LET PERSISTENCE SHEPHERD YOUR SHEAVES

Anonymous authors
Paper under double-blind review

ABSTRACT

Persistence-augmented graph neural networks enrich message passing with global topological information via persistent homology (PH). While some existing PH-based topological descriptors make the underlying graph filtration learnable, they fix a coefficient system. This imposes an unnecessary constraint that may limit the expressive power of PH-based descriptors and potentially hinder downstream predictive performance. To overcome this, we introduce ShePHerD as a model that learns the coefficient system with cellular sheaves, where we persist the resulting sheaf cohomology and enrich persistence events with the spectra of the sheaf Laplacian. Notably, we theoretically prove that integrating persistence sheaf cohomology into PH diagrams increases expressivity. Furthermore, on a suite of 15 Paulus graphs, the sheaf Laplacian spectra successfully distinguish all 105 non-isomorphic pairs that remain indistinguishable to both our baselines and the folklore 2-WL test. Empirically, ShePHerD outperforms existing baselines and standalone sheaf diffusion models on real-world graph-level prediction tasks. Finally, we introduce ShePHerD++, an enhanced variant that integrates sheaf diffusion over the learned sheaf structure, yielding further performance gains.

1 INTRODUCTION

Message-passing *graph neural networks* (GNNs) are widely used for graph representation learning (Gilmer et al., 2017; Sanchez-Gonzalez et al., 2020). However, standard GNNs have their distinguishing power bounded by the 1-WL test (Xu et al., 2019; Morris et al., 2019). To overcome this limitation, a popular approach augments them with *persistent homology* (PH) (Edelsbrunner et al., 2002), incorporating global topological information that vanilla message passing cannot capture (Horn et al., 2022; Rieck et al., 2019). This search for greater distinguishing power has also motivated the development of more expressive PH-based graph descriptors (Immonen et al., 2023; Ballester and Rieck, 2025; Ji et al., 2026; 2025). For instance, Immonen et al. (2023) characterize the distinguishing power of PH on graphs and propose enriching persistence diagrams with information derived from node/edge features. SpectRe (Ji et al., 2025) further incorporates spectral information from filtered subgraphs to enhance expressivity. Despite these advances, existing approaches share a common trait: they compute homology over a fixed *coefficient* field, typically $\mathbb{Z}_2$ or $\mathbb{R}$.

An important but often overlooked aspect of homology is that its coefficient system is determined not only by the coefficient spaces, but also by the maps connecting them. Defaulting to standard homology implicitly assumes these maps are identity maps. This structure is made precise by cellular sheaves: given a graph G, cellular sheaves attach vector spaces $\mathbb{R}^d \cong \mathcal{F}(v)$ and $\mathbb{R}^d \cong \mathcal{F}(e)$, called *stalks*, to every vertex v and every edge e, and linear maps $\mathcal{F}_{v \trianglelefteq e} : \mathcal{F}(v) \to \mathcal{F}(e)$ to each incidence $v \trianglelefteq e$, called *restriction maps*. The trivial sheaf, where $d = 1$ and all restriction maps are identities, recovers the standard coefficient space $\mathbb{R}$. From this perspective, fixing all restriction maps to identities imposes an unnecessary constraint that may limit the expressive power of PH-based descriptors and potentially hinder downstream predictive performance.

The cornerstone of our work is that sheaves provide a flexible framework for generalized coefficient systems. Building on this perspective, *we demonstrate the benefits of learning the coefficient system through the sheaf's restriction maps*. Towards this goal, we adopt a learnable $O(d)$ paradigm, parameterizing the restriction maps as orthogonal matrices to balance computational efficiency and performance (Bodnar et al., 2022). While most sheaf-based models (Bamberger et al., 2025; Ribeiro et al., 2026; Fiorini et al., 2026) learn the restriction maps $\mathcal{F}_{v \trianglelefteq e}$, we instead learn the induced *parallel transport* $R_e = \mathcal{F}^{\top}_{v \trianglelefteq e} \mathcal{F}_{u \trianglelefteq e} : \mathcal{F}(u) \to \mathcal{F}(v)$ across each oriented edge $e = (u, v)$, thus directly capturing the message flow between adjacent nodes. In this setting, zeroth sheaf cohomology $H^0(G, \mathcal{F})$ is the space of globally consistent assignments, that is, sections that agree under transport along every edge,

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.

Table 1: Overview of main contributions.

Contribution	Reference
Construction of ShePHerD	Definition 3.4
ShePHerD recovers SpectRe and RePHINE on the trivial sheaf	Proposition 3.5
ShePHerD is isomorphism invariant	Theorem 3.6
ShePHerD $\succ$ RePHINE, SpectRe and edge-aware 1-WL on attributed graphs	Theorem 3.7
ShePHerD separates all 105 pairs of Paulus graphs	Table 5
Graph-level prediction on seven benchmarks	Section 4

and so $H^0(G, \mathcal{F})$ depends on the learned R_e. Notably, if the stalks are all 1-dimensional and $R_e = 1$ on every edge, then the sheaf cohomology $H^0(G, \mathcal{F})$ recovers the persistent homology $H_0(G)$.

Consequently, learning parallel transport from data induces a (co)homological structure that arises from the node and edge features of the input graph. We persist H^0 over learnable filtrations (Hofer et al., 2020), but change the coefficient to a sheaf learned end to end, which gives our **She**af **Per**sistent (co)**H**omology **D**iagrams (ShePHerD) model. Our key insight is that a coefficient learned from data, rather than fixed on $\mathbb{Z}_2$, enhances the performance of PH-based GNNs. On the theoretical side, we prove that ShePHerD is invariant to graph isomorphism and is strictly more expressive than RePHINE (Immonen et al., 2023) and SpectRe (Ji et al., 2025). Complementing the theory, our experiments show that the Sheaf Persistent cohomology Diagram, together with the spectrum of the sheaf Laplacian, improve graph-level prediction tasks (Table 2). Furthermore, on the BREC benchmark (Wang and Zhang, 2024), spectral features derived from the sheaf Laplacian successfully distinguish 311 of 400 pairs against 216 for SpectRe, and 164 for RePHINE (Table 4). We overview our contributions in Table 1.

Related work. Persistent cellular sheaf cohomology on filtered spaces was introduced by Yegnesh (2016) and developed systematically by Russold (2022), while Wei and Wei (2025) introduced persistent sheaf Laplacians. These works develop persistence for prescribed sheaf data. In contrast, ShePHerD learns the coefficient system end-to-end from graph attributes and couples it to the learnable filtration mechanisms used by RePHINE and SpectRe. Related but yet with no learning component, Boissonnat and Maria (2019) compute persistence over several fields in one reduction and recover the prime divisors of the torsion of the integral homology. That axis is empty on graphs, since the homology of a graph is free in degrees 0 and 1 and its Betti numbers do not depend on the field (Proposition A.8), whereas our learned coefficient system changes the operator itself.

2 Background

In this section, we set the basic notation used throughout the work and overview the necessary background on cellular sheaves, persistent homology, and the Weisfeiler–Leman isomorphism test.

Notation. Let $G = (V, E, X, Z)$ be a finite attributed graph, where $X = (x_v)_{v \in V}$ and $Z = (z_e)_{e \in E}$ denote the vertex and undirected edge attributes. When a graph has no attributes, we regard X and Z as constant. An *attribute-preserving graph isomorphism* $\pi : G \to H$ is a bijection on vertices that preserves adjacency and the corresponding vertex and edge attributes. A graph descriptor $\mathfrak{D}$ is *isomorphism invariant* if $G \cong H$ implies $\mathfrak{D}(G) = \mathfrak{D}(H)$. For an undirected graph G, let $\vec{E}(G) = \{(u, v) : \{u, v\} \in E\}$ denote the set containing both orientations of every edge. We write L_G for the ordinary graph Laplacian and $\Delta_G = I - D^{-1/2} A D^{-1/2}$ for its normalized form, where A and D are the adjacency and (diagonal) degree matrices of G. For a cellular sheaf $\mathcal{F}$, we write $L_{\mathcal{F}}$ and $\Delta_{\mathcal{F}}$ for the sheaf Laplacian and its normalized form, respectively.

2.1 Cellular Sheaves

A cellular sheaf $\mathcal{F}$ defined over G equips each vertex $v \in V$ and edge $e \in E$ with vector spaces $\mathcal{F}(v)$ and $\mathcal{F}(e)$, the *stalks*, alongside linear *restriction maps* $\mathcal{F}_{v \trianglelefteq e} : \mathcal{F}(v) \to \mathcal{F}(e)$ for every incident pair $v \trianglelefteq e$. The *trivial sheaf* is the particular case where $d = 1$ and all restriction maps are the identity.

While diffusion in GNNs is dictated by the graph Laplacian, Sheaf Neural Networks (SNNs) rely on the sheaf Laplacian (Hansen and Ghrist, 2019; Bodnar et al., 2022) or its adaptions (Barbero

et al., 2022; Bamberger et al., 2025; Ribeiro et al., 2026). To define it, we define the 0-cochain and 1-cochain spaces, respectively, $C^0(G,\mathcal{F}) = \bigoplus_{v\in V} \mathcal{F}(v)$ and $C^1(G,\mathcal{F}) = \bigoplus_{e\in E} \mathcal{F}(e)$.

Sheaf Laplacian. Given an arbitrary edge orientation $e = v \to u$, we introduce the coboundary operator $\delta : C^0(G,\mathcal{F}) \to C^1(G,\mathcal{F})$, acting on $\mathbf{X} \in C^0(G,\mathcal{F})$ as $(\delta\mathbf{X})_e = \mathcal{F}_{u\trianglelefteq e}\mathbf{x}_u - \mathcal{F}_{v\trianglelefteq e}\mathbf{x}_v$. Then we define the *sheaf Laplacian* by $L_{\mathcal{F}} = \delta^\top\delta$. Node-wise, it is expressed by: $L_{\mathcal{F}}(\mathbf{X})_v = \sum_{v,u\trianglelefteq e} \mathcal{F}^\top_{v\trianglelefteq e}\left(\mathcal{F}_{v\trianglelefteq e}\mathbf{x}_v - \mathcal{F}_{u\trianglelefteq e}\mathbf{x}_u\right),\ \forall v \in V$. Its normalized variant is given by $\Delta_{\mathcal{F}} = D^{-1/2}L_{\mathcal{F}}D^{-1/2}$, where D represents the block-diagonal of $L_{\mathcal{F}}$. If the sheaf is trivial, the sheaf Laplacian coincides with the graph Laplacian.

Neural Sheaf Diffusion. The Neural Sheaf Diffusion (NSD) model proposed by Bodnar et al. (2022) can be described as follows: given a feature matrix $X = X^{(0)} \in \mathbb{R}^{n\times s}$, the model maps it to the sheaf stalks through an embedding layer, which yields $\mathbf{X}^{(0)} \in \mathbb{R}^{nd\times f}$. If $\mathbf{X}^{(t)} \in \mathbb{R}^{nd\times f}$ is the representation at layer t, the model can be summarized by

$$\mathbf{X}^{(t+1)} = \mathbf{X}^{(t)} - \sigma\left(\Delta_{\mathcal{F}^{(t)}}(I_n \otimes W_1^{(t)})\mathbf{X}^{(t)}W_2^{(t)}\right) \tag{1}$$

where σ is a non-linear activation function, $W_1^{(t)} \in \mathbb{R}^{d\times d}$ and $W_2^{(t)} \in \mathbb{R}^{f\times f}$ are learnable weight matrices at layer t. The restriction maps defining the sheaf $\mathcal{F}^{(t)}$ are learned at each layer through $\mathcal{F}^{(t)}_{u\trianglelefteq e} = \mathrm{MLP}\left(\mathbf{x}_u^{(t)} \| \mathbf{x}_v^{(t)}\right)$, for $v \trianglelefteq e$.

2.2 Persistent Homology on Graphs

The core idea of persistent homology is to track how topological features evolve throughout a filtration, accounting for their appearance and disappearance. A filtration of G is a nested sequence of subgraphs $G_0 \subseteq \cdots \subseteq G_m = G$, obtained from a pair of functions $f = (f_V, f_E)$ that assign values to vertices and edges.

Birth and Death. We employ an edge-based filtration where $G_0 = (V, \emptyset)$ already contains all vertices; thus f_V supplies the decoration α rather than a vertex insertion time (Appendix B). In this setting, a vertex, which represents 0-dimensional information, is born when it first appears in the filtration. When two connected components represented by two vertices v and w merge, a decision rule kills off one of the vertices and marks the remaining one to represent the new connected component. A cycle, representing 1-dimensional information, is born when it appears in the filtration and never dies, since a graph has no 2-cells. The persistence pair of a feature is the tuple (b, d) of its birth and death, with $d = \infty$ if it never dies.

RePHINE (Immonen et al., 2023) decorates each pair with two additional coordinates obtained from the filtration, $\alpha(v) = f_V(x_v)$ and $\gamma(v) = \min_{e\ni v} f_E(e)$, demonstrating that the resulting diagrams are strictly more expressive than ordinary persistence diagrams. SpectRe (Ji et al., 2025) further augments to each tuple the nonzero spectrum of the graph Laplacian of the connected component in which the event occurs. We provide the precise constructions in Appendix B.

2.3 Weisfeiler–Leman tests

The 1-WL test (Weisfeiler and Lehman, 1968) refines an initial coloring of the vertices by replacing the color of v with a hash of its own color and the multiset of the colors of its neighbors. Two graphs are distinguished when the multisets of colors differ at some iteration. Xu et al. (2019) proved that message-passing GNNs are bounded by this test (Xu et al., 2019). When graphs carry edge attributes, the refinement can also read the attribute of each incident edge, which gives the edge-aware variant used in Theorem 3.7. The folklore 2-WL test (Cai et al., 1992; Maron et al., 2019) instead colors pairs of vertices and is strictly stronger than 1-WL. We employ this stronger test in Theorem 3.8, where it gives the same stable coloring on every graph of the family. Appendix B.8 states both refinements.

3 Expressive Sheaf-based Persistence Diagrams

In this section, we introduce our sheaf-based persistence diagram (ShePHerD), analyze its theoretical properties, and propose a sheaf diffusion model incorporating our novel topological descriptors (ShePHerD++). As a preliminary step, we start with a discussion on how to learn the sheaf structure employed in ShePHerD. Proofs are given in Appendix B.

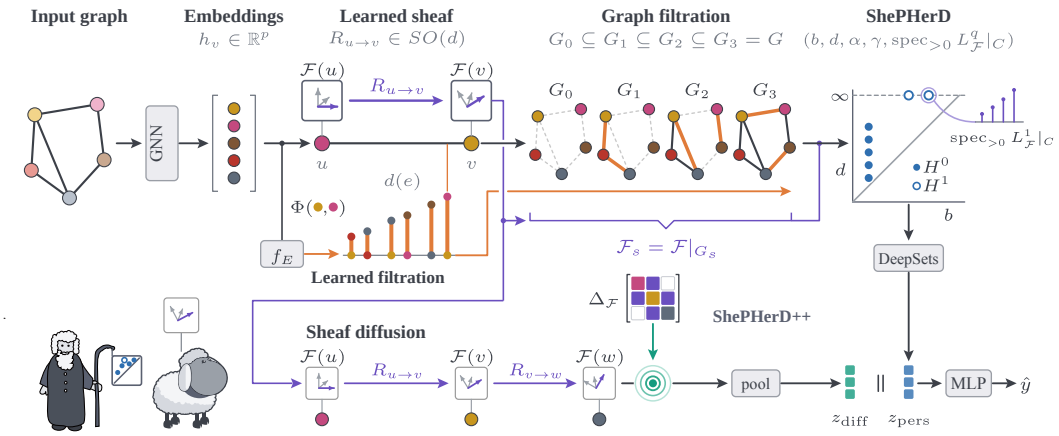


Figure 1: Overview of ShePHerD. A GNN produces node embeddings h_v, from which we learn both the edge filtration f_E, and the orthogonal transport $R_{u\to v} \in SO(d)$ of a cellular sheaf $\mathcal{F}$. The filtration gives $G_0 \subseteq \cdots \subseteq G_3 = G$, and the same sheaf is restricted to every level, $\mathcal{F}_s = \mathcal{F}|_{G_s}$. Persisting its cohomology returns the tuples $(b, d, \alpha, \gamma, \mathrm{spec}_{>0} L_{\mathcal{F}}|_C)$, where C is the connected component of the event: all degree-0 classes are born at G_0 (the stem plot is the spectrum carried by the circled degree-1 event). A DeepSets readout gives z_{pers}. ShePHerD++ additionally diffuses the stalks with the sheaf Laplacian $\Delta_{\mathcal{F}}$ using the same transport maps, which gives z_{diff}; later concatenated. We draw $\Delta_{\mathcal{F}}$ for the three stalks shown in that panel, u, v and w, so the array is 3×3 and each cell stands for a $d \times d$ block: a diagonal cell takes the colour of its own vertex, an off-diagonal cell is purple when the two vertices are adjacent, since that block holds the transport between them and is drawn in the same purple as the arrows, and white when they are not.

3.1 Connection assignments

Firstly, we describe how we learn the transport map $R_{u\to v} : \mathcal{F}(u) \to \mathcal{F}(v)$ that regulates the information flow between u and v. We introduce two methods: i) the feature-conditioned connection, and ii) the spectral connection. Notably, both are *compatible with isomorphisms*, which is an essential property to show that ShePHerD is graph isomorphism invariant (Theorem 3.6).

As an intermediate step, we define an *orthogonal connection* of stalk dimension d on G as a map $R : \vec{E}(G) \to O(d)$ satisfying $R_{v\to u} = R^\top_{u\to v}$ for every $\{u, v\} \in E(G)$. Note that for orthogonal cellular sheaves this is the transport representation of the restriction maps, $R_{u\to v} = \mathcal{F}^\top_{v\trianglelefteq e}\mathcal{F}_{u\trianglelefteq e}, e = \{u, v\}$. Importantly, this construction provides that the corresponding coboundary block can be written as $(\delta_R x)_e = x_v - R_{u\to v}x_u$ and, consequently, the persistent cohomology and the sheaf Laplacian quantities are expressed directly in terms of R.

Definition 3.1. A *connection assignment* associates to each graph G a graph $\widehat{G} \cong G$, a stalk dimension $d(G)$, and matrices $R^G_{u\to v} \in O(d(G))$ for every $(u, v) \in \vec{E}(\widehat{G})$ satisfying $R^G_{v\to u} = (R^G_{u\to v})^\top$. We write $\mathcal{C}_\theta(G) = (\widehat{G}, d(G), R^G_\theta)$. We say that a connection assignment is *compatible with isomorphisms* if $G \cong H$ implies $d(G) = d(H)$ and there are an isomorphism $\widehat{\pi} : \widehat{G} \to \widehat{H}$ and orthogonal matrices $Q_v \in O(d(G))$ such that $R^H_{\widehat{\pi}(u)\to\widehat{\pi}(v)} = Q_v R^G_{u\to v} Q_u^\top$ for every $(u, v) \in \vec{E}(\widehat{G})$.

The transports are compared up to orthogonal changes of stalk coordinates, which send δ to $Q\delta P^\top$ and conjugate the sheaf Laplacians, leaving ranks, cohomology, spectra and hence the descriptor unchanged; every $Q_v = I$ recovers equality of the matrices. Of the two assignments below, the feature-conditioned one has $d(G) \equiv 3$ and the spectral one $d(G) = \dim E_G$.

Feature-conditioned connection. Assume a permutation-equivariant graph encoder outputs $h_v \in \mathbb{R}^p$, and an undirected edge attribute $z_{uv} = z_{vu} \in \mathbb{R}^q$. Using a shared map $\phi_\theta : \mathbb{R}^{2p+q} \to \mathbb{R}^{\binom{d}{2}}$, we define

$$A_{u\to v} = \mathrm{skew}\left(\phi_\theta([h_u \| h_v \| z_{uv}]) - \phi_\theta([h_v \| h_u \| z_{uv}])\right), \quad R_{u\to v} = \exp(A_{u\to v}) \in SO(d). \quad (2)$$

Here, skew : $\mathbb{R}^{\binom{d}{2}} \to \mathfrak{so}(d)$ places the input's entries in the upper triangle, their negatives in the lower triangle, and zeros on the diagonal of the output matrix. Note also that, since $A_{v\to u} = -A_{u\to v}$, it follows that $R_{v\to u} = R_{u\to v}^{\top}$.

Proposition 3.2. *The feature-conditioned connection assignment is compatible with isomorphisms.*

Spectral connection. Let $L_G = D - A$ and order its complete nonzero eigenspaces by increasing multiplicity, breaking ties by decreasing eigenvalue. We add complete eigenspaces in this order until their accumulated dimension is at least three and the endpoint projections of every edge are non-collinear. More precisely, if E_G is the selected eigenspace sum, $d = \dim E_G$, $U \in \mathbb{R}^{|V|\times d}$ is any orthonormal basis of E_G, and $p_v = U^{\top} e_v$, we require

$$\frac{\|p_u \wedge p_v\|}{\|p_u\|\,\|p_v\|} > 10^{-6} \qquad \text{for every } \{u, v\} \in E. \tag{3}$$

For every oriented edge (u, v) we then set

$$B_{u\to v} = \frac{p_v p_u^{\top} - p_u p_v^{\top}}{\sqrt{\|p_u\|^2\|p_v\|^2 - \langle p_u, p_v\rangle^2}}, \qquad R_{u\to v} = \exp\Big(\frac{\pi}{2} B_{u\to v}\Big) \in SO(d). \tag{4}$$

The denominator is the area $\|p_u \wedge p_v\|$, so $B_{u\to v}$ is the unit skew generator of the plane spanned by p_u and p_v. Since $B_{v\to u} = -B_{u\to v}$, we have $R_{v\to u} = R_{u\to v}^{\top}$.

Proposition 3.3. *The spectral connection assignment is compatible with isomorphisms.*

This connection takes $\widehat{G} = G$; the angle $\pi/2$ is the same for every graph. Eigenvectors enter only through complete eigenspaces, so another orthonormal basis replaces U by UQ with $Q \in O(d)$ and conjugates every transport by that same Q, which is what Definition 3.1 allows.

3.2 The ShePHerD descriptor

We now present ShePHerD as a method to track how topological features evolve along a filtration, such as standard persistent homology. The difference is that sheaves change the (co)homological groups used in the persistence steps, which we briefly describe here and provide details in Section B.2.

The filtrations. As in RePHINE (Immonen et al., 2023) and SpectRe (Ji et al., 2025), f_E assigns each edge e the filtration value $d(e) = f_E(e)$, inducing $G_0 = (V, \emptyset) \subseteq G_1 \subseteq \cdots \subseteq G_m = G$. For each vertex v, RePHINE additionally defines $\alpha(v) = f_V(x_v)$ and $\gamma(v) = \min_{e\ni v} f_E(e)$, the smallest filtration value among edges incident to v. We retain these decorations in ShePHerD. We learn a single sheaf $\mathcal{F}$ on the terminal graph G and restrict it along the filtration, $\mathcal{F}_s = \mathcal{F}|_{G_s}$. The inclusions $G_s \hookrightarrow G_t$ induce the sheaf-cohomology copersistence module $H^q(G_0; \mathcal{F}_0) \leftarrow \cdots \leftarrow H^q(G_m; \mathcal{F}_m)$, whose interval decomposition yields the birth–death pairs after reindexing. This is the persistent-sheaf construction of Yegnesh (2016); Russold (2022), instantiated here with a learned coefficient system.

Definition 3.4. Given the filtration $f = (f_V, f_E)$ and the restricted sheaves $\mathcal{F}_s = \mathcal{F}|_{G_s}$, the *Sheaf Persistent (co)Homology Diagram* is

$$\mathrm{ShePHerD}(G, \mathcal{F}) = \underbrace{\big\{(b, d, \alpha, \gamma, \mathrm{spec}_{>0}\, L^0_{\mathcal{F}}|_C)\big\}}_{\text{degree 0}} \sqcup \underbrace{\big\{(1, d(e), 0, 0, \mathrm{spec}_{>0}\, L^1_{\mathcal{F}}|_C)\big\}}_{\text{degree 1}},$$

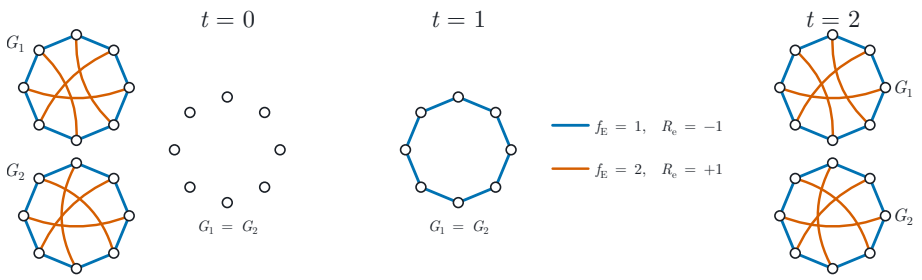
where (b, d) are the birth–death values of the persistent sheaf-cohomology classes, $L^0_{\mathcal{F}} = \delta^{\top}\delta$ and $L^1_{\mathcal{F}} = \delta\delta^{\top}$, and C is the connected component with the respective spectrum.

The pairs are computed from the filtered coboundary matrices, while (α, γ) follow RePHINE's selection rule, extended to rank-d events as detailed in Appendix B. This configuration yields that ShePHerD generalizes SpectRe and RePHINE, as we state bellow:

Proposition 3.5. *If $\mathcal{F}$ is the trivial sheaf, then* $\mathrm{ShePHerD}(G, \mathcal{F})$ *equals SpectRe on the same filtration; dropping the spectral coordinate recovers RePHINE.*

3.3 Expressive Power

Here, we show that ShePHerD is strictly more expressive than RePHINE and SpectRe (Theorem 3.7), and we exhibit a class of graphs, the Paulus graphs, on which its separation power is complete



	$t = 0$	$t = 1$	$t = 2$
ShePHerD(G_1)	$(0, , , ,) \times 8$	$(0, 1, 0, 1, \{0.59, \dots\}) \times 7$, $(0, , 0, 1,), (1, 1, 0, 0, \{0.59, \dots\})$	$(0, 1, 0, 1, \{0.59, \dots\}) \times 7$,**$(0, 2, 0, 1, \{0.59, 0.76, \dots\})$**, $(1, 1, 0, 0, \{0.59, \dots\})$,**$(1, 2, 0, 0, \{0.59, \dots\}) \times 3$**
ShePHerD(G_2)	$(0, , , ,) \times 8$	$(0, 1, 0, 1, \{0.59, \dots\}) \times 7$, $(0, , 0, 1,), (1, 1, 0, 0, \{0.59, \dots\})$	$(0, 1, 0, 1, \{0.59, \dots\}) \times 7$, $(0, \infty, 0, 1, \{2, \dots\})$, $(1, 1, 0, 0, \{0.59, \dots\}), (1, 2, 0, 0, \{2, \dots\}) \times 4$
SpectRe	$(0, , , ,) \times 8$	$(0, 1, 0, 1, \{0.59, \dots\}) \times 7$, $(0, , 0, 1,), (1, 1, 0, 0, \{0.59, \dots\})$	$(0, 1, 0, 1, \{0.59, \dots\}) \times 7$, $(0, \infty, 0, 1, \{2, \dots\})$, $(1, 1, 0, 0, \{0.59, \dots\}), (1, 2, 0, 0, \{2, \dots\}) \times 4$
RePHINE	$(0, , ,) \times 8$	$(0, 1, 0, 1) \times 7$, $(0, , 0, 1), (1, 1, 0, 0)$	$(0, 1, 0, 1) \times 7$, $(0, \infty, 0, 1)$, $(1, 1, 0, 0), (1, 2, 0, 0) \times 4$

Figure 2: The separating pair. Both graphs have two edge levels: the blue 8-cycle enters at $f_E = 1$ and the four chords at $f_E = 2$. At $t = 0$, both sublevels consist of eight isolated vertices; at $t = 1$, both are the same 8-cycle. At $t = 2$, the chords join vertices at cyclic distances $\{3, 3, 4, 4\}$ in G_1 and $\{4, 4, 4, 4\}$ in G_2. RePHINE and SpectRe give the same descriptors at all three levels, while the connection separates the graphs at $t = 2$. Specifically, $R_e = -1$ on blue edges and $R_e = +1$ on orange edges. In G_1, a distance-3 chord forms a cycle with holonomy $(-1)^3 = -1$, giving $\dim H^0(G_1; \mathcal{F}) = 0$. In G_2, every cycle has holonomy $+1$, giving $\dim H^0(G_2; \mathcal{F}) = 1$. Thus the ShePHerD rows agree at $t = 0$ and $t = 1$ and differ at $t = 2$ (green entries).

(Theorem 3.8). Since a descriptor is only useful if isomorphic inputs receive the same value, we first check invariance. The connection is part of the descriptor, so the statement has to cover the way the connection is assigned, and Definition 3.1 is what makes it a single statement for both regimes.

Theorem 3.6. *Let $\rho : G \to G'$ be an isomorphism of attributed graphs, so that ρ preserves vertex and edge attributes, and let the filtration be induced by the attributes, $f'_V \circ \rho = f_V$ and $f'_E \circ \rho = f_E$. If the connection is given by an compatible with isomorphisms assignment, then the filtered cochain complexes of $(G, \mathcal{F})$ and $(G', \mathcal{F}')$ are isomorphic, the sheaf Laplacians $L^0_{\mathcal{F}}$ and $L^1_{\mathcal{F}}$ have the same spectrum at every level, and $\mathrm{ShePHerD}(G, \mathcal{F}) = \mathrm{ShePHerD}(G', \mathcal{F}')$. The descriptor does not depend on the orientations used to write the coboundary.*

Theorem 3.6 says that a connection assignment that is isomorphism compatible in the sense of Definition 3.1, composed with an attribute-induced filtration, yields a graph-isomorphism invariant descriptor. Propositions 3.2 and 3.3 establish that condition for the two assignments we use, so the attributed experiments and the isomorphism benchmarks of Section 4 report one descriptor evaluated under two assignments rather than two different methods.

ShePHerD Expressiveness. For a connected graph, global sections are precisely the vectors fixed by cycle holonomy, $H^0(G; \mathcal{F}) \cong \mathrm{Fix}(\mathrm{Hol}_v(\mathcal{F}))$ (Lemma B.2), which leads to a strict separation illustrated in Figure 2 and proved in Theorem 3.7.

Theorem 3.7. *There exist non-isomorphic attributed graphs G_1, G_2 that are indistinguishable by RePHINE, SpectRe, and edge-aware 1-WL under any attribute-induced filtration. The ShePHerD descriptor separates them with a non-trivial transport by capturing $\dim H^0(G_1, \mathcal{F}) = 0 \neq 1 = \dim H^0(G_2, \mathcal{F})$.*

Theorem 3.7 states that the class of ShePHerD descriptors with orthogonal connections is strictly more expressive than RePHINE and SpectRe. Notably, this is a property for an arbitrary connection not necessarily one of our proposed connections Equations (2) and (3).

While the separation in Theorem 3.7 relies on attributes to construct a non-trivial filtration, SheP-HerD's expressivity fundamentally stems from the learned connection itself. Consequently, on graphs with no attributes at all, where the filtration carries nothing and the baselines collapse completely, the sheaf connection still provides separation power. This is the regime of the spectral assignment. As an

extreme test of ShePHerD separation capability, we consider the Paulus graphs (Paulus, 1973), the fifteen nonisomorphic strongly regular graphs with parameters $(25, 12, 5, 6)$.

Theorem 3.8. *Let $\mathcal{P}$ be the fifteen Paulus graphs with constant vertex and edge attributes, giving $\binom{15}{2} = 105$ pairs. Every filtration computed from these attributes by a 1-WL bounded function is constant on the vertices and on the edges of every $G \in \mathcal{P}$; consequently RePHINE, SpectRe, the graph Laplacian spectrum and folklore 2-WL return the same value on all fifteen.*

Consequently, Theorem 3.8 shows that RePHINE and SpectRe return the same descriptor on all 105 pairs, as do the graph Laplacian spectrum and folklore 2-WL. In contrast, our empirical tests expose that ShePHerD with spectral connection separates all fifteen Paulus graphs as numerically demonstrated at Table 5.

3.4 SHEPHERD++

We now show how ShePHerD can be combined with a sheaf diffusion model. Our sheaf diffusion is based on the NSD model (see Section 2.1). However, we learn the connection R formed by transport maps $R_{u\to v}$ instead of the restriction maps $\mathcal{F}_{u\trianglelefteq e}$, for $e = (u, v)$. Importantly, the connection is learned *once* on G and shared by the persistent sheaf cohomology and sheaf diffusion components of an edge-aware topological sheaf neural network, which we call ShePHerD++.

Let $h = (h_v)_{v\in V}$ denote the node embedding matrix returned by a graph encoder, and $R = \{R_{u\to v}\}$ the shared orthogonal connection defining $\mathcal{F}$. We set

$$z_{\text{gnn}} = \Psi(h), \qquad z_{\text{diff}} = \Psi\left(\mathbf{X}^{(T)}\right), \qquad z_{\text{pers}} = \text{DeepSets}(\text{ShePHerD}(G, \mathcal{F})),$$

where Ψ is a permutation invariant function and $\mathbf{X}^{(T)}$ is obtained after T iterations of Eq. (1) using the normalized sheaf Laplacian induced by R and h as initial node features.

The final graph-level prediction is given by

$$h_G = \left[z_{\text{gnn}} \,\middle\|\, z_{\text{diff}} \,\middle\|\, z_{\text{pers}}\right], \qquad \hat{y} = \text{MLP}(h_G).$$

Notably, we also learn ShePHerD's filtration function end-to-end using the diffusion operator induced by R — used to compute $\mathbf{X}^{(T)}$ — coupled with a scalar read-out layer.

4 EXPERIMENTS

To assess the effectiveness of ShePHerD from an empirical perspective, we consider two sets of experiments. The first explores the combination of topological descriptors and GNNs in real-world tasks. The second consists of isomorphism tests on synthetic benchmarks designed to evaluate the expressive power of graph models. Implementation details are provided in Appendix D.

The two sets of experiments build the sheaf in the two ways of Section 3.1: the attributed tasks use the feature-conditioned connection, which reads node and edge attributes, and the isomorphism benchmarks carry no attributes, so they use the spectral connection, which reads only the graph. The descriptor applied after the assignment is the same in both, and we report it under one name.

We compare eight models. PersLay (Carrière et al., 2020) and TOGL (Horn et al., 2022) are baselines that we rerun in our pipeline, since their published numbers use different splits. GPS Rampášek et al. (2022) and GIN(E) Hu et al. (2020); Xu et al. (2019) do not have the topological component, and sheaf diffusion uses the learned connection without persistence. RePHINE and SpectRe are obtained in the same setup. The last two rows are ShePHerD and ShePHerD++ (that adds sheaf diffusion together), which share a single learned connection. NSD (Bodnar et al., 2022) and CSNN (Ribeiro et al., 2026) are sheaf neural networks that we compare and learn the restriction maps in a different way for diffusion. Within each backbone, all rows use the same data split, encoders, training protocol, and filtration networks, so the comparison isolates the contribution of learning the coefficient system.

4.1 GRAPH-LEVEL PREDICTION

Setup. We use five TU datasets, the MOLHIV molecular benchmark, and the 12k subset of ZINC (Dwivedi et al., 2023). We report classification accuracy on the TU datasets, ROC-AUC on MOLHIV,

Table 2: Graph-level prediction performance: classification accuracy (↑, %), MOLHIV ROC-AUC (↑), and ZINC test MAE (↓). GIN(E) and GPS are the two backbones used without any topological component. Results are mean ± standard deviation over paired seeds, five on the TU datasets and MOLHIV and three on ZINC. PersLay and TOGL are rerun under the same protocol. Bold indicates the best mean in each column and underline the second best.

Method	MUTAG ↑	PTC MR ↑	IMDB-B ↑	NCI1 ↑	NCI109 ↑	MOLHIV ↑	ZINC ↓
PersLay	85.3 ± 4.4	53.7 ± 6.2	67.6 ± 2.1	71.2 ± 1.1	70.8 ± 2.7	70.8 ± 1.4	$1.35 \pm .01$
TOGL	77.9 ± 13.6	53.7 ± 4.7	75.6 ± 4.0	79.7 ± 1.1	75.8 ± 1.7	73.5 ± 2.5	$0.470 \pm .010$
GIN(E)	75.8 ± 10.9	57.1 ± 6.1	76.2 ± 1.3	78.6 ± 1.5	76.6 ± 1.1	75.0 ± 2.9	$0.395 \pm .007$
GPS	76.8 ± 13.7	61.1 ± 5.9	72.4 ± 2.6	74.0 ± 0.6	74.2 ± 0.7	75.0 ± 1.9	$0.246 \pm .014$
Sheaf diffusion	73.7 ± 11.2	60.0 ± 3.5	75.2 ± 1.9	79.7 ± 0.6	$\underline{78.4 \pm 1.8}$	75.7 ± 1.8	$0.226 \pm .027$
NSD	62.1 ± 2.3	57.1 ± 2.0	52.0 ± 2.8	70.2 ± 1.0	69.0 ± 1.7	75.1 ± 2.1	$0.530 \pm .009$
CSNN	82.1 ± 2.9	56.0 ± 1.6	71.4 ± 2.9	79.9 ± 2.2	78.2 ± 3.6	74.7 ± 2.8	$0.310 \pm .026$
RePHINE	91.6 ± 4.7	58.3 ± 4.3	75.0 ± 2.7	$\underline{80.7 \pm 1.2}$	78.2 ± 0.5	76.4 ± 1.8	$0.213 \pm .019$
SpectRe	91.6 ± 6.0	$\underline{63.4 \pm 7.7}$	77.0 ± 2.6	79.8 ± 1.0	77.4 ± 1.2	$\underline{76.5 \pm 2.5}$	$0.227 \pm .022$
ShePHerD	$\mathbf{93.7 \pm 2.4}$	$\mathbf{63.5 \pm 6.7}$	$\mathbf{77.6 \pm 2.1}$	80.2 ± 1.4	78.3 ± 1.1	$\mathbf{77.4 \pm 2.5}$	$\underline{0.198 \pm .010}$
ShePHerD++	$\underline{92.6 \pm 2.9}$	61.7 ± 7.5	$\underline{77.2 \pm 2.9}$	$\mathbf{82.2 \pm 0.8}$	$\mathbf{80.4 \pm 1.8}$	75.8 ± 1.2	$\mathbf{0.177 \pm .010}$

Table 3: Effect of the GNN backbone. Each method is evaluated with GIN(E) and with GPS using RWSE-16 and Laplacian-8 positional encodings; every cell is the best validation split observed configuration of that method on that backbone. TU results and MOLHIV use five paired seeds and ZINC uses three. Bold indicates the stronger backbone for each method and dataset.

Method	Backbone	MUTAG ↑	PTC MR ↑	IMDB-B ↑	NCI1 ↑	NCI109 ↑	ZINC ↓	MOLHIV ↑
Sheaf diffusion	GIN(E)	$\mathbf{73.7 \pm 11.2}$	50.9 ± 5.5	$\mathbf{75.2 \pm 1.9}$	$\mathbf{79.7 \pm 0.6}$	$\mathbf{78.4 \pm 1.8}$	$0.344 \pm .002$	$\mathbf{75.7 \pm 1.8}$
	GPS	70.5 ± 12.7	$\mathbf{60.0 \pm 3.5}$	71.8 ± 3.4	74.6 ± 2.3	72.4 ± 1.4	$\mathbf{0.226 \pm .027}$	73.2 ± 3.5
RePHINE	GIN(E)	$\mathbf{91.6 \pm 4.7}$	$\mathbf{58.3 \pm 4.3}$	$\mathbf{75.0 \pm 2.7}$	$\mathbf{80.7 \pm 1.2}$	$\mathbf{78.2 \pm 0.5}$	$0.324 \pm .005$	75.6 ± 1.1
	GPS	74.7 ± 4.4	56.0 ± 6.3	72.0 ± 4.7	72.7 ± 3.1	74.0 ± 1.7	$\mathbf{0.213 \pm .019}$	$\mathbf{76.4 \pm 1.8}$
SpectRe	GIN(E)	90.5 ± 4.4	52.6 ± 4.3	$\mathbf{77.0 \pm 2.6}$	$\mathbf{79.8 \pm 1.0}$	$\mathbf{77.4 \pm 1.2}$	$0.316 \pm .030$	$\mathbf{76.5 \pm 2.5}$
	GPS	$\mathbf{91.6 \pm 6.0}$	$\mathbf{63.4 \pm 7.7}$	71.0 ± 3.7	73.6 ± 1.1	72.7 ± 3.6	$\mathbf{0.227 \pm .022}$	74.2 ± 2.7
ShePHerD	GIN(E)	$\mathbf{93.7 \pm 2.4}$	57.1 ± 4.5	$\mathbf{77.6 \pm 2.1}$	$\mathbf{80.2 \pm 1.4}$	$\mathbf{78.3 \pm 1.1}$	$0.306 \pm .007$	$\mathbf{77.4 \pm 2.5}$
	GPS	89.5 ± 5.3	$\mathbf{63.5 \pm 6.7}$	72.4 ± 4.5	73.9 ± 2.0	74.1 ± 2.6	$\mathbf{0.198 \pm .010}$	73.9 ± 1.5
ShePHerD++	GIN(E)	$\mathbf{92.6 \pm 2.9}$	$\mathbf{61.7 \pm 7.5}$	$\mathbf{77.2 \pm 2.9}$	$\mathbf{82.2 \pm 0.8}$	$\mathbf{80.4 \pm 1.8}$	$0.295 \pm .001$	$\mathbf{75.8 \pm 1.2}$
	GPS	89.5 ± 5.3	58.3 ± 6.3	73.8 ± 2.9	72.5 ± 1.6	72.4 ± 2.2	$\mathbf{0.177 \pm .010}$	75.7 ± 1.1

and mean absolute error on ZINC, as mean and standard deviation over paired seeds, five on the TU datasets and MOLHIV and three on ZINC.

Results. Table 2 shows the results on real-world data. A model with a learned coefficient system attains the highest mean in all seven cases: ShePHerD on MUTAG, PTC_MR, IMDB-B and MOLHIV, and ShePHerD with diffusion on NCI1, NCI109 and ZINC. The same two rows take both top positions on MUTAG, IMDB-B and ZINC. In the remaining four columns the runner-up is a baseline, SpectRe on PTC_MR and MOLHIV, RePHINE on NCI1 and sheaf diffusion on NCI109; the first three are the same pipeline with the connection fixed, so the difference between those rows and ShePHerD is the coefficient system alone. The margins are uneven and we report them as they are: 2.1 points over the best baseline on MUTAG, 0.9 on MOLHIV and 1.5 on NCI1, against 0.1 on PTC_MR and 0.6 on IMDB-B, which is well inside the seed deviation on those two small datasets. ShePHerD improves on PersLay and TOGL on every dataset, and on both plain backbones. Overall, these results support the idea that GNNs can benefit from a learned coefficient system on real-world graph learning tasks, and that persistence and diffusion are complementary: ShePHerD is the best model on MUTAG, PTC_MR, IMDB-B and MOLHIV, and ShePHerD++ on NCI1, NCI109 and ZINC.

Backbone impact. For each method, the backbone of Table 3 with the better mean validation score on each dataset is selected, and its test score is reported; PersLay, TOGL, NSD and CSNN use the GIN(E) backbone. The backbone preference is a property of the dataset rather than of the method: GIN(E) is stronger for every method on IMDB-B, NCI1 and NCI109, and GPS is stronger for every method on ZINC, while MUTAG, PTC_MR and MOLHIV split by method. The learned coefficient system leads on most of the backbones, so its gain is not an effect of the backbone choice. ZINC shows this most plainly, where the progression from diffusion through the two constant-sheaf descriptors to ShePHerD and its combination with diffusion is monotone on GIN(E), $0.344 \to 0.324 \to 0.316 \to 0.306 \to 0.295$, and reaches 0.177 on GPS.

Table 4: BREC pairs distinguished by the complete descriptors. All methods use the BREC/SpectRe filtration and no canonicalization, and ShePHerD uses the spectral connection. The three baseline rows reproduce the values reported by Ji et al. (2025).

Method	Basic	Regular	Extension	CFI	4-vertex	Distance-reg.	Strongly reg.	Total
PH	2	0	7	3	0	0	0	12/400
RePHINE	59	47	55	3	0	0	0	164/400
SpectRe	60	50	100	5	0	1	0	216/400
ShePHerD	60	50	100	11	20	20	50	**311/400**

4.2 BREC AND SR25

BREC. Following Wang and Zhang (2024), we consider the four hundred pairs of the BREC benchmark, grouped into seven categories. We compare four descriptors obtained with fixed filtration functions, namely node degrees and augmented Forman–Ricci curvatures, $f_V(v) = \deg(v)$ and $f_E(uv) = 4 - \deg(u) - \deg(v) + 3|N(u) \cap N(v)|$, which are the functions released with BREC and SpectRe. No canonical form is computed for any descriptor. ShePHerD uses the spectral connection, which places one rotation on each edge. A pair counts as distinguished when both graphs return the same descriptor under a random relabeling, so every entry is a lower bound.

Results. Table 4 presents the number of pairs distinguished by each descriptor. ShePHerD separates 311 pairs against 216 for SpectRe, 164 for RePHINE and 12 for ordinary persistence. The gain falls in the categories where the constant-sheaf descriptors separate almost none: all twenty four-vertex and fifty strongly regular pairs, where they separate none, and all twenty distance-regular pairs, where SpectRe separates one. This is the behaviour Theorem 3.8 predicts, since on a regular family the filtration carries no information and only the coefficient system can separate. On CFI, where every descriptor separates few pairs, ShePHerD separates eleven against five for SpectRe; one graph there does not reproduce its descriptor under relabeling, and its pair is counted as not distinguished.

Table 5: Pairwise separation on the fifteen nonisomorphic $SR(25, 12, 5, 6)$ graphs. The constant, degree-sum and BREC/SpectRe filtrations collapse to a single edge level on this family; ShePHerD uses the spectral connection.

Method	Pairs
Edge-aware 1-WL	$0/105$
Folklore 2-WL	$0/105$
Laplacian spectrum	$0/105$
RePHINE	$0/105$
SpectRe	$0/105$
ShePHerD	$\mathbf{105/105}$

SR25. The Paulus graphs additionally show how persistent sheaf cohomology can improve distinguishability. Since all fifteen graphs share the parameters $(25, 12, 5, 6)$, the three filtrations considered in Theorem 3.8 — namely, constant, degree-sum, and BREC/SpectRe — collapse into a single trivial filtration across the entire family.

Results. Table 5 shows that Edge-aware 1-WL, folklore 2-WL, the Laplacian spectrum, RePHINE and SpectRe separate no pair of the family, whereas the spectral connection separates all 105. Together with Table 4, we infer that the gain of a learned coefficient system is largest exactly where the filtration is least informative.

5 CONCLUSION

We introduced ShePHerD, a topological descriptor that learns the coefficient system of graph persistence using cellular sheaves. While cellular sheaves equip graphs with vector spaces and maps that describe how information is transported between them, ShePHerD brings this additional structure to persistence, making the geometry encoded by these maps part of the persistent representation. Notably, by combining persistent sheaf cohomology with the spectrum of the sheaf Laplacian, our descriptor recovers SpectRe for the trivial sheaf and RePHINE, when the spectral information is removed. Our theoretical and empirical results show that learned coefficient systems can capture graph structure that persistence with fixed coefficients does not see.

Limitations and future work. ShePHerD is developed here for graphs, with sheaf structure on vertices and edges. Cellular sheaves, however, are naturally defined on higher-dimensional cell complexes. Extending ShePHerD to such domains would allow the coefficient system to encode relations involving higher-order cells and would make it possible to study persistent sheaf cohomology beyond the graph setting. This is the main direction left open by the present work.

AI USE STATEMENT

We used generative AI tools to draft and polish parts of the paper. We also used AI to assist in coding and experiments. All statements and proofs in this paper were written and double-checked by the authors — AI was used for polishing and checking. It also helped us find the Paulus family of graphs. Although we already knew the properties of the graphs we were interested in, an AI-guided search helped us identify that these graphs are a well-documented graph-family in the literature.

REPRODUCIBILITY STATEMENT

The protocol is fixed and shared by every row: the splits, the seeds, the optimizer and the budget are given in Appendix D, together with the dataset statistics in Table 6 and the complete hyperparameters in Table 7. No per-dataset tuning is performed, so every comparison is at the same width, depth and training budget. RePHINE and SpectRe are not quoted but obtained in our own pipeline, and PersLay and TOGL are rerun under the same protocol, so all rows of Table 2 are directly comparable. Each run writes a record carrying its dataset, method, seed, best validation score, test score at that epoch and parameter count. The isomorphism experiments involve no training: the eigenspace rule, the angle and the two rounding levels used to decide a separation are stated in Appendix D.

REFERENCES

Rubén Ballester and Bastian Rieck. On the expressivity of persistent homology in graph learning. In *Proceedings of the Third Learning on Graphs Conference*, volume 269 of *Proceedings of Machine Learning Research*. PMLR, 2025.

Jacob Bamberger, Federico Barbero, Xiaowen Dong, and Michael Bronstein. Bundle neural networks for message diffusion on graphs. In *International Conference on Learning Representations (ICLR)*, 2025.

Federico Barbero, Cristian Bodnar, Haitz Sáez de Ocáriz Borde, Michael Bronstein, Petar Veličković, and Pietro Liò. Sheaf neural networks with connection Laplacians. In *ICML Workshop on Topology, Algebra, and Geometry in Machine Learning (TAG-ML)*, 2022.

Silvia Beddar-Wiesing, Giuseppe Alessio D'Inverno, Caterina Graziani, Veronica Lachi, Alice Moallemy-Oureh, Franco Scarselli, and Josephine Maria Thomas. Weisfeiler–Lehman goes dynamic: An analysis of the expressive power of graph neural networks for attributed and dynamic graphs. *Neural Networks*, 2024.

Cristian Bodnar, Francesco Di Giovanni, Benjamin P. Chamberlain, Pietro Liò, and Michael Bronstein. Neural sheaf diffusion: A topological perspective on heterophily and oversmoothing in GNNs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Jean-Daniel Boissonnat and Clément Maria. Computing persistent homology with various coefficient fields in a single pass. *Journal of Applied and Computational Topology*, 2019.

Jin-Yi Cai, Martin Fürer, and Neil Immerman. An optimal lower bound on the number of variables for graph identification. *Combinatorica*, 1992.

Mathieu Carrière, Frédéric Chazal, Yuichi Ike, Théo Lacombe, Martin Royer, and Yuhei Umeda. PersLay: A neural network layer for persistence diagrams and new graph topological signatures. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.

Justin Curry. Sheaves, cosheaves and applications. *arXiv preprint arXiv:1303.3255*, 2014.

Vijay Prakash Dwivedi, Chaitanya K. Joshi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. *Journal of Machine Learning Research*, 2023.

Edelsbrunner, Letscher, and Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4), 2002.

Stefano Fiorini, Hakan Aktas, Iulia Duta, Pietro Morerio, Alessio Del Bue, Pietro Lio, and Stefano Coniglio. Sheaves reloaded: A direction awakening. In *International Conference on Learning Representations (ICLR)*, 2026.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *International Conference on Machine Learning (ICML)*, 2017.

Jakob Hansen and Robert Ghrist. Toward a spectral theory of cellular sheaves. *Journal of Applied and Computational Topology*, 2019.

Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

C. Hofer, F. Graf, B. Rieck, M. Niethammer, and R. Kwitt. Graph filtration learning. In *International Conference on Machine Learning (ICML)*, 2020.

Max Horn, Edward De Brouwer, Michael Moor, Yves Moreau, Bastian Rieck, and Karsten Borgwardt. Topological graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2022.

Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HJlWWJSFDH.

Johanna Immonen, Amauri H. Souza, and Vikas Garg. Going beyond persistent homology using persistent homology. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Mattie Ji, Amauri H. Souza, and Vikas Garg. Graph persistence goes spectral. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Mattie Ji, Indradyumna Roy, and Vikas Garg. Contraction and hourglass persistence for learning on graphs, simplices, and cells. In *International Conference on Learning Representations (ICLR)*, 2026.

Mario Lezcano-Casado and David Martínez-Rubio. Cheap orthogonal constraints in neural networks: A simple parametrization of the orthogonal and unitary group. In *International Conference on Machine Learning (ICML)*, 2019.

Haggai Maron, Heli Ben-Hamu, Hadar Serviansky, and Yaron Lipman. Provably powerful graph networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: higher-order graph neural networks. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

A. J. L. Paulus. Conference matrices and graphs of order 26. Technical report, Technische Hogeschool Eindhoven, 1973.

Ladislav Rampášek, Michael Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

André Ribeiro, Ana Luiza Tenório, Juan Belieni, Amauri H. Souza, and Diego Mesquita. Cooperative sheaf neural networks. In *International Conference on Learning Representations (ICLR)*, 2026.

Bastian Rieck, Christian Bock, and Karsten Borgwardt. A persistent Weisfeiler–Lehman procedure for graph classification. In *International Conference on Machine Learning (ICML)*, 2019.

Florian Russold. Persistent sheaf cohomology. *arXiv preprint arXiv:2204.13446*, 2022.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter W. Battaglia. Learning to simulate complex physics with graph networks. In *International Conference on Machine Learning (ICML)*, 2020.

Yanbo Wang and Muhan Zhang. An empirical study of realized GNN expressiveness. In *International Conference on Machine Learning (ICML)*, 2024.

Xiaoqi Wei and Guo-Wei Wei. Persistent sheaf Laplacians. *Foundations of Data Science*, 2025.

B. Weisfeiler and A. A. Lehman. A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsia*, 2(9):12–16, 1968.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019.

Karthik Yegnesh. Persistence and sheaves. *arXiv preprint arXiv:1612.03522*, 2016.

A Simplicial Homology and Sheaf Cohomology

This section states the difference between simplicial homology and cellular sheaf cohomology, and recalls the results that connect them for a specific sheaf and a class of spaces.

We state the definitions in the abelian group setting.

Definition A.1. A *chain complex* is a sequence of abelian groups $\{C_i\}$ equipped with homomorphisms $\partial_i : C_i \to C_{i-1}$ such that $\partial_i \circ \partial_{i+1} = 0$. The maps ∂_i are called *boundary maps*.

The condition $\partial_i \circ \partial_{i+1} = 0$ implies $\mathrm{Im}(\partial_{i+1}) \subseteq \ker(\partial_i)$, and the i-th homology group is $H_i(C_\bullet) = \ker(\partial_i)/\,\mathrm{Im}(\partial_{i+1})$.

Definition A.2. A *cochain complex* is a sequence $\{C^i\}$ equipped with homomorphisms $\delta^i : C^i \to C^{i+1}$ satisfying $\delta^{i+1} \circ \delta^i = 0$. Its i-th cohomology group is $H^i(C^\bullet) = \ker(\delta^i)/\,\mathrm{Im}(\delta^{i-1})$.

Simplicial homology and cellular sheaf cohomology are particular instances of these constructions. A graph itself is a 1-dimensional simplicial complex; higher-dimensional simplices arise only after choosing an additional simplicial completion, which we do not use here. We recall the general simplicial definitions following Immonen et al. (2023).

Definition A.3. A *simplicial complex* K is a set of simplices such that every face of a simplex in K is also in K, and the intersection of any two simplices in K is either empty or a common face.

Definition A.4. Let K be a simplicial complex. The *i-chains* are formal sums $c = \sum a_p \sigma_p$, where $a_p \in \mathbb{Z}/2\mathbb{Z}$ and σ_p are i-simplices in K.

Note here we work over $\mathbb{Z}/2\mathbb{Z}$, as in the ordinary persistence construction used by RePHINE. Over other coefficient fields or rings one chooses orientations and includes the corresponding incidence signs in the boundary operator. Together with componentwise addition, i-chains form the group $C_i(K)$.

For a simplex $\sigma = (v_0, \ldots, v_i) \in K$, the *boundary* is $\partial_i \sigma = \sum_{j=0}^{i} (v_0, \ldots, v_{j-1}, v_{j+1}, \ldots, v_i)$, i.e., a sum of the $(i-1)$-dimensional faces of σ. Extending linearly gives the *boundary homomorphism* $\partial_i : C_i(K) \to C_{i-1}(K)$, and since $\partial_i \circ \partial_{i+1} = 0$ the groups $C_i(K)$ form a chain complex. The *i-th homology group* with coefficients in $\mathbb{Z}/2\mathbb{Z}$ is $H_i(K, \mathbb{Z}/2\mathbb{Z}) = \ker \partial_i / \,\mathrm{Im}\, \partial_{i+1}$, whose rank is the i-th *Betti number* β_i.

To construct the simplicial cohomology, we take the dual space $C^i(X; \mathbb{Z}/2\mathbb{Z}) = \mathrm{Hom}(C_i(X; \mathbb{Z}/2\mathbb{Z}), \mathbb{Z}/2\mathbb{Z})$, and the coboundary maps are the transpose of the boundary maps. The Universal Coefficient Theorem (Hatcher, 2002) gives the following.

Proposition A.5 (Universal coefficients over a field)**.** *Let K be a finite simplicial complex and F a field. Then $H^i(K; F) \cong \mathrm{Hom}_F(H_i(K; F), F)$ naturally. In particular, $\dim_F H^i(K; F) = \dim_F H_i(K; F)$.*

To connect ordinary cohomology with cellular sheaf cohomology, we recall the latter construction and then identify the rank-one constant-sheaf case, following Curry (2014).

Definition A.6. Let X be a regular cell complex. A *cellular sheaf* $\mathcal{F}$ of vector spaces on X consists of a vector space $\mathcal{F}(\sigma)$ for each cell $\sigma \in X$, and a linear restriction map $\mathcal{F}(\sigma \leq \tau) : \mathcal{F}(\sigma) \to \mathcal{F}(\tau)$ for each pair $\sigma \leq \tau$, satisfying $\mathcal{F}(\sigma \leq \sigma) = \mathrm{id}$ and $\mathcal{F}(\tau \leq \rho) \circ \mathcal{F}(\sigma \leq \tau) = \mathcal{F}(\sigma \leq \rho)$. For graphs, the absence of higher-order cells satisfies these conditions automatically.

The *i-th cochain group* is $C^i(X;\mathcal{F}) = \bigoplus_{\dim\sigma=i}\mathcal{F}(\sigma)$, and the *coboundary operator* $\delta^i : C^i(X;\mathcal{F}) \to C^{i+1}(X;\mathcal{F})$ is

$$(\delta^i c)(\tau) = \sum_{\substack{\sigma\le\tau \\ \dim\sigma=i}} [\tau:\sigma]\,\mathcal{F}(\sigma\le\tau)\big(c(\sigma)\big),$$

where $[\tau:\sigma]$ is the incidence number. Since $\delta^{i+1}\circ\delta^i = 0$, this is a cochain complex, and the *i-th cellular sheaf cohomology group* is $H^i(X;\mathcal{F}) = \ker\delta^i/\operatorname{Im}\delta^{i-1}$.

Theorem A.7 (Constant sheaf cohomology is cellular cohomology)**.** *Let X be a regular cell complex and $\underline{\mathbb{R}}$ the constant cellular sheaf on X. Then for all $i \ge 0$ there is a natural isomorphism $H^i(X;\underline{\mathbb{R}}) \cong H^i(X;\mathbb{R})$.*

Proof. Both cochain groups are $\bigoplus_{\dim\sigma=i}\mathbb{R}$, so $\Phi^i(c)(\sigma) = c(\sigma)$ is a vector space isomorphism. For the coboundaries, the constant sheaf has identity restriction maps, so $(\delta^i_{\text{sheaf}}c)(\tau) = \sum_{\sigma\le\tau}[\tau:\sigma]\,c(\sigma)$, while δ^i_{simp} is the transpose of ∂_{i+1}, giving $(\delta^i_{\text{simp}}\Phi^i(c))(\tau) = \sum_{\sigma\le\tau}[\tau:\sigma]\,c(\sigma)$. The two agree, so Φ is an isomorphism of cochain complexes and induces an isomorphism on cohomology. □

Proposition A.8 (Graph Betti numbers are field-independent)**.** *Let $G = (V,E)$ be a graph with c connected components and let F be any field. Then $\dim_F H_0(G;F) = c$ and $\dim_F H_1(G;F) = |E| - |V| + c$. In particular these dimensions do not depend on the field.*

Proof. Since G has no cells of dimension greater than 1, its chain complex over F is $0 \to F^{|E|} \xrightarrow{\partial_1} F^{|V|} \to 0$. The incidence matrix ∂_1 has rank $|V| - c$ over any field, since on each connected component the edges of a spanning tree give rank one less than the number of vertices. Therefore $\dim_F H_0(G;F) = |V| - \operatorname{rank}\partial_1 = c$ and $\dim_F H_1(G;F) = |E| - \operatorname{rank}\partial_1 = |E| - |V| + c$. □

Thus, for graphs, rank-one constant-sheaf cohomology over $\mathbb{R}$ and ordinary homology over $\mathbb{Z}/2\mathbb{Z}$ have the same dimensions in degrees 0 and 1. We do not identify these vector spaces across different coefficient fields. Instead, the persistence-level equality required for Proposition 3.5 is established directly in the next section.

B Sheaf Persistent Pairs

In this section we recall persistent homology on graph filtrations, describe the RePHINE/SpectRe descriptors that ShePHerD extends, and then formalize the corresponding persistent sheaf-cohomology construction.

B.1 Graph persistence and the RePHINE/SpectRe baseline

Definition B.1. A *filtration* of a simplicial complex K is a nested sequence of subcomplexes $K_1 \subseteq K_2 \subseteq \cdots \subseteq K_n = K$.

For graphs, the filtration gives an increasing sequence of subgraphs, one subset of vertices and edges at each step. In practice, the filtration is a learned parameterized function f, and throughout it, persistent homology tracks how topological features evolve. In the edge filtration considered here, all vertices are present from the initial level, so the 0-dimensional classes are born at the beginning of the filtration. As the filtration progresses and two separate connected components merge, a selection rule dictates which representative is assigned the *death* of the corresponding 0-dimensional class, while the other persists to represent the combined component. Correspondingly, 1-dimensional features are created at their respective filtration thresholds, but unlike 0-dimensional classes they never die in a graph. The complete lifespan of a topological class is encoded as a tuple (b,d), called a *persistent pair*; in RePHINE these classes are associated with representative vertices in degree 0 and cycle-creating edges in degree 1. Here b is the birth and d the death, with $d = \infty$ for a feature that never dies.

Once we have the sequence of subgraphs $G_0 = (V,\emptyset) \subseteq G_1 \subseteq \cdots \subseteq G_m = G$, to each graph G_s we associate a homology group $H_i(G_s)$, $i = 0, 1$, whose rank is the Betti number β_i. Hence

the sequence $H_i(G_0) \to H_i(G_1) \to \cdots \to H_i(G_m)$ encodes the birth and death of i-dimensional homology classes.

RePHINE (Immonen et al., 2023) decorates these persistence pairs with the filtration-derived coordinates (b, d, α, γ). SpectRe (Ji et al., 2025) further associates to each such tuple the nonzero spectrum of the graph Laplacian of the corresponding connected component. Thus its degree-0 tuples have the form $(b, d, \alpha, \gamma, \mathrm{spec}_{>0} L_G|_C)$, while a cycle-creating edge contributes $(1, d(e), 0, 0, \mathrm{spec}_{>0} L_G|_C)$.

B.2 Persistent sheaf cohomology

In ShePHerD, we instead learn a single cellular sheaf $\mathcal{F}$ on the terminal graph G and restrict it to every filtration level, $\mathcal{F}_s = \mathcal{F}|_{G_s}$. For $s \le t$, the inclusion $i_{s,t} : G_s \hookrightarrow G_t$ induces a restriction map $i^*_{s,t} : H^q(G_t; \mathcal{F}_t) \to H^q(G_s; \mathcal{F}_s)$. Hence the filtration determines the copersistence module

$$H^q(G_0; \mathcal{F}_0) \longleftarrow H^q(G_1; \mathcal{F}_1) \longleftarrow \cdots \longleftarrow H^q(G_m; \mathcal{F}_m),$$

whose interval decomposition gives the persistent sheaf-cohomology classes, with the indexing reversed to report the usual increasing-filtration birth–death convention. This is the persistent cellular-sheaf construction introduced by Yegnesh (2016) and developed by Russold (2022); our contribution is to learn the coefficient system jointly with the graph representation.

Relative to SpectRe, ShePHerD therefore replaces ordinary graph homology by this persistent sheaf cohomology and replaces the graph Laplacian by the corresponding sheaf Laplacians $L^0_{\mathcal{F}} = \delta^\top \delta$ and $L^1_{\mathcal{F}} = \delta \delta^\top$.

Persistence events. For a rank-d sheaf, an entering edge contributes a block of d rows to the filtered coboundary matrix. Suppose that this block increases $\mathrm{rank}\,\delta$ by r, with $0 \le r \le d$. Since for a graph $H^0 = \ker \delta$ and $H^1 = C^1 / \mathrm{im}\,\delta$, the dimension of H^0 decreases by r, while the dimension of H^1 increases by $d - r$:

$$\dim H^0(G_s; \mathcal{F}_s) = \dim H^0(G_{s-1}; \mathcal{F}_{s-1}) - r,$$
$$\dim H^1(G_s; \mathcal{F}_s) = \dim H^1(G_{s-1}; \mathcal{F}_{s-1}) + (d - r).$$

Therefore, when $0 < r < d$, a single entering edge may simultaneously kill degree-0 classes and create degree-1 classes.

RePHINE decorations. We extend RePHINE's (Immonen et al., 2023) selection rule to these multiplicities as follows. At an edge merging two connected components, the r degree-0 death tuples inherit the (α, γ) decorations of the losing unassociated representative, which is then marked as associated for that event, following RePHINE's selection rule. If the endpoints already belong to the same connected component, any degree-0 deaths use the component representative without consuming it. Each of the $d - r$ degree-1 births receives the entering edge coordinates $(1, d(e), 0, 0)$.

B.3 Recovery of RePHINE and SpectRe

Proposition 3.5. If $\mathcal{F}$ is the trivial sheaf, then $\mathrm{ShePHerD}(G, \mathcal{F})$ equals SpectRe on the same filtration; dropping the spectral coordinate recovers RePHINE.

Proof. If $\mathcal{F} = \underline{\mathbb{R}}$ is the trivial sheaf, every restriction map is the identity, so the sheaf coboundary δ is the ordinary signed incidence matrix of the graph.

When an entering edge joins two previously disconnected components, $\mathrm{rank}\,\delta$ increases by one and one degree-0 class dies. When its endpoints already lie in the same component, the new incidence row is linearly dependent, so the rank does not increase and one degree-1 class is born. Since a graph has no 2-cells, such degree-1 classes never subsequently die. These events depend only on whether the entering edge merges components or closes a cycle, and are therefore identical over $\mathbb{R}$ and $\mathbb{Z}/2\mathbb{Z}$. Hence the RePHINE persistence pairs and decorations agree.

For the spectral labels, $L^0_{\underline{\mathbb{R}}} = \delta^\top \delta = L_G$, so the degree-0 label is exactly SpectRe's nonzero graph-Laplacian spectrum. Moreover, $\delta^\top \delta$ and $\delta \delta^\top$ have the same nonzero eigenvalues with multiplicity, so the degree-1 spectral label agrees as well. Therefore $\mathrm{ShePHerD}(G, \underline{\mathbb{R}}) = \mathrm{SpectRe}(G)$, and removing the spectral coordinate gives RePHINE. □

B.4 Holonomy and global sections

For a collection of linear maps $\mathcal{H}$ acting on $\mathcal{F}(v)$, we denote by $\mathrm{Fix}(\mathcal{H}) = \{x \in \mathcal{F}(v) : Hx = x \text{ for every } H \in \mathcal{H}\}$ their common fixed subspace.

Lemma B.2 (Global sections are the vectors fixed by holonomy)**.** *Let G be connected and $\mathcal{F}$ an orthogonal connection sheaf. For any vertex v, $H^0(G;\mathcal{F}) \cong \mathrm{Fix}\big(\mathrm{Hol}_v(\mathcal{F})\big)$.*

Proof. Let $x = (x_u)_{u\in V} \in H^0(G;\mathcal{F})$. Since $\delta x = 0$, the values at adjacent vertices agree under transport, so along any path $\gamma_{v\to u}$ we have $x_u = P^{\gamma}_{v\to u} x_v$. In particular, if γ is closed at v then $P^{\gamma}_{v\to v} x_v = x_v$, so x_v is fixed by every element of $\mathrm{Hol}_v(\mathcal{F})$, and evaluation at v defines a linear map $H^0(G;\mathcal{F}) \to \mathrm{Fix}(\mathrm{Hol}_v(\mathcal{F}))$.

Conversely, let $x_v \in \mathrm{Fix}(\mathrm{Hol}_v(\mathcal{F}))$. Since G is connected, for every vertex u choose a path $\gamma_{v\to u}$ and define $x_u = P^{\gamma}_{v\to u} x_v$. This is independent of the path: given two paths $\gamma_{v\to u}$ and $\widetilde{\gamma}_{v\to u}$, following γ and then the reverse of $\widetilde{\gamma}$ is closed at v, so $(P^{\widetilde{\gamma}}_{v\to u})^{-1} P^{\gamma}_{v\to u} x_v = x_v$, and multiplying by $P^{\widetilde{\gamma}}_{v\to u}$ gives $P^{\gamma}_{v\to u} x_v = P^{\widetilde{\gamma}}_{v\to u} x_v$. Finally, for an oriented edge $e = (u, w)$, choosing the path to w through u gives $x_w = R_{u\to w} x_u$, so $\delta x = 0$ and $x \in H^0(G;\mathcal{F})$. Evaluation at v and extension by parallel transport are inverse linear maps, which proves the claim. □

B.5 Compatibility of the two connection assignments

Proposition 3.2. *The feature-conditioned connection assignment is compatible with isomorphisms.*

Proof. Let $\pi : G \to H$ be an attribute-preserving isomorphism. Permutation equivariance of the encoder gives $h^H_{\pi(v)} = h^G_v$ for every vertex, and π preserves edge attributes, so $z^H_{\pi(u)\pi(v)} = z^G_{uv}$. Since ϕ_θ is one shared map applied to the concatenation of these quantities,

$$a^H_{\pi(u)\to\pi(v)} = \phi_\theta([h^G_u \| h^G_v \| z^G_{uv}]) - \phi_\theta([h^G_v \| h^G_u \| z^G_{uv}]) = a^G_{u\to v},$$

hence A and therefore $R^H_{\pi(u)\to\pi(v)} = R^G_{u\to v}$ on every directed edge. Taking $\widehat{\pi} = \pi$ in Definition 3.1 shows the assignment is compatible with isomorphisms. □

Proposition 3.3. *The spectral connection assignment is compatible with isomorphisms.*

Proof. Let $\pi : G \to H$ be an isomorphism with permutation matrix P, so that $L_H = P L_G P^\top$. The selection rule reads only eigenvalues and their multiplicities, and P preserves both, so it selects corresponding eigenspaces, $E_H = P E_G$, and the two graphs receive the same d. If U_G and U_H are orthonormal bases of E_G and E_H, then PU_G is another orthonormal basis of E_H, hence $U_H = PU_G Q$ for some $Q \in O(d)$, and

$$p^H_{\pi(v)} = U_H^\top e_{\pi(v)} = Q^\top U_G^\top P^\top e_{\pi(v)} = Q^\top U_G^\top e_v = Q^\top p^G_v.$$

Norms and inner products are unchanged by $Q^\top$, so corresponding edges have the same denominator and the same pairs are linearly independent. Hence

$$B^H_{\pi(u)\to\pi(v)} = \frac{Q^\top \big(p^G_v (p^G_u)^\top - p^G_u (p^G_v)^\top\big) Q}{\|p^G_u \wedge p^G_v\|} = Q^\top B^G_{u\to v} Q,$$

and conjugation commutes with the exponential, so $R^H_{\pi(u)\to\pi(v)} = Q^\top R^G_{u\to v} Q$. Taking $\widehat{\pi} = \pi$ and $Q_v = Q^\top$ for every v in Definition 3.1 shows the assignment is compatible with isomorphisms. The same Q occurs at every vertex, so the two connections differ by one change of orthonormal basis in the stalks. No vertex or edge attribute enters the construction. □

B.6 Isomorphism invariance

Theorem 3.6. *Let $\rho : G \to G'$ be an isomorphism of attributed graphs, so that ρ preserves vertex and edge attributes, and let the filtration be induced by the attributes, $f'_V \circ \rho = f_V$ and $f'_E \circ \rho = f_E$. If the connection is given by an compatible with isomorphisms assignment, then the filtered cochain*

complexes of $(G, \mathcal{F})$ and $(G', \mathcal{F}')$ are isomorphic, the sheaf Laplacians $L^0_{\mathcal{F}}$ and $L^1_{\mathcal{F}}$ have the same spectrum at every level, and $\mathrm{ShePHerD}(G, \mathcal{F}) = \mathrm{ShePHerD}(G', \mathcal{F}')$. *The descriptor does not depend on the orientations used to write the coboundary.*

Proof. Let $Q_v \in O(d)$ be the matrices supplied by Definition 3.1, so that $R'_{\rho(u)\to\rho(v)} = Q_{\rho(v)} R_{u\to v} Q_{\rho(u)}^\top$ on every edge. Let $P : C^0(G; \mathcal{F}) \to C^0(G'; \mathcal{F}')$ send the block of v to the block of $\rho(v)$ and act on it by $Q_{\rho(v)}$, and define Q on C^1 in the same way, acting on the block of $e = (u, v)$ by $Q_{\rho(v)}$. Both are orthogonal, being block permutations composed with block-diagonal orthogonal matrices. Fix an edge $e = (u, v)$. Its block row in δ carries $+I_d$ in the block of v and $-R_{u\to v}$ in the block of u, so

$$(\delta' P x)_{\rho(e)} = Q_{\rho(v)} x_v - Q_{\rho(v)} R_{u\to v} Q_{\rho(u)}^\top Q_{\rho(u)} x_u = Q_{\rho(v)} (\delta x)_e.$$

Collecting the rows,

$$\delta' = Q\, \delta\, P^\top.$$

This is an isomorphism of filtered cochain complexes, compatible with the restriction maps, so the two copersistence modules are isomorphic and their interval decompositions agree. We nevertheless record the rank argument below, because the descriptor carries the decorations α and γ and the component spectra in addition to the intervals. Because $f'_V \circ \rho = f_V$ and $f'_E \circ \rho = f_E$, the two filtrations have the same values and the same order, so the same relation holds level by level, $\delta'_s = Q_s \delta_s P_s^\top$. Orthogonal maps preserve rank, hence $\operatorname{rank} \delta'_s = \operatorname{rank} \delta_s$ for every s, and therefore the rank increment r of each entering edge agrees. By the event count of the previous section, the number of degree-0 deaths and of degree-1 births at each level agrees, so the persistence pairs agree. The decorations α and γ agree because they are computed from the same filtration values.

For the spectral coordinates,

$$L'^0_{\mathcal{F}} = \delta'^\top \delta' = P\, \delta^\top Q^\top Q\, \delta\, P^\top = P\, L^0_{\mathcal{F}}\, P^\top, \qquad L'^1_{\mathcal{F}} = Q\, L^1_{\mathcal{F}}\, Q^\top,$$

so both are conjugate to their counterparts by an orthogonal matrix and have the same spectrum. Since ρ maps connected components onto connected components, the spectra attached to corresponding tuples agree. Hence the two multisets of tuples are equal and $\mathrm{ShePHerD}(G, \mathcal{F}) = \mathrm{ShePHerD}(G', \mathcal{F}')$.

Under the spectral assignment the matrices Q_v are all equal to one $Q \in O(d)$ by Proposition 3.3, and no attribute is read at any point, so the same argument gives the equality on graphs with no attributes.

Orientation. Let $e = (u, v)$ be reoriented as (v, u), so that its transport becomes $R_{v\to u} = R_{u\to v}^\top$. Writing B_e for the block row of e and $B_{e'}$ for that of the reoriented edge,

$$-R_{u\to v}^\top B_e = \big[\ \cdots\ -R_{u\to v}^\top \text{ at } v\ \cdots\ R_{u\to v}^\top R_{u\to v} = I_d \text{ at } u\ \cdots\big] = B_{e'}.$$

Thus the two block rows span the same subspace, so every rank increment is unchanged. Collecting these relations gives $\delta' = M\delta$ with M block diagonal and orthogonal, so $L^0_{\mathcal{F}}$ is unchanged and $L^1_{\mathcal{F}}$ has the same spectrum. The descriptor is therefore independent of the orientations chosen to write the coboundary. □

B.7 The Paulus graphs

These are the fifteen strongly regular graphs with parameters $(25, 12, 5, 6)$, enumerated by Paulus (1973) in the classification of the strongly regular graphs on 25 and 26 vertices. The parameters are those of a conference graph, $k = (n-1)/2$, $\lambda = (n-5)/4$ and $\mu = (n-1)/4$ at $n = 25$: every vertex has twelve neighbours, two adjacent vertices have exactly five common neighbours, and two nonadjacent vertices exactly six. One member is the Paley graph of $\mathrm{GF}(25)$, in which $u \sim v$ when $u - v$ is a nonzero square; the remaining fourteen are not of that form.

What the family shares is everything a classical invariant reads. The parameters force the adjacency spectrum $\{12^1, 2^{12}, (-3)^{12}\}$, hence the graph Laplacian spectrum $\{0^1, 10^{12}, 15^{12}\}$, identical on all fifteen, and they force 1-WL and folklore 2-WL to the same stable colouring, which is the content of Theorem 3.8. Every filtration built from those colourings is therefore constant, and the filtration has a single edge level.

What differs is the arrangement of the edges, which the parameters do not determine. Counting the 4-cliques gives the seven values $73, 75, 79, 83, 87, 89, 90$ across the family, and the automorphism

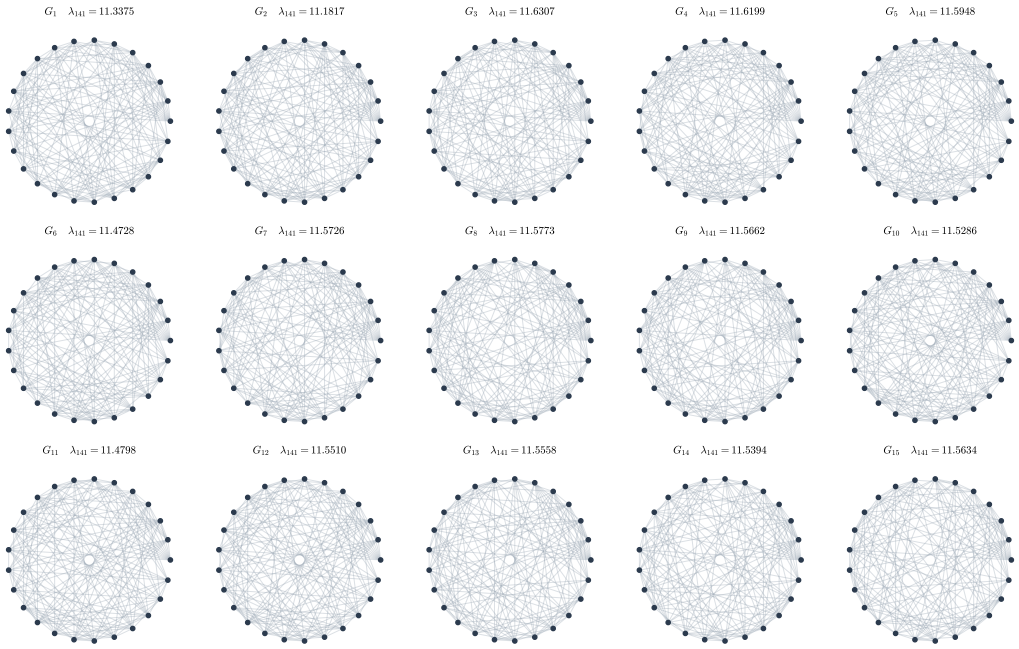


Figure 3: The fifteen Paulus graphs $SR(25, 12, 5, 6)$. They share the same degree sequence, adjacency spectrum and 1-WL colouring, so every filtration of Theorem 3.8 is constant on each of them, and the graph Laplacian $D - A$ has a single spectrum for the whole family, determined by $(25, 12, 5, 6)$; this is the spectrum SpectRe attaches to its tuples, which is why it separates no pair. With the trivial sheaf the sheaf Laplacian is $L_G \otimes I_d$ and inherits that spectrum. Under the spectral connection, instead, $d = 12$ and the fifteen sheaf-Laplacian spectra are pairwise distinct; the panels report the 141st smallest eigenvalue, which alone already takes fifteen distinct values.

groups have the six orders $1, 2, 3, 6, 72, 600$; the order 600 is the Paley graph and two of the fifteen have no nontrivial automorphism. Neither quantity takes fifteen distinct values, so neither separates the family on its own, and neither is visible to a filtration.

Theorem 3.8. *Let $\mathcal{P}$ be the fifteen Paulus graphs with constant vertex and edge attributes, giving $\binom{15}{2} = 105$ pairs. Every filtration computed from these attributes by a 1-WL bounded function is constant on the vertices and on the edges of every $G \in \mathcal{P}$, and consequently RePHINE, SpectRe, the graph Laplacian spectrum and folklore 2-WL return the same value on all fifteen graphs.*

Proof. Collapse of the baselines. Let $G \in \mathcal{P}$, so G is strongly regular with parameters $(25, 12, 5, 6)$ and all attributes are constant. Every vertex has the same initial color, and every vertex has exactly 12 neighbors. If all vertices share a color at round k, then every vertex receives the same multiset of neighbor colors at round $k + 1$, hence the same new color. By induction, 1-WL assigns one color to all vertices at every round, and the same holds for the edge-aware refinement because all edge attributes are equal. Consequently any filtration computed by a 1-WL bounded function is constant on V and constant on E, so the filtration has a single edge level: $G_0 = (V, \emptyset)$ followed by G itself.

At that single level the ordinary persistence pairs are determined by $|V|$, $|E|$ and the number of connected components, all of which are fixed by the parameters $(25, 12, 5, 6)$, so RePHINE returns the same tuples for every $G \in \mathcal{P}$. For SpectRe, the adjacency matrix of a strongly regular graph satisfies $A^2 = kI + \lambda A + \mu(J - I - A)$, so its eigenvalues, and hence the spectrum of $L_G = kI - A$, are determined by the parameters alone; the spectral decoration is therefore also the same. Folklore 2-WL gives the same stable coloring on all graphs of $\mathcal{P}$, by the argument recorded with its definition in Section B.8: on a strongly regular family the number of common neighbors of a pair is determined by λ and μ, so the color of a pair depends only on whether it is adjacent.

Separation. The parameters $(25, 12, 5, 6)$ give L_G the nonzero eigenvalues 10 and 15, each of multiplicity 12, so the selection rule takes the eigenspace of eigenvalue 10 and $d = 12$ on every $G \in \mathcal{P}$. A constant transport placed on every edge without reference to an orientation must satisfy $S = R_{u\to v} = R_{v\to u} = R_{u\to v}^{\top}$, hence $S = S^{\top}$, and then $L_S = 12\, I_{25} \otimes I_{12} - A \otimes S$, whose spectrum is determined by the spectrum of A and that of S and is therefore the same on the whole family. The spectral connection instead places a different rotation on each edge, determined by the plane that its two endpoints span in E_G, so L is not of that form and the argument fixing its spectrum from the parameters does not apply. Whether it separates a given pair is settled computationally: under the protocol of Appendix D it separates all 105, and every graph reproduces its descriptor under relabeling.

□

B.8 Edge-aware 1-WL and strict expressivity

For graphs with node attributes x_v and edge attributes e_{uv}, we use the edge-aware 1-WL refinement

$$c^{(0)}(v) = x_v, \qquad c^{(k+1)}(v) = \mathrm{HASH}\Big(c^{(k)}(v), \{\{(e_{uv}, c^{(k)}(u)) : u \in N(v)\}\}\Big),$$

where HASH is injective on its arguments and $\{\{\cdot\}\}$ denotes a multiset. Two attributed graphs are distinguished by edge-aware 1-WL if, at some iteration, the multisets of their vertex colors differ. Thus, unlike ordinary 1-WL, the refinement records both the current colors of neighboring vertices and the attributes of the incident edges. This attributed version of 1-WL and its relation to the expressive power of GNNs with node and edge attributes are studied by Beddar-Wiesing et al. (2024).

The folklore 2-WL test (Cai et al., 1992) colors pairs of vertices rather than single vertices; we follow the presentation of Maron et al. (2019). Writing x_v for the attribute of v and $\mathbf{1}[uv \in E]$ for the adjacency indicator, it starts from

$$c^{(0)}(u, v) = \big(x_u,\, x_v,\, \mathbf{1}[uv \in E]\big)$$

and refines by

$$c^{(k+1)}(u, v) = \mathrm{HASH}\Big(c^{(k)}(u, v), \{\{\, \big(c^{(k)}(u, w),\, c^{(k)}(w, v)\big) : w \in V \,\}\}\Big).$$

Two graphs are distinguished when the multisets of pair colors differ at some iteration. The test is strictly stronger than 1-WL, and it is the standard reference point above the message-passing bound. On a strongly regular family it gives no information: the number of common neighbors of a pair depends only on whether the pair is adjacent, through the parameters λ and μ, so every pair of a given adjacency type receives the same color at every iteration, and all graphs with equal parameters receive the same stable coloring. This is the fact used in the proof of Theorem 3.8.

Theorem 3.7. *There exist non-isomorphic attributed graphs G_1, G_2 that are indistinguishable by RePHINE, SpectRe, and edge-aware 1-WL under any attribute-induced filtration. The ShePHerD descriptor separates them with a non-trivial transport by capturing* $\dim H^0(G_1, \mathcal{F}) = 0 \neq 1 = \dim H^0(G_2, \mathcal{F})$.

Proof. Label the top vertex v_0 and the remaining vertices $v_1, \ldots, v_7$ clockwise around the blue 8-cycle. With this labeling, the red edges in Figure 2 form the matchings $M_1 = \{\{v_0, v_3\}, \{v_1, v_5\}, \{v_2, v_6\}, \{v_4, v_7\}\}$ for G_1, and $M_2 = \{\{v_0, v_4\}, \{v_1, v_5\}, \{v_2, v_6\}, \{v_3, v_7\}\}$ for G_2.

The attributed graphs are not isomorphic: any attribute-preserving isomorphism maps blue edges to blue edges and therefore restricts to an automorphism of the blue 8-cycle, which preserves cyclic distance. The cyclic distances between the endpoints of the red edges are $\{3, 3, 4, 4\}$ for G_1 and $\{4, 4, 4, 4\}$ for G_2.

Edge-aware 1-WL does not distinguish them. All vertices have the same initial attribute, and every vertex is incident to exactly two blue edges and one red edge. If every vertex has the same color at iteration k, then every vertex receives the same multiset at iteration $k + 1$, hence the same new color. By induction all vertices of both graphs keep a common color.

RePHINE and SpectRe agree for every attribute-induced filtration. All node attributes are equal, so f_V assigns one value α to every vertex, and there are only two edge attributes, so let $\lambda_B = f_E(\text{blue})$ and $\lambda_R = f_E(\text{red})$. If $\lambda_B < \lambda_R$, the only nontrivial intermediate subgraph is the blue 8-cycle, the same in both graphs; if $\lambda_R < \lambda_B$, it is the red perfect matching, isomorphic in both; if $\lambda_B = \lambda_R$, all edges enter together. At the final level the underlying graphs are isomorphic. Moreover every vertex is incident to the same multiset of edge attributes, so $\gamma(v) = \min\{\lambda_B, \lambda_R\}$ for every vertex of both graphs, and the decorations agree. Since the filtered subgraphs are isomorphic at each level, their graph Laplacians are isospectral and SpectRe returns the same descriptor.

An orthogonal connection separates the pair. Take the rank-one connection $R_e = -1$ on blue edges and $R_e = +1$ on red edges. In G_1 the red edge $\{v_0, v_3\}$ together with the blue path v_0, v_1, v_2, v_3 forms a cycle with three blue edges, whose holonomy is $(+1)(-1)^3 = -1$. Every vector fixed by the holonomy group satisfies $x = -x$, so $x = 0$ and Lemma B.2 gives $\dim H^0(G_1; \mathcal{F}) = 0$. For G_2, choose $a \in \mathbb{R}$ and set $x_{v_i} = (-1)^i a$. Along every blue edge $x_{v_{i+1}} = -x_{v_i}$, matching $R_e = -1$; every red edge of G_2 joins opposite vertices v_i and v_{i+4}, and $x_{v_{i+4}} = (-1)^{i+4} a = (-1)^i a = x_{v_i}$, matching $R_e = +1$. Each a gives a global section, and since G_2 is connected a global section is determined by its value at one vertex, so $\dim H^0(G_2; \mathcal{F}) = 1$.

Thus ShePHerD distinguishes G_1 and G_2 while RePHINE, SpectRe and edge-aware 1-WL do not. Together with Proposition 3.5, which recovers SpectRe and RePHINE in the constant case, this proves the strict expressivity claim. □

Remark B.3. Theorem 3.7 concerns the class of ShePHerD descriptors with orthogonal connections. The rank-one witness above uses $-1 \in O(1) \setminus SO(1)$. The same parity separation is realized in $SO(2)$ by $R_{\text{blue}} = -I_2$ and $R_{\text{red}} = I_2$: the odd-blue cycle in G_1 has holonomy $-I_2$ and no nonzero fixed vector, while in G_2 every $a \in \mathbb{R}^2$ extends by $x_{v_i} = (-1)^i a$, so $\dim H^0(G_1; \mathcal{F}) = 0$ and $\dim H^0(G_2; \mathcal{F}) = 2$. The experiments use the $SO(d)$ parameterization of Section 3.1; the theorem is a statement about the descriptor class, not about that specific learner.

C Orthogonal groups

Recall that $O(d) = \{R \in \mathbb{R}^{d\times d} : R^\top R = I\}$, $SO(d) = \{R \in O(d) : \det R = 1\}$, and $\mathfrak{so}(d) = \{A \in \mathbb{R}^{d\times d} : A^\top = -A\}$.

Proposition C.1 ($O(d)$ is not path-connected)**.** *For every $d \geq 1$, the orthogonal group $O(d)$ is not path-connected.*

Proof. The determinant is continuous on $O(d)$ and takes only the values ± 1. We have $\det(I) = 1$ and $\det(P) = -1$ for $P = \text{diag}(-1, 1, \ldots, 1) \in O(d)$. A continuous path $\Gamma : [0, 1] \to O(d)$ from I to P would make $\det \circ \Gamma$ a continuous function from $[0, 1]$ onto a set containing 1 and -1, so by the Intermediate Value Theorem it would attain 0, which is impossible for an orthogonal matrix. Hence no such path exists. □

Proposition C.2 ($SO(d)$ is path-connected)**.** *For every $d \geq 1$, $SO(d)$ is path-connected, and the exponential map $\exp : \mathfrak{so}(d) \to SO(d)$ is surjective.*

Proof. It is standard that the Lie algebra of $SO(d)$ is the space of skew-symmetric matrices and that exp is surjective onto $SO(d)$ (Lezcano-Casado and Martínez-Rubio, 2019). Given $R_0 = \exp(A_0)$ and $R_1 = \exp(A_1)$ with $A_0, A_1 \in \mathfrak{so}(d)$, set $A(t) = A_0 + t(A_1 - A_0)$. Since $\mathfrak{so}(d)$ is a vector space, $A(t)^\top = -A(t)$ for every t, so $\Gamma(t) = \exp(A(t))$ is a continuous path in $SO(d)$ from R_0 to R_1. □

D Experimental Details

D.1 Datasets and training protocol

We use five TU datasets (MUTAG, PTC_MR, IMDB-BINARY, NCI1, NCI109), the MOLHIV molecular property prediction dataset, and the 12k subset of the ZINC graph-regression benchmark (Dwivedi et al., 2023). The TU datasets have no official split, so we use the partition of the RePHINE

and SpectRe codebase unchanged: one stratified $80/10/10$ split with a fixed random state, giving $150/19/19$ on MUTAG, $275/34/35$ on PTC_MR, $800/100/100$ on IMDB-B, $3288/411/411$ on NCI1 and $3301/413/413$ on NCI109. ZINC and MOLHIV use their official splits. Since the split is fixed, the reported deviation is over the five training seeds $0, \dots, 4$, and every method sees the same split and the same seeds, so differences are taken per seed.

Accuracy is reported on the TU datasets, ROC-AUC on MOLHIV, and mean absolute error on ZINC. MOLHIV is scored over the whole evaluation split with the official evaluator, because a per-batch ROC-AUC is undefined on batches without a positive label.

All configurations are trained with Adam using learning rate 10^{-3}, weight decay 10^{-5}, and batch size 64. The learning rate is multiplied by 0.5 after 10 epochs without validation improvement, down to 10^{-6}. Training runs for at most 60 epochs and stops early after 15 epochs without validation improvement. For each run we retain the checkpoint with the best validation score and report its test score. No per-dataset tuning is performed: one setting is shared by all methods on all datasets, so every comparison is at the same width, depth, optimizer and budget; parameter counts still differ across methods and are reported per configuration.

Table 6: Summary statistics of the datasets. Node and edge counts are averages over the whole dataset. The TU splits are the stratified $80/10/10$ partition of the RePHINE and SpectRe codebase; ZINC and MOLHIV use their public splits.

Dataset	Graphs	Nodes	Edges	Task	Metric
MUTAG	188	17.9	19.8	binary classification	accuracy
PTC_MR	344	14.3	14.7	binary classification	accuracy
IMDB-BINARY	1 000	19.8	96.5	binary classification	accuracy
NCI1	4 110	29.9	32.3	binary classification	accuracy
NCI109	4 127	29.7	32.1	binary classification	accuracy
ZINC (12k)	12 000	23.2	24.9	regression	MAE
ogbg-molhiv	41 127	25.5	27.5	binary classification	ROC-AUC

Table 7: Hyperparameters. One setting is shared by every method on every dataset; no per-dataset tuning is performed. Values marked GPS apply only to the GPS backbone, and values marked ShePHerD only to the rows with a learned connection. The transport MLP ϕ reads the two endpoint features and the edge feature of width e and returns the $\binom{d}{2} = 3$ coordinates of a skew-symmetric matrix; the eigenvalue MLP ψ_{spec} is applied to each nonzero eigenvalue separately before the tuples are pooled.

Optimization		**Architecture**	
optimizer	Adam	hidden width	64
learning rate	10^{-3}	backbone depth	2
weight decay	10^{-5}	local layer	GINE
schedule	halve on plateau	attention heads	4 (GPS)
plateau patience	10	structural encoding	RWSE 16 (GPS)
minimum rate	10^{-6}	positional encoding	LapPE 8 (GPS)
maximum epochs	60	stalk dimension d	3 (ShePHerD)
early stopping	15	transport MLP ϕ	$(2 \cdot 64 + e) \to 32 \to 3$
batch size	64	filtration channels	1
seeds	$0, \dots, 4$	filtration MLP f_V, f_E	$64 \to 16 \to 1$, sigmoid
selection	best validation	eigenvalue MLP ψ_{spec}	$1 \to 4 \to 4$
loss	cross-entropy, L^1 on ZINC	readout width	64
		diffusion layers	4
		diffusion channels	16

D.2 SHEPHERD ARCHITECTURE

Encoders and backbone. Categorical vertex and edge attributes are embedded in $\mathbb{R}^{64}$; IMDB-BINARY has no vertex attributes, so a degree one-hot encoding is used instead. Both backbones have

two layers and produce vertex representations $h \in \mathbb{R}^{|V|\times 64}$. For GINE, each layer is a GINEConv operator parameterized by a two-layer MLP; on NCI1 and NCI109, which carry no edge attributes, this reduces to GIN. For GPS, each layer combines the same local operator with four-head global self-attention (Rampášek et al., 2022), together with a 16-dimensional random-walk structural encoding and an 8-dimensional Laplacian positional encoding, both linearly projected and added to the vertex representation.

Learned sheaf connection. The connection is parameterized as in Section 3.1. The shared two-layer MLP ϕ has hidden width 32 and receives $[h_u \| h_v \| e_{uv}]$, where e_{uv} is the embedded edge attribute. We use stalk dimension $d = 3$, so ϕ produces $\binom{3}{2} = 3$ coordinates, and

$$a_e = \phi([h_u \| h_v \| e_{uv}]) - \phi([h_v \| h_u \| e_{uv}]), \qquad R_e = \exp\big(\mathrm{Skew}(a_e)\big) \in SO(3).$$

One connection $R = \{R_e\}_{e\in E}$ is learned per graph. When both ShePHerD persistence and sheaf diffusion are used, they share this same connection. At $d = 2$ the group $SO(2)$ is abelian, so every holonomy commutator vanishes, though nontrivial holonomy is still possible there (Remark B.3); we take $d = 3$.

Sheaf diffusion. The backbone representations are lifted to stalk-valued features and propagated through four sheaf-diffusion layers with 16 channels, each using the normalized sheaf Laplacian built from R (Bodnar et al., 2022).

ShePHerD persistence. We use one filtration channel. The filtration networks f_V and f_E are two-layer MLPs with dimensions $64 \to 16 \to 1$ and sigmoid output, and

$$\alpha(v) = f_V(h_v), \qquad d(e) = f_E(h_u + h_v), \qquad \gamma(v) = \min_{e \ni v} d(e).$$

The symmetric input $h_u + h_v$ makes the edge-filtration value independent of the orientation chosen for e. Degree-0 tuples have persistence coordinates $(0, d, \alpha, \gamma)$ and a degree-1 event at an entering edge e has coordinates $(1, d(e), 0, 0)$. An essential degree-0 class has $d = \infty$; for the neural encoding only, its death coordinate is represented by 1, the upper endpoint of the sigmoid range.

Differentiable sheaf-Laplacian spectra. The persistence pairing is a discrete operation: the filtration order and the rank increments of the filtered coboundary determine which degree-0 classes die and which degree-1 classes are born. Once the events are determined, their coordinates (d, α, γ) are gathered from the filtration networks and remain differentiable. The spectral decorations are computed directly from the learned connection: for each event we assemble the sheaf Laplacian of its connected component C from the transports R_e and take $\mathrm{spec}_{>0}(L^0_{\mathcal{F}}|_C)$ and $\mathrm{spec}_{>0}(L^1_{\mathcal{F}}|_C)$. These matrices are assembled inside the computational graph and diagonalized with a differentiable symmetric eigensolver, so the decorations are differentiable functions of R_e. In particular, when the model uses persistence without diffusion, the discrete events give no gradient to ϕ, whereas the eigenvalues do; the connection is therefore not detached when computing the event spectra.

Eigendecompositions are performed in double precision, and numerical zero eigenvalues are removed before the readout, consistently with $\mathrm{spec}_{>0}$ in Definition 3.4. For stability at degenerate eigenspaces the decomposition is retried with a small structured diagonal perturbation when necessary. For each tuple the nonzero eigenvalues are encoded independently by a shared MLP $\psi_{\mathrm{spec}} : \mathbb{R} \to \mathbb{R}^4$ with dimensions $1 \to 4 \to 4$, and the encoded eigenvalues are concatenated with the four persistence coordinates. Degree-0 and degree-1 tuples are processed by separate DeepSets readouts, each applying an element-wise MLP, mean aggregation, and a linear map to $\mathbb{R}^{64}$; the degree-wise representations are added and passed through batch normalization and a two-layer MLP.

Graph-level prediction. Writing z_{pers} for the ShePHerD representation and z_{diff} for the sheaf-diffusion representation, the graph representation is $h_G = [\mathrm{pool}(h) \| z_{\mathrm{diff}} \| z_{\mathrm{pers}}]$, with absent components omitted, and a two-layer MLP maps h_G to the prediction. The backbone readout is always present, so each persistence row is a strict extension of the plain backbone row.

Number of parameters. On MUTAG with the GIN(E) backbone the models contain 26,375 parameters for the plain backbone, 33,723 for sheaf diffusion, 60,912 for RePHINE, 64,958 for

SpectRe, 74,164 for ShePHerD and 81,016 for ShePHerD with diffusion. The remaining datasets differ only in the input encoder, so the counts stay in the same range. Exact parameter matching is not imposed, but no row is given extra width or depth.

Hardware. The real-world experiments use one NVIDIA V100 GPU with 16 GB of memory per run, except for the MOLHIV runs, which use one A100. The isomorphism tests involve no training and run on CPU.

D.3 Configurations in Table 2

All rows use the same data split, encoders, backbone architecture, optimization protocol, seeds, and training budget, and the persistence-based configurations also share the same filtration architecture. The rows differ in the topological information supplied to the predictor:

- *plain backbone*: no topological component is used;
- *RePHINE*: the persistence component is evaluated in the rank-one constant-sheaf case, $d = 1$ and $R_e = 1$, and the spectral decoration is removed;
- *SpectRe*: the same constant-sheaf case, retaining the ordinary graph-Laplacian spectral decoration;
- *sheaf diffusion*: sheaf diffusion with a learned $SO(3)$ connection, without persistent sheaf cohomology;
- *ShePHerD*: ShePHerD persistence with a learned $SO(3)$ connection and differentiable sheaf-Laplacian spectral decorations, without sheaf diffusion;
- *ShePHerD++*: both components, sharing the same learned $SO(3)$ connection.

The RePHINE and SpectRe rows are in-pipeline implementations of their corresponding persistence descriptors; Proposition 3.5 establishes their descriptor-level recovery in the rank-one constant-sheaf case. Table 2 should therefore be read as a controlled comparison of the persistence descriptors, the learned coefficient system, and sheaf diffusion under a shared experimental setup.

D.4 Structural experiments

The BREC and SR25 results use the spectral connection of Section 3.1 and involve no training. No canonical form is computed for any descriptor. The eigenspace of L_G of smallest multiplicity at least three is taken, ties broken by eigenvalue, and further complete eigenspaces are added in the same order while some edge has $\|p_u \wedge p_v\|/(\|p_u\|\,\|p_v\|) \le 10^{-6}$; the angle is $\pi/2$ for every graph. The selected dimension is $d = 12$ on SR25 and ranges from 3 to 30 over the BREC categories, with $d = 27$ on the four-vertex family. The filtration is the one released with BREC and SpectRe, $f_V(v) = \deg(v)$ and $f_E(uv) = 4 - \deg(u) - \deg(v) + 3|N(u) \cap N(v)|$. Vertices and edges are ordered by an edge-colored 1-WL refinement together with the row multisets of the projector onto E_G; both are constant on a strongly regular graph, so they separate no pair of that family and serve only to fix the order in which tied edges enter the reduction. A pair counts as distinguished when the complete decorated multisets differ after rounding to three decimals and also after rounding to two, so that a separation cannot rest on the last retained digit, and when both of its graphs return the same descriptor under a random relabeling; a pair failing that check is counted as not distinguished.