guibinz@outlook.com, j.carlyang@emory.edu

Abstract

*Federated Graph Learning (FGL) has emerged as a solution to address real-world privacy concerns and data silos in graph learning, which relies on Graph Neural Networks (GNNs). Nevertheless, the homophily level discrepancies within the local graph data of clients, termed homophily heterogeneity, significantly degrade the generalizability of a global GNN. Existing research ignores this issue and suffers from unpromising collaboration. In this paper, we propose **FedSPA**, an effective hyperparameter-free framework that addresses homophily heterogeneity from the perspectives of homophily conflict and homophily bias, concepts that have yet to be defined or explored. In the first place, the homophily conflict arises when training on inconsistent homophily levels across clients. Correspondingly, we propose **S**ubgraph Feature **P**ropagation Decoupling (SFPD), thereby achieving collaboration on unified homophily levels across clients. To further address homophily bias, we design Homophily Bias-Driven **A**ggregation (HBDA) which emphasizes clients with lower biases. It enables the adaptive adjustment of each client contribution to the global GNN based on its homophily bias. The superiority of **FedSPA** is validated through extensive experiments.*

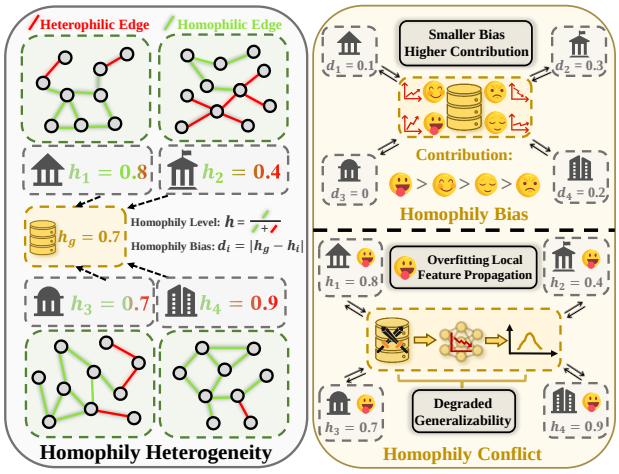
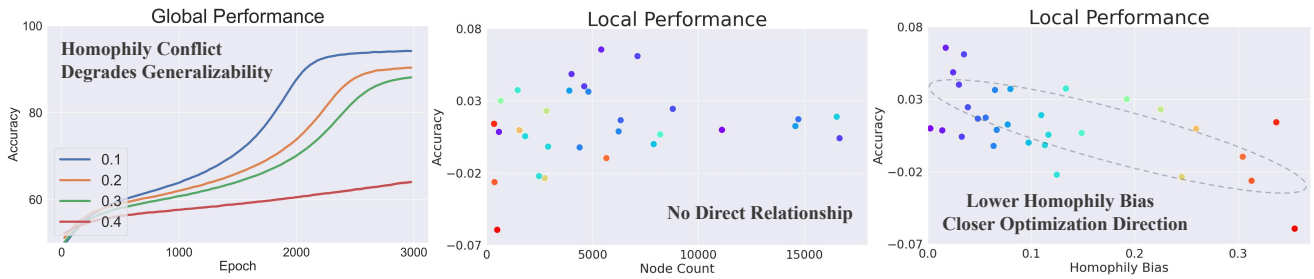


Figure 1. Problem illustration. The homophily level of a graph determines the proportion of edges connecting nodes with the same label. Discrepancies in homophily levels across clients, namely homophily heterogeneity, introduce two main challenges. (a) **Homophily conflict**: Under **inconsistent** homophily levels, client i tends to overfit its local level h_i, resulting in a conflict of feature propagation scheme and degrading global generalizability. (b) **Homophily bias**: client i with **lower** homophily level bias $d_i = |h_g - h_i|$ is more closely aligned with the global optimization direction, thus having a **greater** contribution to the globe.

1. Introduction

Federated learning (FL) [26, 28, 29, 52, 55, 60, 83] offers the ability to conduct distributed collaborative machine learning without leaking privacy [24, 30–33]. However, in real-world scenarios, a considerable amount of data generated by edge devices are graph-structured [6, 74, 75, 79], such as epidemiology [48] and scene graph [3, 9, 78, 91]. Therefore, Federated Graph Learning (FGL) emerges to address the challenges of leaking privacy and data silos

*Equal contribution. † Corresponding author.



(a) Impact of Growing Homophily Heterogeneity (b) Local Impact of Node Count (c) Local Impact of Homophily Bias

Figure 2. Phenomenon Demonstration. (a) Accuracy curves under a uniform distribution of client homophily levels with the same mean but varying gaps. It reveals that **inconsistent** homophily levels across clients lead to intensified homophily conflict, thereby significantly degrading generalizability. (b) There is no direct relationship between node count and local performance. (c) The **lower** the homophily bias, the higher the local performance. It indicates a closer optimization direction and **greater** contribution to the global GNN.

[17, 21, 47, 86] and has become a promising direction.

Nevertheless, most existing research overlooks the inconsistency in homophily levels across clients. The homophily level, which dictates the proportion of edges connecting nodes within the same class, has a substantial impact on the feature propagation scheme within the graph. We innovatively define this phenomenon as **homophily heterogeneity**. Furthermore, we are the first to investigate this issue from two novel perspectives. First, discrepancies in feature propagation schemes create diverse optimization directions across clients, defined as **homophily biases**. Additionally, GNNs tend to overfit to the local homophily level, thus undermining generalizable collaboration, termed as **homophily conflict**. A detailed illustration of these two challenges is provided in Fig. 2. Nevertheless, most non-iid FGL methods [27, 65, 70, 82] primarily improve from the perspective of structure. While a few studies [43] have acknowledged variations in homophily levels, they have neither fully examined the effects nor presented targeted solutions As a result, existing approaches fail to sufficiently address the homophily conflict and homophily bias, limiting the potential for effective collaboration.

To examine the impact of homophily conflict, we investigate using the widely used synthetic dataset CSBM [12]. As shown in Fig. 2 (a), the homophily levels of clients are uniformly distributed with the same means but varying gaps. It indicates that increased conflict significantly degrades generalizability, as GNNs tend to overfit the local homophily levels and feature propagation schemes. Drawing from the analysis above, we raise question **I)**: *How can we address homophily conflict under **inconsistent** homophily levels?*

In addition, global generalizability benefits from clients with optimization directions that align more clo sely with the global objective, resulting in unequal contributions from different clients. Conventional aggregation based solely on node count, as shown in Fig. 2 (b), is therefore unreasonable. We propose that clients with lower homophily biases to the globe are more valuable for the global GNN. For verifying, we empirically reveal in Fig. 2 (c) that clients with **lower** homophily biases from the globe exhibit greater performance growth. This suggests that such clients align more closely with the global optimization directions and have **greater** contributions to global generalizability. Based on this, we pose question **II)**: *How can we design an appropriate aggregation strategy that reasonably considers client contributions under homophily biases?*

To start with, we aim to address the homophily conflict outlined in question **I)**, which arises from training on inconsistent levels of homophily. To resolve the issue, we propose a targeted Subgraph Feature Propagation Decoupling (SFPD) that decouples feature propagation on homophilic and heterophilic edges using distinct channels. Specifically, Subgraph Feature Propagation Decoupling assigns separate channels to focus on local subgraphs of homophily levels of zero or one (fully heterophilic or homophilic). It facilitates collaboration on a unified homophily level for each type of channel, effectively mitigating the homophily conflict. Furthermore, considering invisible labels, a harmonizer integrates the process of the two channels for edges of uncertain relationships. While the concept of decoupling homophily and heterophily has been explored in some graph learning methods, these approaches primarily aim to improve GNN performance on heterophilic edges. In contrast, our strategy introduces a novel mechanism that enables effective collaboration at unified homophily levels across clients, significantly mitigating homophily conflicts in FGL.

To answer question **II)**, we aim to qualify client contribution under homophily bias. Based on our SFPD architecture, we innovatively summarize homophily bias into two aspects: First, channel harmonizers focus on inconsistent unknown homophily levels across clients, which we define as prior distribution bias for the levels. Moreover, these biases still influence the structural properties of subgraphs for the two channels, which can be potentially reflected in parameter updates. We refer to this as posterior parameter bias. Considering the two aspects, we design

Homophily Bias-Driven Aggregation (HBDA) to emphasize clients with lower bias and increase their weight during aggregation. For prior distribution bias, we first provide a theoretical proof of the negative correlation between graph high-frequency area and its homophily level in Appendix A. Correspondingly, the high-frequency area is exploited for quantifying the biases and the reweighing instead of the unavailable homophily level due to invisible labels. For posterior parameter bias, we investigate the Fisher Information Matrix (FIM) [1, 16, 51] which enables the quantifying of parameter sensitivity distribution disparity between a client and the globe. Combining these two strategies for distinct components in GNNs, HBDA facilitates a reasonable reweighing, enabling the global GNN to benefit more from clients with lower homophily biases.

In conclusion, our key contributions are:

- To the best of our knowledge, we are the first to define and explore homophily conflict and homophily biases.
- We discover two unrevealed phenomena of homophily heterogeneity: First, the conflict under **inconsistent** homophily levels degrades generalizability. Second, a **lower** homophily bias of a client indicates **greater** contribution.
- We innovatively design two hyperparameter-free strategies based on our motivation, including a decoupling strategy SFPD and an aggregation approach HBDA.
- Experiments on homophilic and heterophilic datasets validate the superiority of our framework FedSPA.

2. Related Work

2.1. Federated Learning

Federated learning(FL) provides a distributed machine-learning framework that does not require data sharing [34, 59, 80, 93, 94]. Nevertheless, clients may have diverse label and feature distributions due to their various behaviors and habits [14, 46, 58, 64], which poses the non-iid issue to traditional FL [37, 38, 68, 90]. To address the issue, FedProx designs convergence guarantees [41], FedNova mitigates objective inconsistency problems [72], MOON conducts model-level contrastive learning [40]. Meanwhile, many approaches aim to customize models that perform optimally for each client locally, namely Personalized Federated Learning (pFL) [2, 81, 95]. For example, FedALA [87] proposed personalized masks while FedCP [88] separates global and personal knowledge through model splitting. Furthermore, FedPHP preserves historical information [45], APPLE proposes core models for collaboration and employing directed relationship vectors for personalization [50], FedRoD [8] learns with two heads for both global and personalized tasks. Nevertheless, due to their lack of tailored design for graph data, FL methods often partially lose their superiority in FGL [15, 23, 57, 63, 76].

2.2. Federated Graph Learning

Federated Graph Learning (FGL) [7, 21] addresses data silos while safeguarding graph data privacy [20, 39, 42, 73]. Current FGL research can generally be categorized into two types: inter-graph FGL and intra-graph FGL. In inter-graph FGL, various clients aim to obtain a globally generalizable model or optimize local models by training GNNs on different local graphs and collaborating in FL scenarios [65, 82]. In intra-graph FGL, clients collaborate to accomplish tasks such as missing link prediction [10], subgraph community detection [4, 89], and node classification [27, 44, 70]. Nevertheless, existing FGL algorithms overlook the homophily level heterogeneity in real-world FGL scenarios, thereby inevitably losing the global generalizability due to homophily conflict and homophily bias. This paper primarily focuses on inter-graph FGL scenarios under homophily heterogeneity, innovatively revealing the unexplored impacts of homophily heterogeneity. We effectively address the issue from two aspects: promoting training on unified homophily levels across clients and enhancing the global emphasis on clients with lower homophily bias.

2.3. Homophilic and Heterophilic Graphs

Homophilic graphs are characterized by nodes with similar attributes tending to be connected, whereas heterophilic graphs have connections between nodes of dissimilar features and distinct class labels. Extensive research has achieved significant success in graph modeling [35, 71, 84, 85], such as GCN [36] and GraphSAGE [19]. Additionally, GAT [69], AGNN [67], and Geom-GCN [56] have further strengthened the effectiveness of GNNs in capturing graph features. However, many of these traditional studies have focused on strong graph homophily assumptions. Recently, numerous studies have considered heterophilic graphs, where GNNs have traditionally underperformed [92, 98]. For instance, H2GNN proposed three key designs for heterophily [96], and BM-GCN leverages pseudo labels in the graph convolution [22]. CPGNN incorporates an interpretable compatibility matrix for modeling the heterophily or homophily level [97], while ACM-GCN leverages an adaptive channel mixing [49]. Despite extensive exploration in handling graph data [13, 18, 61], homophily heterogeneity in FGL remains unaddressed. Correspondingly, we investigate from unexplored perspectives on homophily conflict and homophily bias. Furthermore, experiments on both homophilic and heterophilic graphs validate the superiority of our framework FedSPA.

3. Problem Statement

3.1. Preliminary

Notations. Consider the graph data represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the set of nodes comprising $|\mathcal{V}| =$

N vertices, and the edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ indicates the connections between them. The adjacency matrix is denoted as $\mathbf{A} \in \mathbb{R}^{N \times N}$, where $\mathbf{A}_{uv} = 1$ represents the existing edge $e_{uv} \in \mathcal{E}$ and $\mathbf{A}_{uv} = 0$ otherwise. The normalized form of the adjacency matrix is given by $\hat{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$, where the degree matrix $\mathbf{D}$ is defined as $D_{uu} = \sum_v \mathbf{A}_{uv}$. The Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}$, while the symmetric normalized Laplacian matrix is $\tilde{\mathbf{L}} = \mathbf{I} - \hat{\mathbf{A}}$, with $\mathbf{I}$ being the identity matrix. The symmetric normalized Laplacian matrix can be decomposed as $\tilde{\mathbf{L}} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$, in which $\mathbf{\Lambda} = \text{diag}(\lambda_0, \ldots, \lambda_{N-1})$ is a diagonal matrix of eigenvalues satisfying $0 = \lambda_0 \leq \lambda_1 \leq \ldots \leq \lambda_{N-1} \leq 2$, and $\mathbf{U}$ is a unitary matrix whose columns are the eigenvectors.

Graph Homophily Level h**:** *For a given graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ *with nodes labeled by the vector* $\mathbf{y}$*, the graph homophily is a measure of the similarity of connected nodes. It is defined as the ratio of edges that connect nodes with the same label to the total number. Formally, the homophily* h *of graph* $\mathcal{G}$ *is calculated as:*

$$h = \frac{\sum_{(u,v)\in\mathcal{E}} \mathbb{I}(\mathbf{y}_u = \mathbf{y}_v)}{|\mathcal{E}|}, \tag{1}$$

where $|\mathcal{E}|$ *is the total count of edges, and* $\mathbb{I}(\mathbf{y}_u = \mathbf{y}_v)$ *is a function that returns 1 if* $\mathbf{y}_u = \mathbf{y}_v$ *and 0 otherwise.*

High-frequency Area S_{high}**:** *The high-frequency area* S_{high} *is the integral of* $1 - f(t)$ *over the interval from* 0 *to* N*, mathematically expressed as* $S_{high} = \int_0^N (1 - f(t))dt$*, where* $t \in [\lambda_k, \lambda_{k+1})$ *and* $1 \leq k \leq N - 1$*. According to [66],* S_{high} *can be calculated through simple algebraic manipulations without eigen-decomposition:*

$$S_{\text{high}} = \frac{\sum_{k=1}^{N} \lambda_k \hat{x}_k^2}{\sum_{k=1}^{N} \hat{x}_k^2} = \frac{\mathbf{x}^T\mathbf{L}\mathbf{x}}{\mathbf{x}^T\mathbf{x}}. \tag{2}$$

Homophily Heterogeneity in FGL: *The homophily level determines the proportion of homophilic edges that connect nodes from the same class, significantly influencing the feature propagation schemes of GNNs. Nevertheless, the homophily levels are inconsistent across clients, leading to homophily heterogeneity.* ***Homophily conflict:*** *Clients overfit their local homophily level and feature propagation schemes, resulting in conflict during collaboration.* ***Homophily Bias:*** *Homophily level discrepancy between a client and the globe is defined as homophily bias. Based on HBDA GNN architecture, we attribute it to prior distribution bias and posterior parameter bias. A lower homophily bias indicates a closer alignment with the global optimization direction and higher contribution.*

Motivation of Decoupling. Homophily heterogeneity inevitably introduces biases in feature propagation schemes across clients. As a result, GNNs tend to overfit local homophily levels leading to homophily conflicts during collaboration. We empirically demonstrate that the conflict under **inconsistent** homophily levels degrades the global generalizability in Fig. 2(a). Specifically, the widely-used Contextual Stochastic Block Model (CSBM) is utilized to generate graphs with varying homophily levels for simulating, please refer to Appendix C for a detailed description. To address it, our Homophily Bias-Driven Aggregation (HBDA) strategy decouples the feature propagation on fully homophilic and heterophilic subgraphs with distinct channels. The two channels focus on homophily levels of 0 and 1, respectively, ensuring that clients collaborate within unified homophily levels. Furthermore, a harmonizer integrates outputs of the two channels for subgraphs where edge relationships are unknown due to invisible labels.

Motivation of High-frequency area. We posit that clients with lower homophily bias have optimization directions that are closer to the global objective. Therefore, they are expected to contribute more to a generalizable global GNN. As demonstrated in Fig. 2(c), clients with **lower** homophily bias experience greater performance growth. It indicates that the lower the homophily bias of a client, the closer its optimization direction aligns with the global objective, leading to **greater** contribution to global collaboration. Based on this observation, while it is intuitive to use the homophily level directly to quantify client contribution, the overall local homophily levels are inaccessible due to partially visible labels. Alternatively, we investigate graph spectral high-frequency areas to measure the bias from a novel spectral perspective. Specifically, we theoretically prove the negative correlation between the high-frequency area S_{high} and the homophily level h (refer to Appendix A for the full proof). Therefore, the homophily biases can be quantified through high-frequency area discrepancies between clients and the globe. To address this, we design Prior Bias-Driven Aggregation, the first component in HBDA, which identifies the clients with lower high-frequency area bias to the global, thus placing greater emphasis on them. This strategy is applied to the channel harmonizer to address the prior homophily level distribution bias issue.

Motivation of Fisher Information Matrix. Additionally, we aim to address posterior parameter bias in the homophilic and heterophilic channels which have avoided prior homophily biases. Specifically, the FIM is exploited [1, 16, 51] for measuring the GNN parameters sensitivity distributions bias between local GNNs and the globe, thereby identifying the clients with lower homophily bias. The FIM quantifies the information carried by observable random variables about the unknown parameters of the target distribution. Let $\mathcal{G}$ represent the observed graph data and θ represent the parameters of the distribution $p(\mathcal{G}|\theta)$. Due to the impracticality of directly computing the FIM in over-parameterized networks, we approximate it using the

empirical distribution as follows:

$$\mathbf{I}_{\text{diag}}(\theta) \approx \mathbb{E}_{\mathcal{G}\sim p(\mathcal{G}|\theta)}\left[\left(\frac{\partial \log p(\mathcal{G}|\theta)}{\partial \theta}\right)^2\right]. \quad (3)$$

It serves as a measure of the relative sensitivity of each parameter in the model and is applied to the two channels where prior distribution bias has been addressed through SFPD. Based on this, we propose Posterior Bias-Driven Aggregation, the second component of HBDA. For the comprehensive explanation of FIM, please refer to Appendix B.

4. Methodology

Overview. In Subgraph Feature Propagation Decoupling, we decouple the feature propagation on homophilic and heterophilic subgraphs through distinct channels. Correspondingly, the channels respectively focus on the a unified homophily level across clients, either 0 for the homophilic or 1 for the heterophilic. In addition, considering invisible labels, a harmonizer is designed to combine the two channels for subgraphs whose edges are of unknown relationship. In Homophily Bias-Driven Aggregation, we consider prior and posterior biases under homophily biases. For harmonizers that focus on unknown relationships, the homophily level distribution bias explicitly exists across clients. To address it, prior bias-driven aggregation based on spectral high-frequency area enables the global GNN to benefit more from clients with lower bias. For homophilic and heterophilic channels that have already avoided the prior homophily biases, posterior bias-driven aggregation based on the Fisher Information Matrix (FIM) raises the weights of those beneficial clients with smaller posterior biases from the perspective of parameter sensitivity. Please refer to Fig. 3 for a comprehensive framework illustration.

4.1. Subgraph Feature Propagation Decoupling

First and foremost, our SFPD model architecture decouples the handling of fully homophilic and heterophilic subgraphs by distinct channels, ensuring these two channels are trained on unified homophily levels across clients, mitigating homophily conflict during collaboration.

Assuming node v has a neighbor u, the features propagation process from u to v can be expressed as follows:

$$z^l_{u\to v} = \begin{cases} f_{\text{hom}}(z^{l-1}_u;\theta_{\text{hom}}) & \text{if } \mathbf{y}_u = \mathbf{y}_v \\ f_{\text{het}}(z^{l-1}_u;\theta_{\text{het}}) & \text{if } \mathbf{y}_u \neq \mathbf{y}_v \\ f_{\text{unk}}(z^{l-1}_u;\theta_{\text{unk}}) & \text{if } \mathbf{y}_u = \varnothing \text{ or } \mathbf{y}_v = \varnothing, \end{cases} \quad (4)$$

where the terms f_{hom}, f_{het}, and f_{unk} process the features from the homophilic, heterophilic, and unknown neighbor u respectively. $\mathbf{y}$ represent the node label, while $\varnothing$ denoting that the label is invisible. Specifically, an unknown relationship occurs when the label of either node in a connected pair is not visible. Ultimately, $z^l_{u\to v}$ represents the feature passed from neighbor u to the center node v. Notably, f_{hom} and f_{het} are the functions that own their respective parameters, while θ_{unk} is integrated from θ_{hom} and θ_{het} through a harmonizer, enabling the mixed feature propagation from unknown neighbors via the homophilic and heterophilic channels. It can be presented as follows:

$$\alpha = \tau(\phi(z^{l-1}_v);\theta_\phi), \quad \theta_{\text{unk}} = \alpha\cdot\theta_{\text{hom}} + (1-\alpha)\cdot\theta_{\text{het}}, \quad (5)$$

where α represents the output of a sigmoid function τ applied to the result of channel harmonizer ϕ on the feature z^{l-1}_v of node v from the previous layer. The parameter θ_{unk} is then derived by a weighted combination of θ_{hom} and θ_{het}, where α serves as the weight for θ_{hom} and $1-\alpha$ as the weight for θ_{het}. The updated feature for node v from unknown neighbors, $z^{l,\text{unk}}_{u\to v}$, is then computed by aggregating the features from its unknown neighbors $u \in \mathcal{N}^{\text{unk}}_v$ utilizing f_{unk} with the combined parameter θ_{unk} in Eq. (6).

To present the feature propagation process in our strategy SFPD, feature aggregation of a given node v in our graph convolution can be implemented as follows:

$$\begin{aligned} z^l_v = f_{\text{self}}(z^{l-1}_v;\theta_{\text{self}}) &+ \sum_{u\in\mathcal{N}^{\text{hom}}_v} f_{\text{hom}}(z^{l-1}_u;\theta_{\text{hom}}) \\ &+ \sum_{u\in\mathcal{N}^{\text{het}}_v} f_{\text{het}}(z^{l-1}_u;\theta_{\text{het}}) + \sum_{u\in\mathcal{N}^{\text{unk}}_v} f_{\text{unk}}(z^{l-1}_u;\theta_{\text{unk}}), \end{aligned} \quad (6)$$

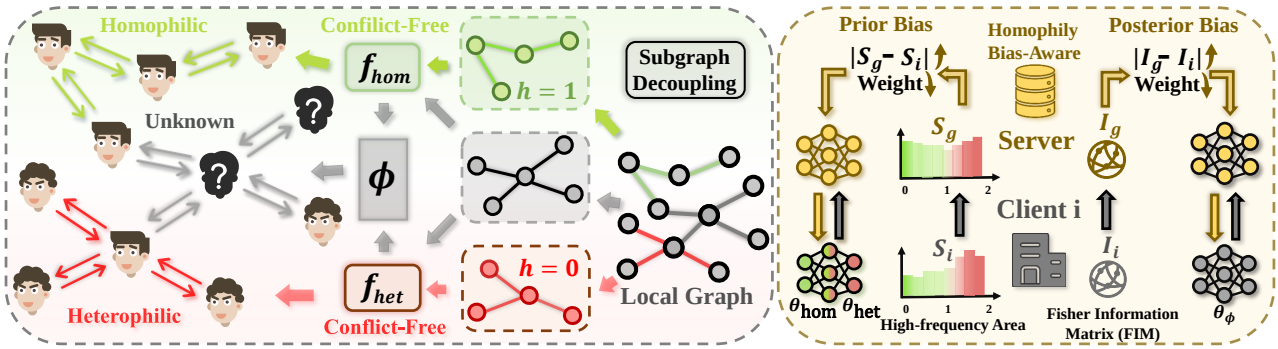
where z^l_v represents the feature of node v after feature propagation by graph convolution layer l. The function $f_{\text{self}}(z^{l-1}_v;\theta_{\text{self}})$ updates the feature of node v based on its own feature from the previous layer $l-1$. Following our design in Eq. (4), u denotes neighbors of node v and belongs to the corresponding neighbor set $\mathcal{N}^{\text{hom}}_v$, $\mathcal{N}^{\text{het}}_v$, or $\mathcal{N}^{\text{unk}}_v$. By extending the node-level operations to the entire local graphs, clients can achieve feature propagation decoupling for homophilic and heterophilic subgraphs. It promotes the training of GNN with unified homophily levels across clients, thereby mitigating homophily conflict.

4.2. Homophily Bias-Driven Aggregation

Prior Bias-Driven Aggregation For channel harmonizer parameters θ_ϕ training on subgraphs of unknown relationships where exists prior homophily level distribution bias across clients, we investigate graph spectra in high-frequency areas to reasonably measure client importance for a suitable aggregation strategy. First and foremost, high-frequency area $\mathbf{S}$ for each client is firstly formulated as:

$$\mathbf{S} = \frac{1}{F}\sum_{f=1}^{F} s_f = \frac{1}{F}\sum_{f=1}^{F} \frac{\mathbf{X}^T_{:,f}\mathbf{L}\mathbf{X}_{:,f}}{\mathbf{X}^T_{:,f}\mathbf{X}_{:,f}}, \quad (7)$$

where F denotes the feature dimension and s_f represents high-frequency area for the f-th feature dimension. Here,



(a) Subgraph Feature Propagation Decoupling **(b) Homophily Bias-Driven Aggregation**

Figure 3. Framework illustration. The left box (a) refers to Subgraph Feature Propagation Decoupling (SFPD), which mitigates homophily conflict by ensuring that each channel focuses on a **unified** homophily level across clients, either 1 for the homophilic channel f_{hom} or 0 for the heterophilic channel f_{het}. Harmonizer ϕ integrates f_{hom} and f_{het} to process subgraphs where edge relationships are unknown due to invisible labels. The right box (b) denotes Homophily Bias-Driven Aggregation (HBDA). It is applied to harmonizers ϕ and the two channels f_{hom} and f_{het} distinctively. HBDA sets **higher** weights for clients with **lower** homophily biases from the globe.

$\mathbf{X} \in \mathbb{R}^{N\times F}$ is the node feature matrix, where $\mathbf{X}_{:,f}$ denotes the f-th column of $\mathbf{X}$, corresponding to the feature vector for all nodes at the f-th dimension. $\mathbf{L} \in \mathbb{R}^{N\times N}$ represents the graph Laplacian matrix. $\mathbf{S}$ is obtained by averaging over all feature dimensions and normalizing by F. Specifically, the homophily bias of client i from the prior distribution bias perspective is calculated as follows:

$$\delta_i = \left| \mathbf{S}_i - \frac{\sum_{i=1}^{n} N_i \mathbf{S}_i}{\sum_{i=1}^{n} N_i} \right|, \tag{8}$$

where δ_i represents the bias and $\mathbf{S}_i$ denotes the high-frequency area of client i. n denotes the total number of clients and N_i denotes local node count for client i. The bias δ_i measures how much the high-frequency area of the client i deviates from the globe. Specifically, the larger the bias δ_i, the less the client contribution. Furthermore, to measure the weight for aggregation, $\gamma(\delta_i, \boldsymbol{\delta})$ is calculated as follows:

$$\gamma(\delta_i, \boldsymbol{\delta}) = \frac{\delta_i \cdot \min(\boldsymbol{\delta})}{\max(\boldsymbol{\delta}) \cdot \min(\boldsymbol{\delta})}, \tag{9}$$

where $\boldsymbol{\delta}$ represents the set of all high-frequency area disparities. Based on $\gamma(\delta_i, \boldsymbol{\delta})$ which ensures all disparities are on the same scale, the ultimate weight ${w_i}^h$ for the harmonizer parameters θ_ϕ of the client i is calculated as follows:

$${w_i}^h = \frac{\exp(-\gamma(\delta_i, \boldsymbol{\delta}))}{\sum_{i=1}^{n} \exp(-\gamma(\delta_i, \boldsymbol{\delta}))}, \tag{10}$$

where $\gamma(\delta_i, \boldsymbol{\delta})$ represents the normalized disparity. The utilization of the negative exponential function is intended to convert disparities into weights, where a larger disparity results in a smaller value. It ensures that clients with smaller biases have greater importance. ${w_i}^h$ represents the ultimate weights qualified by high-frequency areas that adequately evaluate prior homophily bias from a spectral perspective.

Posterior Bias-Driven Aggregation For homophilic and heterophilic channels that have already overcome prior homophily level distribution bias, we consider posterior parameter bias as the implicit impact of homophily biases. Within a unified architecture, graphs with various homophily levels lead to distinctive parameter sensitivity. For our previously designed dual-channel GNN architecture that decouples subgraphs, homophily biases affect different subgraph channels differently. Correspondingly, parameter sensitivity distribution for each channel is calculated individually for aggregation. To begin with, the distribution bias for client i is formulated as follows:

$$\Delta_i = \frac{1}{m} \sum_{j=1}^{m} \left| \mathbf{I}_i - \frac{1}{\sum_{i=1}^{n} |N_i|} \sum_{i=1}^{N} |N_i| \mathbf{I}_i \right|, \tag{11}$$

where Δ_i measures the difference between the parameter sensitivity distribution $\mathbf{I}_i$ of the client i by the FIM in Eq. (3) and the aggregated global sensitivity distribution. m represents the dimension number of $\mathbf{I}_i$ and $\mathbf{I}_g$. Therefore, mean deviation Δ_i provides a scalar value that quantifies how much the sensitivity distribution of a client deviates from the global. Subsequently, we normalize Δ_i using the max-min normalization Eq. (9). Then the same method is employed to determine weights w based on the parameter sensitivity distribution bias as follows:

$${w_i}^f = \frac{\exp(-\gamma(\Delta_i, \boldsymbol{\Delta}))}{\sum_{i=1}^{n} \exp(-\gamma(\Delta_i, \boldsymbol{\Delta}))}, \tag{12}$$

where the weight ${w_i}^f$ is obtained through same procedure as ${w_i}^h$. Correspondingly, $\boldsymbol{\Delta}$ denotes the set of all distribution biases, while Δ_i representing bias of client i. As discussed in the motivation, considering the inconsistent impact of homophily bias on feature propagation in different subgraphs, we compute ${w_i}^f$ independently for the homophilic and heterophilic channels by FIM.

5. Experiments

5.1. Experimental Settings

Baselines. We compare FedSPA with several state-of-the-art approaches. (1) **FedAvg** [54]; Two traditional FL methods, including (2) **FedProx** [41] which address heterogeneity issues in FL and (3) **FedNova** [72] which focus on objective inconsistency; One personalized FL algorithm (4) **FedFa** [95] which address the negative impact of feature shift; Seven FGL algorithms, including (5) **FedSage+** [89], (6) **FedStar** [65], (7) **FedPub** [4](8) **FGSSL** [27], (9) **FedGTA** [44], (10) **AdaFGL** [43], and (11) **FGGP** [70] which provide targeted solution for FGL.

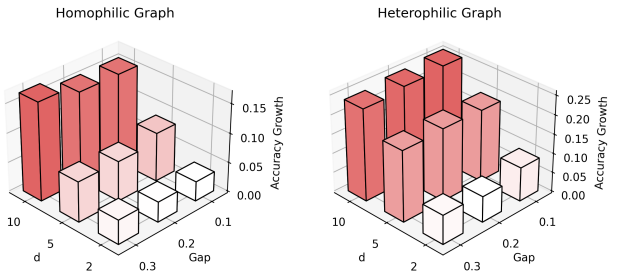
Datasets. We conducted experiments on both homophilic and heterophilic graph datasets to validate the superiority of our framework FedSPA. The homophilic datasets include Cora [53], Coauthor-CS, Coauthor-Physics [62], while the heterophilic datasets comprise Cornell, Wisconsin [11], Minesweeper [5], and Arxiv-Year [25]. Detailed descriptions of these datasets can be found in Appendix D.

Evaluation Metric. Following mainstream research practices [27, 43, 44, 70], we utilize the accuracy of the node classification task as the evaluation metric.

Implement Details. The experiments are conducted on the framework FLGo [77]. Following the mainstream research, the community detection Louvain algorithm is leveraged to simulate the subgraph systems. Despite the large graph dataset arxiv-year, we split the nodes of other datasets into the train, valid, and test sets, where the ratio is 0.6, 0.2, and 0.2, while the radio is 0.05, 0.475, and 0.475 for arxiv-year. Considering the graph size, different client scales are set for each dataset. Specifically, Cornell and Wisconsin have 5 clients each, Arxiv-Year has 30 clients, and the other datasets are set to 10 clients to reasonably simulate real-world subgraph-FL environments. Furthermore, we conduct experiments five times and reported the average accuracy of the last five communication epochs as the test performance. We conduct experiments with the ACM-GCN [49], which achieves a strong ability on both homophilic and heterophilic graph datasets. We utilize the ACM-GCN with 2 layers, with the hidden layer size of 128.

5.2. Performance Comparison

We demonstrate the federated node classification performance with different graph datasets and summarize the final average test accuracy in Tab. 1. These results indicate that FedSPA outperforms all other baselines in six of the seven settings. Specifically, algorithms tailored for FGL such as FedStar, FedPub, and FedGTA outperform traditional FL algorithms due to their designs for graph-structure data.



(a) Homophilic **(b)** Heterophilic

Figure 4. Analysis of the performance growth between FedSPA and Fedavg under different homophily gaps and average node degrees (average neighbor count of a node), where d denotes the average node degree and Gap refers to the range where client local homophily levels uniformly distribute.

5.3. Varying Homophily Gaps and Node Degrees

We explore the performance improvement of our method compared to FedAvg in both homophilic and heterophilic FGL under different homophily gaps. Specifically, Fig. 4 presents the accuracy growth of our method under three different average node degrees and homophily gaps. We derive two conclusions: firstly, the greater the average node degree, the more significant the impact of homophily level on the feature propagation of GNNs, thus enabling FedSPA to experience greater improvements. Furthermore, our method maintains its advantage across different homophily gaps. For implementation details, we conduct our experiments with the widely used synthetic graph dataset CSBM. Specifically, scenarios of both homophilic and heterophilic graphs are demonstrated. In the homophilic scenario, the average homophily of clients is set to 0.75, while in the heterophilic scenario, it is set to 0.25. The gap represents the distribution range of client local homophily levels, with they uniformly distributed within it. Furthermore, $d = 2, 5, 10$ denotes the average node degree, and node count in the graph randomly varies between 1,000 and 5,000.

5.4. Ablation Study

Effects of Key Components Mechanism of FedSPA: To better understand the impact of specific components on the overall performance of FedSPA, we conduct an ablation study where we vary these components of Coauthor-CS, Minesweeper, and Arxiv-year. As shown in Tab. 2, compared to FedAvg, SFPD significantly enhances accuracy when applied independently. Furthermore, HBDA attains a notable improvement in performance as it reasonably quantifies client importance based on SFPD architecture and addresses homophily biases adequately.

Effects of Key Components Mechanism of HBDA: we also conduct an ablation study where we vary components

Methods	Cora	CS	Physics	Cornell	Winsconsin	Minesweeper	Arxiv-year
Homophily Level	0.81	0.81	0.93	0.30	0.20	0.68	0.22
FedAvg [ASTAT17]	80.8 ± 0.6	91.8 ± 0.3	94.3 ± 0.6	72.3 ± 4.3	85.7 ± 3.5	81.1 ± 0.2	35.6 ± 0.6
FedProx [arXiv18]	80.9 ± 0.4 $\uparrow_{0.1}$	91.9 ± 0.5 $\uparrow_{0.1}$	94.4 ± 0.5 $\uparrow_{0.1}$	71.9 ± 3.7 $\downarrow_{0.4}$	86.2 ± 3.7 $\uparrow_{0.5}$	81.5 ± 0.1 $\uparrow_{0.4}$	35.5 ± 0.7 $\downarrow_{0.1}$
FedNova [NeurIPS20]	80.6 ± 0.7 $\downarrow_{0.2}$	91.7 ± 0.4 $\downarrow_{0.1}$	94.5 ± 0.3 $\uparrow_{0.2}$	71.9 ± 3.4 $\downarrow_{0.4}$	86.3 ± 5.1 $\uparrow_{0.6}$	81.4 ± 0.2 $\uparrow_{0.3}$	35.8 ± 0.8 $\uparrow_{0.2}$
FedFa [ICLR23]	82.1 ± 0.3 $\uparrow_{1.3}$	92.2 ± 0.1 $\uparrow_{0.4}$	93.9 ± 0.3 $\downarrow_{0.4}$	72.5 ± 5.2 $\uparrow_{0.2}$	86.1 ± 3.9 $\uparrow_{0.4}$	81.3 ± 0.1 $\uparrow_{0.2}$	36.3 ± 0.4 $\uparrow_{0.7}$
FedSage+ [NeurIPS19]	81.5 ± 0.7 $\uparrow_{0.7}$	92.6 ± 0.2 $\uparrow_{0.8}$	94.5 ± 0.9 $\uparrow_{0.2}$	73.1 ± 3.3 $\uparrow_{0.8}$	$\underline{88.2 \pm 4.8}$ $\uparrow_{2.5}$	81.8 ± 0.2 $\uparrow_{0.7}$	35.8 ± 1.0 $\uparrow_{0.2}$
FedStar [AAAI23]	81.4 ± 0.5 $\uparrow_{0.6}$	92.0 ± 0.4 $\uparrow_{0.2}$	94.0 ± 0.6 $\downarrow_{0.3}$	74.3 ± 2.7 $\uparrow_{2.0}$	87.9 ± 3.7 $\uparrow_{2.2}$	81.2 ± 0.1 $\uparrow_{0.1}$	36.9 ± 0.6 $\uparrow_{1.3}$
FedPub [ICML23]	81.0 ± 0.8 $\uparrow_{0.2}$	92.3 ± 0.2 $\uparrow_{0.5}$	94.4 ± 0.3 $\uparrow_{0.1}$	73.4 ± 3.5 $\uparrow_{1.1}$	86.5 ± 3.2 $\uparrow_{0.8}$	81.3 ± 0.2 $\uparrow_{0.2}$	36.6 ± 0.4 $\uparrow_{1.0}$
FGSSL [IJCAI23]	$\mathbf{82.8 \pm 0.7}$ $\uparrow_{2.0}$	$\underline{93.1 \pm 0.3}$ $\uparrow_{1.3}$	94.7 ± 0.6 $\uparrow_{0.4}$	73.6 ± 4.6 $\uparrow_{1.3}$	86.3 ± 3.5 $\uparrow_{0.6}$	81.7 ± 0.7 $\uparrow_{0.6}$	36.8 ± 0.3 $\uparrow_{1.2}$
FedGTA [VLDB24]	82.0 ± 0.4 $\uparrow_{1.2}$	91.9 ± 0.4 $\uparrow_{0.1}$	94.6 ± 0.4 $\uparrow_{0.3}$	72.6 ± 4.2 $\uparrow_{0.3}$	86.0 ± 4.2 $\uparrow_{0.3}$	81.4 ± 0.3 $\uparrow_{0.3}$	$\underline{37.1 \pm 0.6}$ $\uparrow_{1.5}$
AdaFGL [ICDE24]	81.2 ± 0.7 $\uparrow_{0.4}$	92.3 ± 0.5 $\uparrow_{0.5}$	$\underline{94.8 \pm 0.7}$ $\uparrow_{0.5}$	$\underline{74.9 \pm 3.2}$ $\uparrow_{2.6}$	88.0 ± 3.9 $\uparrow_{2.3}$	81.3 ± 0.1 $\uparrow_{0.2}$	36.9 ± 0.5 $\uparrow_{1.3}$
FGGP [AAAI24]	82.0 ± 0.4 $\uparrow_{1.2}$	92.8 ± 0.4 $\uparrow_{1.0}$	94.7 ± 0.5 $\uparrow_{0.4}$	73.8 ± 2.8 $\uparrow_{1.5}$	87.4 ± 4.7 $\uparrow_{1.7}$	$\underline{81.9 \pm 0.4}$ $\uparrow_{0.8}$	36.7 ± 0.6 $\uparrow_{1.1}$
FedSPA (ours)	$\underline{82.5 \pm 0.8}$ $\uparrow_{1.7}$	$\mathbf{94.4 \pm 0.1}$ $\uparrow_{2.6}$	$\mathbf{95.3 \pm 0.3}$ $\uparrow_{1.0}$	$\mathbf{77.0 \pm 2.9}$ $\uparrow_{4.7}$	$\mathbf{90.3 \pm 2.5}$ $\uparrow_{4.6}$	$\mathbf{83.2 \pm 0.2}$ $\uparrow_{2.1}$	$\mathbf{38.1 \pm 0.3}$ $\uparrow_{2.5}$

Table 1. Comparison with the state-of-the-art methods on both homophilic and heterophilic graph datasets. We report node classification accuracies for downstream task performance. Best in bold and second with underline. The growth or reduction is compared to FedAvg.

SFPD	HBDA	Dataset		
		CS	Minesweeper	Arxiv-year
✗	✗	91.78 ± 0.31	81.14 ± 0.21	35.60 ± 0.56
✓	✗	92.29 ± 0.45	81.62 ± 0.12	37.26 ± 0.88
✓	✓	$\mathbf{94.37 \pm 0.13}$	$\mathbf{83.21 \pm 0.21}$	$\mathbf{38.12 \pm 0.30}$

Table 2. Ablation study of key components (SFPD, HBDA) of FedSPA on Coauthor-CS, Minesweeper, and Arxiv-year.

of HBDA. As shown in Tab. 3, compared to only leveraging SFPD, prior and posterior bias-driven aggregation enhances the performance respectively. Consequently, their combined leveraging results in the optimal effectiveness, validating that HBDA effectively assesses the homophily bias and adjusts the client weights reasonably.

Prior	Posterior	Dataset		
		CS	Minesweeper	Arxiv-year
✗	✗	92.29 ± 0.45	81.62 ± 0.12	37.26 ± 0.88
✓	✗	93.31 ± 0.39	82.39 ± 0.42	37.77 ± 0.52
✗	✓	93.56 ± 0.74	82.67 ± 0.23	37.42 ± 0.68
✓	✓	$\mathbf{94.37 \pm 0.13}$	$\mathbf{83.21 \pm 0.21}$	$\mathbf{38.12 \pm 0.30}$

Table 3. Ablation study of key components (Prior Bias-Driven Aggregation and Posterior Bias-Driven Aggregation) of HBDA on Coauthor-CS, Minesweeper, and Arxiv-year.

5.5. Convergence

Fig. 5 shows curves of the average test accuracy during training across five random runs with the minesweeper and arxiv-year datasets. The comparison demonstrates the superiority of our framework FedSPA through its outstanding performance and stable convergence.

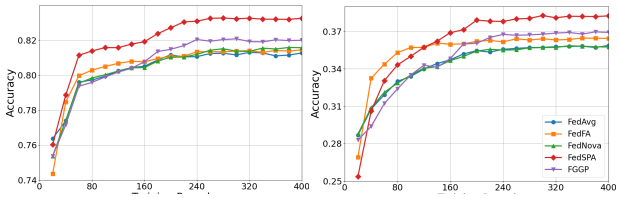
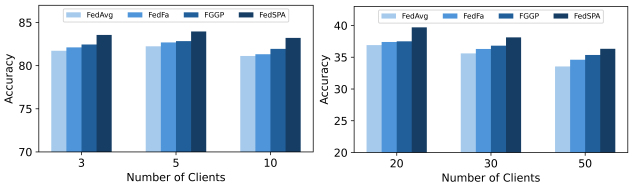


Figure 5. Visualization of training curves of the average test accuracy of FedSPA and other FL, pFL, and FGL algorithms on datasets minesweeper (left) and arxiv-year (right).

5.6. Varying Client Scales

The Fig. 6 demonstrates the performance of the FedAvg, FGGP, and FedSPA algorithms across different client scales on the graph datasets Minesweeper and Arxiv-Year. Our results show that FedSPA consistently outperforms other methods even as the client scale varies. For different graph sizes, the number of clients is adjusted accordingly.



(a) Minesweeper **(b)** Arxiv-year

Figure 6. Analysis of performance under different client numbers.

6. Conclusion

In this paper, we innovatively identify the issue of homophily heterogeneity in FGL and address it from homophily conflict and bias perspectives. In the first place, we

propose SFPD which decouples feature propagation on homophilic and heterophilic subgraphs through distinct channels, thus attaining training on unified homophily levels across clients for each type of channel, either one or zero. In addition, considering invisible labels, subgraphs whose edges are of unknown relationship are processed by both channels through a harmonizer. Furthermore, existing aggregation based on node count fails to reasonably quantify client contribution under homophily bias. Instead, we innovatively summarize the bias into prior distribution bias and posterior parameter bias. Correspondingly, an aggregation strategy HBDA that adequately addresses homophily bias is proposed to emphasize clients with lower homophily bias, thereby allowing the global GNN to benefit more from them. Combining these strategies, we propose FedSPA and conduct extensive experiments on homophilic and heterophilic graph datasets to validate its superiority.

7. Acknowledgement

This research is supported by the National Key Research and Development Project of China (2024YFC3308400), the National Natural Science Foundation of China (Grants 62361166629, 62176188, 623B2080), the Wuhan University Undergraduate Innovation Research Fund Project, the National Research Foundation. The supercomputing system at the Supercomputing Center of Wuhan University supported the numerical calculations in this paper. Carl Yang was not supported by any funds from China.

References

[1] Shun-ichi Amari and Hiroshi Nagaoka. *Methods of information geometry*. American Mathematical Soc., 2000. 3, 4, 13

[2] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019. 3

[3] Iro Armeni, Zhi-Yang He, JunYoung Gwak, Amir R Zamir, Martin Fischer, Jitendra Malik, and Silvio Savarese. 3d scene graph: A structure for unified semantics, 3d space, and camera. In *CVPR*, pages 5664–5673, 2019. 1

[4] Jinheon Baek, Wonyong Jeong, Jiongdao Jin, Jaehong Yoon, and Sung Ju Hwang. Personalized subgraph federated learning. In *ICML*, pages 1396–1415, 2023. 3, 7

[5] Dmitry Baranovskiy, Ivan Oseledets, and Andrey Babenko. A critical look at the evaluation of gnns under heterophily: Are we really making progress? *arXiv preprint arXiv:2302.11640*, 2023. 7, 14

[6] Chaoqi Chen, Jiongcheng Li, Zebiao Zheng, Yue Huang, Xinghao Ding, and Yizhou Yu. Dual bipartite graph learning: A general approach for domain adaptive object detection. In *CVPR*, pages 2703–2712, 2021. 1

[7] Fahao Chen, Peng Li, Toshiaki Miyazaki, and Celimuge Wu. Fedgraph: Federated graph learning with intelligent sampling. *IEEE PDS*, 33(8):1775–1786, 2021. 3

[8] Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. *arXiv preprint arXiv:2107.00778*, 2021. 3

[9] Jintai Chen, Biwen Lei, Qingyu Song, Haochao Ying, Danny Z Chen, and Jian Wu. A hierarchical graph network for 3d object detection on point clouds. In *CVPR*, pages 392–401, 2020. 1

[10] Mingyang Chen, Wen Zhang, Zonggang Yuan, Yantao Jia, and Huajun Chen. Fede: Embedding knowledge graphs in federated setting. In *IJCKG*, pages 80–88, 2021. 3

[11] Mark Craven, Dan DiPasquo, Dayne Freitag, Andrew McCallum, Tom Mitchell, Kamal Nigam, and Seán Slattery. Learning to extract symbolic knowledge from the world wide web. *AAAI/IAAI*, 1998. 7, 14

[12] Yash Deshpande, Subhabrata Sen, Andrea Montanari, and Elchanan Mossel. Contextual stochastic block models. *NeurIPS*, 31, 2018. 2, 13

[13] Mohammed Haroon Dupty, Yanfei Dong, Sicong Leng, Guoji Fu, Yong Liang Goh, Wei Lu, and Wee Sun Lee. Constrained layout generation with factor graphs. In *CVPR*, pages 12851–12860, 2024. 3

[14] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. In *NeurIPS*, pages 3557–3568, 2020. 3

[15] Chun-Mei Feng, Kai Yu, Nian Liu, Xinxing Xu, Salman Khan, and Wangmeng Zuo. Towards instance-adaptive inference for federated learning. In *CVPR*, pages 23287–23296, 2023. 3

[16] Ronald A Fisher. On the mathematical foundations of theoretical statistics. *Philosophical transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character*, 1922. 3, 4, 13

[17] Xingbo Fu, Binchi Zhang, Yushun Dong, Chen Chen, and Jundong Li. Federated graph machine learning: A survey of concepts, techniques, and applications. *arXiv preprint arXiv:2207.11812*, 2022. 2

[18] Shiwei Gan, Yafeng Yin, Zhiwei Jiang, Hongkai Wen, Lei Xie, and Sanglu Lu. Signgraph: A sign sequence is worth graphs of nodes. In *CVPR*, pages 13470–13479, 2024. 3

[19] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *NeurIPS*, 30, 2017. 3

[20] Zeeshan Hayder and Xuming He. Dsgg: Dense relation transformer for an end-to-end scene graph generation. In *CVPR*, pages 28317–28326, 2024. 3

[21] Chaoyang He, Keshav Balasubramanian, Emir Ceyani, Carl Yang, Han Xie, Lichao Sun, Lifang He, Liangwei Yang, Philip S Yu, Yu Rong, et al. Fedgraphnn: A federated learning system and benchmark for graph neural networks. In *ICLR*, 2021. 2, 3

[22] Dongxiao He, Chundong Liang, Huixin Liu, Mingxiang Wen, Pengfei Jiao, and Zhiyong Feng. Block modeling-guided graph convolutional neural networks. In *AAAI*, pages 4022–4029, 2022. 3

[23] Erdong Hu, Yuxin Tang, Anastasios Kyrillidis, and Chris Jermaine. Federated learning over images: vertical decompositions and pre-trained backbones are difficult to beat. In *CVPR*, pages 19385–19396, 2023. 3

[24] Ming Hu, Zhihao Yue, Xiaofei Xie, Cheng Chen, Yihao Huang, Xian Wei, Xiang Lian, Yang Liu, and Mingsong Chen. Is aggregation the only choice? federated learning via layer-wise model recombination. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1096–1107, 2024. 1

[25] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs, 2021. 7, 14

[26] Wenke Huang, Mang Ye, and Bo Du. Learn from others and be yourself in heterogeneous federated learning. In *CVPR*, 2022. 1

[27] Wenke Huang, Guancheng Wan, Mang Ye, and Bo Du. Federated graph semantic and structural learning. In *IJCAI*, pages 3830–3838, 2023. 2, 3, 7

[28] Wenke Huang, Mang Ye, Zekun Shi, and Bo Du. Generalizable heterogeneous federated cross-correlation and instance similarity learning. *TPAMI*, 2023. 1

[29] Wenke Huang, Mang Ye, Zekun Shi, He Li, and Bo Du. Rethinking federated learning with domain shift: A prototype view. In *CVPR*, 2023. 1

[30] 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. *arXiv preprint arXiv:2311.06750*, 2023. 1

[31] Wenke Huang, Zekun Shi, Mang Ye, He Li, and Bo Du. Self-driven entropy aggregation for byzantine-robust heterogeneous federated learning. In *Forty-first International Conference on Machine Learning*, 2024.

[32] Wenke Huang, Mang Ye, Zekun Shi, Bo Du, and Dacheng Tao. Fisher calibration for backdoor-robust heterogeneous federated learning. In *European Conference on Computer Vision*, 2024.

[33] Wenke Huang, Mang Ye, Zekun Shi, Guancheng Wan, He Li, and Bo Du. Parameter disparities dissection for backdoor defense in heterogeneous federated learning. In *NeurIPS*, 2025. 1

[34] Fatih Ilhan, Gong Su, and Ling Liu. Scalefl: Resource-adaptive federated learning with heterogeneous clients. In *CVPR*, pages 24532–24541, 2023. 3

[35] Bo Jiang, Ziyan Zhang, Doudou Lin, Jin Tang, and Bin Luo. Semi-supervised learning with graph learning-convolutional networks. In *CVPR*, pages 11313–11320, 2019. 3

[36] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017. 3

[37] Gihun Lee, Minchan Jeong, Sangmook Kim, Jaehoon Oh, and Se-Young Yun. Fedsol: Stabilized orthogonal learning with proximal restrictions in federated learning. In *CVPR*, pages 12512–12522, 2024. 3

[38] Hongxia Li, Wei Huang, Jingya Wang, and Ye Shi. Global and local prompts cooperation via optimal transport for federated learning. In *CVPR*, pages 12151–12161, 2024. 3

[39] Jiankai Li, Yunhong Wang, Xiefan Guo, Ruijie Yang, and Weixin Li. Leveraging predicate and triplet learning for scene graph generation. In *CVPR*, pages 28369–28379, 2024. 3

[40] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *CVPR*, pages 10713–10722, 2021. 3

[41] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 2020. 3, 7

[42] Xianxian Li, Qiyu Li, Haodong Qian, Jinyan Wang, et al. Contrastive learning of graphs under label noise. *Neural Networks*, page 106113, 2024. 3

[43] Xunkai Li, Zhengyu Wu, Wentao Zhang, Henan Sun, Rong-Hua Li, and Guoren Wang. Adafgl: A new paradigm for federated node classification with topology heterogeneity. *arXiv preprint arXiv:2401.11750*, 2024. 2, 7

[44] Xunkai Li, Zhengyu Wu, Wentao Zhang, Yinlin Zhu, Rong-Hua Li, and Guoren Wang. Fedgta: Topology-aware averaging for federated graph learning. In *VLDB*, 2024. 3, 7

[45] Xin-Chun Li, De-Chuan Zhan, Yunfeng Shao, Bingshuai Li, and Shaoming Song. Fedphp: Federated personalization with inherited private models. In *ECML*, 2021. 3

[46] Paul Pu Liang, Terrance Liu, Liu Ziyin, Nicholas B Allen, Randy P Auerbach, David Brent, Ruslan Salakhutdinov, and Louis-Philippe Morency. Think locally, act globally: Federated learning with local and global representations. *arXiv preprint arXiv:2001.01523*, 2020. 3

[47] Rui Liu and Han Yu. Federated graph neural networks: Overview, techniques and challenges. *arXiv preprint arXiv:2202.07256*, 2022. 2

[48] Zewen Liu, Guancheng Wan, B Aditya Prakash, Max SY Lau, and Wei Jin. A review of graph neural networks in epidemic modeling. *arXiv preprint arXiv:2403.19852*, 2024. 1

[49] Sitao Luan, Chenqing Hua, Qincheng Lu, Jiaqi Zhu, Mingde Zhao, Shuyuan Zhang, Xiao-Wen Chang, and Doina Precup. Revisiting heterophily for graph neural networks. In *NeurIPS*, 2022. 3, 7

[50] Jun Luo and Shandong Wu. Adapt to adaptation: Learning personalization for cross-silo federated learning. In *IJCAI*, pages 2166–2173, 2022. 3

[51] Alexander Ly, Maarten Marsman, Josine Verhagen, Raoul P.P.P. Grasman, and Eric-Jan Wagenmakers. A tutorial on fisher information. *Journal of Mathematical Psychology*, 2017. 3, 4, 13

[52] Xiaosong Ma, Jie Zhang, Song Guo, and Wenchao Xu. Layer-wised model aggregation for personalized federated learning. In *CVPR*, pages 10092–10101, 2022. 1

[53] Andrew Kachites McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3 (2):127–163, 2000. 7, 14

[54] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, pages 1273–1282, 2017. 7

[55] 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*, pages 1273–1282. PMLR, 2017. 1

[56] Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. *arXiv preprint arXiv:2002.05287*, 2020. 3

[57] Athanasios Psaltis, Anestis Kastellos, Charalampos Z Patrikakis, and Petros Daras. Fedlid: Self-supervised federated learning for leveraging limited image data. In *CVPR*, pages 1039–1048, 2023. 3

[58] Fan Qi and Shuai Li. Adaptive hyper-graph aggregation for modality-agnostic federated learning. In *CVPR*, pages 12312–12321, 2024. 3

[59] Zixuan Qin, Liu Yang, Qilong Wang, Yahong Han, and Qinghua Hu. Reliable and interpretable personalized federated learning. In *CVPR*, pages 20422–20431, 2023. 3

[60] Liangqiong Qu, Yuyin Zhou, Paul Pu Liang, Yingda Xia, Feifei Wang, Ehsan Adeli, Li Fei-Fei, and Daniel Rubin. Rethinking architecture design for tackling data heterogeneity in federated learning. In *CVPR*, pages 10061–10071, 2022. 1

[61] Wei Shao, YangYang Shi, Daoqiang Zhang, JunJie Zhou, and Peng Wan. Tumor micro-environment interactions guided graph learning for survival analysis of human cancers from whole-slide pathological images. In *CVPR*, pages 11694–11703, 2024. 3

[62] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. In *NeurIPS*, 2018. 7, 14

[63] Guangyu Sun, Matias Mendieta, Jun Luo, Shandong Wu, and Chen Chen. Fedperfix: Towards partial model personalization of vision transformers in federated learning. In *CVPR*, pages 4988–4998, 2023. 3

[64] Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. In *NeurIPS*, pages 21394–21405, 2020. 3

[65] Yue Tan, Yixin Liu, Guodong Long, Jing Jiang, Qinghua Lu, and Chengqi Zhang. Federated learning on non-iid graphs via structural knowledge sharing. In *AAAI*, pages 9953–9961, 2023. 2, 3, 7

[66] Jianheng Tang, Jiajin Li, Ziqi Gao, and Jia Li. Rethinking graph neural networks for anomaly detection. In *ICML*, 2022. 4

[67] Kiran K Thekumparampil, Chong Wang, Sewoong Oh, and Li-Jia Li. Attention-based graph neural network for semi-supervised learning. *arXiv preprint arXiv:1803.03735*, 2018. 3

[68] Nguyen Anh Tu, Assanali Abu, Nartay Aikyn, Nursultan Makhanov, Min-Ho Lee, Khiem Le-Huy, and Kok-Seng Wong. Fedfslar: A federated learning framework for few-shot action recognition. In *CVPR*, pages 270–279, 2024. 3

[69] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017. 3

[70] Guancheng Wan, Wenke Huang, and Mang Ye. Federated graph learning under domain shift with generalizable prototypes. In *AAAI*, 2024. 2, 3, 7

[71] Chen Wang, Yuheng Qiu, Dasong Gao, and Sebastian Scherer. Lifelong graph learning. In *CVPR*, pages 13719–13728, 2022. 3

[72] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. In *NeurIPS*, pages 7611–7623, 2020. 3, 7

[73] Jinyan Wang, Qiyu Li, Yuhang Hu, Xianxian Li, et al. A privacy preservation framework for feedforward-designed convolutional neural networks. *Neural Networks*, pages 14–27, 2022. 3

[74] Lei Wang, Min Dai, Jianan He, and Jingwei Huang. Regularized primitive graph learning for unified vector mapping. In *CVPR*, pages 16817–16826, 2023. 1

[75] Yuan Wang, Kun Yu, Chen Chen, Xiyuan Hu, and Silong Peng. Dynamic graph learning with content-guided spatial-frequency relation reasoning for deepfake detection. In *CVPR*, pages 7278–7287, 2023. 1

[76] Yuan Wang, Huazhu Fu, Renuga Kanagavelu, Qingsong Wei, Yong Liu, and Rick Siow Mong Goh. An aggregation-free federated learning for tackling data heterogeneity. In *CVPR*, pages 26233–26242, 2024. 3

[77] Zheng Wang, Xiaoliang Fan, Zhaopeng Peng, Xueheng Li, Ziqi Yang, Mingkuan Feng, Zhicheng Yang, Xiao Liu, and Cheng Wang. Flgo: A fully customizable federated learning platform, 2023. 7

[78] Xin Wei, Ruixuan Yu, and Jian Sun. View-gcn: View-based graph convolutional network for 3d shape analysis. In *CVPR*, pages 1850–1859, 2020. 1

[79] Jie Wen, Chengliang Liu, Gehui Xu, Zhihao Wu, Chao Huang, Lunke Fei, and Yong Xu. Highly confident local structure based consensus graph learning for incomplete multi-view clustering. In *CVPR*, pages 15712–15721, 2023. 1

[80] Xinghao Wu, Xuefeng Liu, Jianwei Niu, Guogang Zhu, and Shaojie Tang. Bold but cautious: Unlocking the potential of personalized federated learning through cautiously aggressive collaboration. In *CVPR*, pages 19375–19384, 2023. 3

[81] Xinghao Wu, Xuefeng Liu, Jianwei Niu, Guogang Zhu, and Shaojie Tang. Bold but cautious: Unlocking the potential of personalized federated learning through cautiously aggressive collaboration. In *ICCV*, 2023. 3

[82] Han Xie, Jing Ma, Li Xiong, and Carl Yang. Federated graph classification over non-iid graphs. In *NeruIPS*, 2021. 2, 3

[83] An Xu, Wenqi Li, Pengfei Guo, Dong Yang, Holger R Roth, Ali Hatamizadeh, Can Zhao, Daguang Xu, Heng Huang, and Ziyue Xu. Closing the generalization gap of cross-silo federated medical image segmentation. In *CVPR*, pages 20866–20875, 2022. 1

[84] Qiangeng Xu, Xudong Sun, Cho-Ying Wu, Panqu Wang, and Ulrich Neumann. Grid-gcn for fast and scalable point cloud learning. In *CVPR*, pages 5661–5670, 2020. 3

[85] Yichao Yan, Qiang Zhang, Bingbing Ni, Wendong Zhang, Minghao Xu, and Xiaokang Yang. Learning context graph for person search. In *CVPR*, pages 2158–2167, 2019. 3

[86] Huanding Zhang, Tao Shen, Fei Wu, Mingyang Yin, Hongxia Yang, and Chao Wu. Federated graph learning – a position paper. *arXiv preprint arXiv:2105.11099*, 2021. 2

[87] Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Fedala: Adaptive local aggregation for personalized federated learning. In *AAAI*, pages 11237–11244, 2023. 3

[88] Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Fedcp: Separating feature information for personalized federated learning via conditional policy. In *SIGKDD*, pages 3249–3261, 2023. 3

[89] Ke Zhang, Carl Yang, Xiaoxiao Li, Lichao Sun, and Siu Ming Yiu. Subgraph federated learning with missing neighbor generation. In *NeurIPS*, pages 6671–6682, 2021. 3, 7

[90] Joshua C Zhao, Ahaan Dabholkar, Atul Sharma, and Saurabh Bagchi. Leak and learn: An attacker's cookbook to train using leaked data from federated learning. In *CVPR*, pages 12247–12256, 2024. 3

[91] Weixi Zhao, Weiqiang Wang, and Yunjie Tian. Graformer: Graph-oriented transformer for 3d pose estimation. In *CVPR*, pages 20438–20447, 2022. 1

[92] Xin Zheng, Yi Wang, Yixin Liu, Ming Li, Miao Zhang, Di Jin, Philip S Yu, and Shirui Pan. Graph neural networks for graphs with heterophily: A survey. *arXiv preprint arXiv:2202.07082*, 2022. 3

[93] Zhengyi Zhong, Weidong Bao, Ji Wang, Xiaomin Zhu, and Xiongtao Zhang. Flee: A hierarchical federated learning framework for distributed deep neural network over cloud, edge, and end device. *ACM Transactions on Intelligent Systems and Technology (TIST)*, pages 1–24, 2022. 3

[94] Zhengyi Zhong, Ji Wang, Weidong Bao, Jingxuan Zhou, Xiaomin Zhu, and Xiongtao Zhang. Semi-hfl: semi-supervised federated learning for heterogeneous devices. *Complex & Intelligent Systems*, pages 1995–2017, 2023. 3

[95] Tianfei Zhou and Ender Konukoglu. Fedfa: Federated feature augmentation. In *ICLR*, 2023. 3, 7

[96] Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. *NeurIPS*, 33:7793–7804, 2020. 3

[97] Jiong Zhu, Ryan A Rossi, Anup Rao, Tung Mai, Nedim Lipka, Nesreen K Ahmed, and Danai Koutra. Graph neural networks with heterophily. In *AAAI*, 2021. 3

[98] Jiong Zhu, Junchen Jin, Donald Loveland, Michael T Schaub, and Danai Koutra. How does heterophily impact the robustness of graph neural networks? theoretical connections and practical implications. 2022. 3

A. Proof: Correlation Between High-frequency Area S_{high} and Homophily Level h

We aim to prove the negative correlation between the high-frequency area of the graph spectrum, denoted as S_{high}, and the homophily level h. To begin with, according to the graph Laplacian energy, we have:

$$x^T \mathbf{L} x = \sum_{(u,v)\in E} (x_u - x_v)^2 = \mathcal{E}_{\text{diff}},$$

where $\mathcal{E}_{\text{diff}}$ represents the sum of signal differences across edges. The expected value of the sum is given by:

$$\mathbb{E}[\mathcal{E}_{\text{diff}}] = \mathbb{E}\left[\sum_{(u,v)\in E} (x_u - x_v)^2\right].$$

Assuming that the difference $(x_u - x_v)^2$ between connected nodes u and v in graph G is independent and identically distributed (i.i.d) and that its expectation is influenced by the homophily level h, we have:

$$\mathbb{E}[(x_u - x_v)^2 \mid \mathbf{y}_u = \mathbf{y}_v] \ll \mathbb{E}[(x_u - x_v)^2 \mid \mathbf{y}_u \neq \mathbf{y}_v],$$

where $\mathbf{y}_u$ and $\mathbf{y}_v$ denote the labels of nodes u and v, respectively. Therefore, the overall expectation of the signal difference is approximately:

$$\mathbb{E}[(x_u - x_v)^2] \approx (1-h) \cdot \mathbb{E}[(x_u - x_v)^2 \mid \mathbf{y}_u \neq \mathbf{y}_v].$$

$$\mathbb{E}[\mathcal{E}_{\text{diff}}] \approx |\mathcal{E}| \cdot (1-h) \cdot \mathbb{E}[(x_u - x_v)^2 \mid \mathbf{y}_u \neq \mathbf{y}_v],$$

where $|\mathcal{E}|$ is the number of edges in the graph. Assuming that the signal x has length F and its elements x_f are i.i.d. following a normal distribution $\mathcal{P}(0, \sigma^2)$, we have:

$$\mathbb{E}[x^T x] = \mathbb{E}\left[\sum_{f=1}^{F} x_f^2\right] = \sum_{f=1}^{F} \mathbb{E}[x_f^2] = F\sigma^2.$$

Consequently, the expected value of S_{high} is expressed as:

$$\mathbb{E}[S_{\text{high}}] = \mathbb{E}\left[\frac{\mathcal{E}_{\text{diff}}}{F\sigma^2}\right] \approx \frac{|\mathcal{E}| \cdot (1-h) \cdot \mathbb{E}[(x_u - x_v)^2 | \mathbf{y}_u \neq \mathbf{y}_v]}{F\sigma^2},$$

$$\mathbb{E}[S_{\text{high}}] \propto (1-h),$$

which shows the negative correlation between the high-frequency area S_{high} and the homophily level h.

B. Fisher Information Matrix

Parameter sensitivity is quantified using the Fisher Information Matrix (FIM) [1, 16, 51], which provides a measure of the amount of information that the observed data carries about the GNN parameters. Let $\mathcal{G}$ represent the graph data, and let θ represent the parameters of the distribution $p(\mathcal{G}|\theta)$ underlying the GNN. The FIM is defined as:

$$\mathbf{I}(\theta) = \mathbb{E}_{\mathcal{G}\sim p(\mathcal{G}|\theta)}\left[\left(\frac{\partial \log p(\mathcal{G}|\theta)}{\partial \theta}\right)\left(\frac{\partial \log p(\mathcal{G}|\theta)}{\partial \theta}\right)^{\top}\right], \tag{13}$$

where the expectation is taken over the distribution of the observed graph data. The FIM essentially captures the curvature of the likelihood function in the parameter space, indicating how sensitive the likelihood is to small changes in the parameters. This sensitivity is crucial in various applications, including parameter estimation, uncertainty quantification, and model selection.

Nevertheless, the FIM for GNNs is computationally infeasible due to the high dimensionality of the parameter space. Therefore, an approximation is leveraged in our method. Specifically, the FIM can be approximated using an empirical distribution derived from the observed data. This approximation, particularly its diagonal form, simplifies the computation significantly:

$$\mathbf{I}_{\text{diag}}(\theta) \approx \mathbb{E}_{\mathcal{G}\sim p(\mathcal{G}|\theta)}\left[\left(\frac{\partial \log p(\mathcal{G}|\theta)}{\partial \theta}\right)^2\right]. \tag{14}$$

To further motivate, we consider the effect of small perturbations in the model parameters on the output of the GNN. This relationship can be rigorously quantified using the Kullback-Leibler (KL) divergence, which measures the difference between the original distribution and a perturbed distribution. Specifically, if the parameters are perturbed by a small amount δ, the second-order Taylor expansion of the KL divergence leads to the following approximation:

$$\mathbb{E}_{\mathcal{G}}\left[D_{\text{KL}}\left(p(\mathcal{G}|\theta) \| p(\mathcal{G}|\theta+\delta)\right)\right] \approx \frac{1}{2}\delta^{\top}\mathbf{I}(\theta)\delta, \tag{15}$$

where δ is a small perturbation in the parameter space. This result shows that the FIM not only measures the sensitivity of the likelihood function to parameter changes but also quantifies the expected change in model output as a result of these parameter perturbations. In other words, the FIM provides a fundamental connection between the parameter space and the output space of the GNN.

In summary, the Fisher Information Matrix serves as a measure of parameter sensitivity for GNNs. By approximating the FIM, we can quantify the posterior parameter sensitivity bias between a client and the globe. Correspondingly, a posterior bias-driven aggregation is proposed for reasonably measuring client contribution and enabling the global GNN to benefit more from those with lower biases.

C. Contextual Stochastic Block Model

In this research, we utilize the Contextual Stochastic Block Model (CSBM) [12] to generate synthetic graphs. These

graphs feature variable edge probabilities both within and between different groups. The primary idea is that nodes of the same class share a uniform feature distribution. The resulting graph is denoted as $\mathcal{G} \sim \text{CSBM}(N, F, \sigma, \mu)$, where n is the total number of nodes, F represents the feature dimension, and σ and μ are the hyperparameters. These hyperparameters, σ and μ, control the influence of the graph structure and node features respectively. We consider two equal-sized classes, c_1 and c_0, each with $N/2$ nodes.

The CSBM produces features of a node u as follow:

$$\mathbf{x}_u = \sqrt{\frac{\mu}{N}}\mathbf{y}_u\zeta + \frac{q_u}{\sqrt{F}}, \tag{16}$$

where $\mathbf{y}_u \in \{-1, +1\}$ indicates the class label of node v_u, μ represents the mean of the Gaussian distribution, $\zeta \sim \mathcal{N}(0, I/F)$, and q_u consists of independently distributed standard normal variables. The average degree of the generated graph is denoted as d, and the adjacency matrix $\mathbf{A}$ for the CSBM graph is defined by:

$$P(\mathbf{A}_{uv} = 1) = \begin{cases} \frac{1}{N}(d + \sigma\sqrt{d}) & \text{if } \mathbf{y}_u = \mathbf{y}_v \\ \frac{1}{N}(d - \sigma\sqrt{d}) & \text{if } \mathbf{y}_u \neq \mathbf{y}_v. \end{cases} \tag{17}$$

The level of homophily h can be tuned by adjusting $\sigma = \sqrt{d(2h-1)}$, within the range $-\sqrt{d} \leq \sigma \leq \sqrt{d}$. A fully heterophilic graph is represented by $\sigma = -\sqrt{d}$, while a fully homophilic graph is described by $\sigma = \sqrt{d}$.

D. Datasets

We perform experiments on node classification tasks on both homophilic and heterophilic graph datasets to demonstrate the superiority of our proposed method FedSPA.

Cornell, Wisconsin: These datasets are subsets of the WebKB dataset [11]. The WebKB dataset was introduced in 1998, comprising web pages from the computer science departments of various universities, including Cornell University and the University of Wisconsin. These pages are categorized into five classes: student, faculty, course, project, and staff. In this dataset, each node represents a webpage, and edges denote hyperlinks between them. The dataset is commonly used for tasks such as webpage classification and link prediction, serving as a benchmark for evaluating machine learning models in graph-based learning scenarios.

Cora: The Cora dataset [53] is a widely used benchmark in machine learning and graph analysis, particularly for tasks like node classification and link prediction. It comprises 2,708 scientific publications categorized into seven classes: Case-Based Reasoning, Genetic Algorithms, Neural Networks, Probabilistic Methods, Reinforcement Learning, Rule Learning, and Theory. Each publication is represented as a node, and edges between nodes denote citation relationships, forming a citation network with 5,429 links. Node features are binary vectors indicating the presence or absence of 1,433 unique words from the publication's content. This dataset is instrumental in evaluating the performance of various graph-based algorithms and models.

Coauthor-CS, Coauthor-Physics: The Coauthor-CS and Coauthor-Physics datasets [62] are derived from the Microsoft Academic Graph and were used in the KDD Cup 2016 challenge. In these datasets, nodes represent authors, and edges denote co-authorship relationships. The Coauthor-CS dataset contains 18,333 nodes and 81,894 edges, with node features representing the keywords of papers authored by each individual and labels indicating classification into 15 fields of study. The Coauthor-Physics dataset includes 34,493 nodes and 247,962 edges, with similar node features and labels representing classification into 5 main research areas. These datasets are widely used for node classification tasks in graph neural network research as standard benchmarks to evaluate model performance.

Minesweeper: The Minesweeper dataset [5] is a synthetic graph dataset inspired by the classic Minesweeper game. In this dataset, the graph is structured as a regular 100x100 grid, where each node represents a cell connected to its eight neighboring nodes, except for edge nodes which have fewer neighbors. Approximately 20% of the nodes are randomly designated as mines. The primary task is to predict which nodes contain mines. Node features are one-hot encoded to represent the number of neighboring mines. However, for a randomly selected 50% of the nodes, these features are unknown, indicated by a separate binary feature. This dataset is commonly used to evaluate the performance of GNNs under heterophily.

ArXiv-year: The arXiv-year dataset [25] is a benchmark dataset designed for graph learning tasks, consisting of a citation network with nodes, edges, node features, and temporal labels. Each node represents a paper from the arXiv repository, and the edges denote citation relationships between papers, forming an undirected graph. The node features are typically embeddings related to the content of the paper, such as text-based representations. The node labels correspond to the publication year of each paper.