

Graph Neural Networks: Architectures, Fundamental Properties and Applications

Navid NaderiAlizadeh, Alejandro Ribeiro, Luana Ruiz, Zhiyang Wang

Web: gnn.seas.upenn.edu/aaai-2025/

Feb 26 2025

Manifold Neural Networks and Generalization Analysis

Zhiyang Wang

Dept. of Electrical and Systems Engineering
University of Pennsylvania

zhiyangw@seas.upenn.edu
<http://zhiyangwang.net>

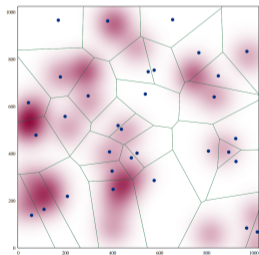
TH15: Graph Neural Networks: Architectures, Fundamental Properties and Applications
<https://gnn.seas.upenn.edu/aaai-2025/>

Feb 26 2025

- | **Continuous limit models** help to understand discrete GNNs and implement them in large-scale
- | **Graphon NNs** show power over dense and random graphs – maintain geometric information?
 -) E.g., wireless communication systems, robotic systems, point clouds for 3D models



Cellular network



Collaborative robots



Point clouds

- | We study **continuous limits of graph NNs** as the size of graph grows to infinity) **manifold NNs**

| Continuous limit model brings insights into sampled discrete models) graphs and images



100 nodes



800 nodes



143 95

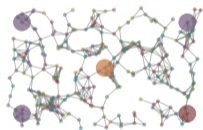


600 399

| Continuous models easier for theoretical insights , Discrete models easier for practical application

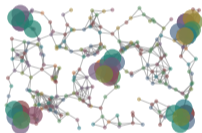
| **Graph convolutional filters** are linear combinations of polynomials on matrix representations of graphs

$$y = \sum_{k=0}^{\mathcal{K}-1} h_k S^k x$$



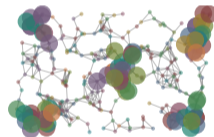
$h_0 S^0 x$

+



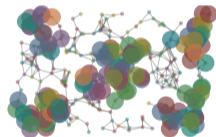
$h_1 S^1 x$

+



$h_2 S^2 x$

+

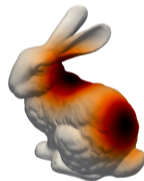
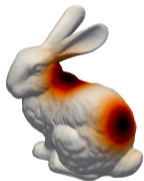


$h_3 S^3 x$

| **Graph neural networks (GNNs)** compose layers of **graph filters** and **point-wise non-linearities**

- Manifold convolutional filters are linear combinations of Laplace-Beltrami operator exponentials

$$g(x) = \int_0^T \tilde{h}(t) e^{tL} f(x) dt$$

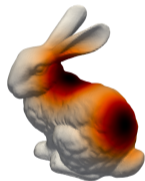
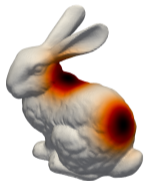


$$\tilde{h}(0T_s) e^{0T_s L} f + \tilde{h}(1T_s) e^{1T_s L} f + \tilde{h}(2T_s) e^{2T_s L} f + \tilde{h}(3T_s) e^{3T_s L} f$$

- Manifold neural networks (MNNs) compose layers of manifold filters and point-wise non-linearities

- Manifold convolutional filters are linear combinations of Laplace-Beltrami operator exponentials

$$g(x) = \sum_{k=0}^{\infty} \tilde{h}(kT_s) e^{kT_s L} f(x)$$



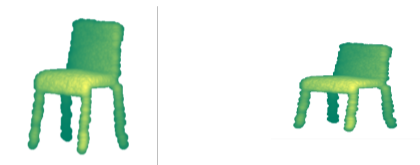
$$\tilde{h}(0T_s) e^{0T_s L} f + \tilde{h}(1T_s) e^{1T_s L} f + \tilde{h}(2T_s) e^{2T_s L} f + \tilde{h}(3T_s) e^{3T_s L} f$$

- Manifold neural networks (MNNs) compose layers of manifold filters and point-wise non-linearities

- | Graph convolutions are algebraically equivalent to standard convolutions on images



Sample from high res to low res

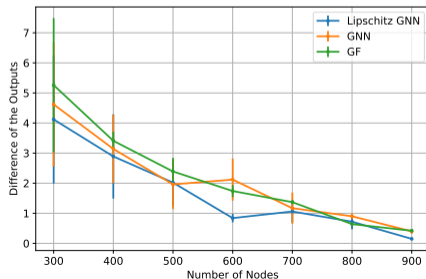
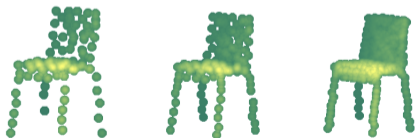
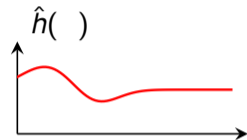


Deform from high res

- | GNNs have fundamental properties derived from MNNs to understand their performances
 -) Transferability: Convergence of GNNs to MNNs implies transferability of GNNs across scales
 -) Stability: Stability of MNNs to manifold deformations reveals stability of GNNs
 -) Generalization: GNNs can generalize to unseen data with robustness on a manifold

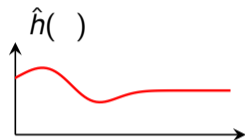
GNNs converge to the underlying MNNs provided the filters satisfy spectral continuity conditions

$$(H; L_n; P_n f) P_n (H; L; f) = O \left(\frac{N}{n} + A_h \rho - \frac{\log(n)}{n} k f K_{L^2(M)} \right)$$



- GNNs converge to the underlying MNNs provided the filters satisfy spectral continuity conditions

$$(H; L_n; P_n f) P_n (H; L; f) = O \left(\frac{N}{n} + A_h \rho + \frac{\log(n)}{n} \text{Kf} K_{L^2(M)} \right)$$

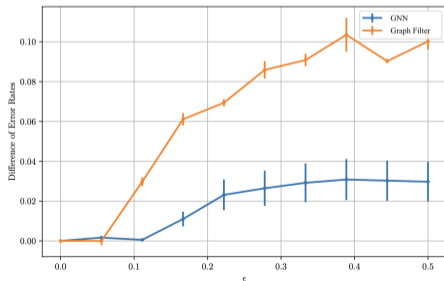
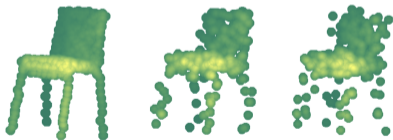
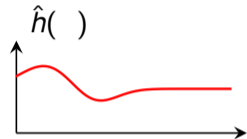


GNNs trained on small graphs with continuous filters are able to transfer to large graphs

Z. Wang et al, *Geometric Graph Filters and Neural Networks: Limit Properties and Discriminability Trade-offs*, IEEE Trans on Signal Processing

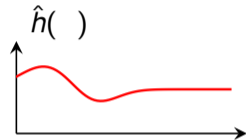
- | **Stability** of **MNNs** to manifold deformations implies **stability** of **GNNs** with **continuous filters**

$$(\mathbf{H}; L; f) - (\mathbf{H}; L^0; f) = O\left(\frac{N}{k} + A_h + \frac{M}{k} + B_h\right) \quad kf \in K_{L^2(M)}$$



- | **Stability** of **MNNs** to manifold deformations implies **stability** of **GNNs** with **continuous filters**

$$\|(\mathbf{H}; L; f) - (\mathbf{H}; L^0; f)\|_{KfK_{L^2(M)}} = O\left(\frac{N}{k} + A_h + \frac{M}{k} + B_h\right)$$



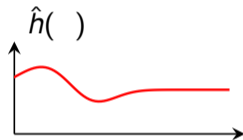
GNNs with continuous filters are stable to deformations

Z. Wang et al., *Stability to Deformations of Manifold Filters and Manifold Neural Networks*, IEEE Trans on Signal Processing

- The generalization gap between **graph empirical risk** and **manifold statistical risk**

$$GA_M = \sup_{H \subseteq \mathcal{H}} \frac{1}{n} \sum_{i=1}^n \ell([H; L_n; \mathbf{x}]_i; [y]_i) - \int_M \ell([H; L; f](x); g(x)) d(x)$$

$$GA_M = O \left(C_L \rho_{\frac{1}{n}} + \frac{\log(1/\epsilon)}{n} + \frac{\log n}{n} \right)^{\frac{1}{d}}$$

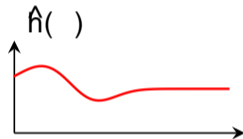


Generalization of Graph Neural Networks via Manifolds

I The generalization gap between **graph empirical risk** and **manifold statistical risk**

$$GA_M = \sup_{H \subseteq \mathcal{H}} \left[\frac{1}{n} \sum_{i=1}^n \ell(H; L_n; x_i; y_i) - \int_M \ell(H; L; f(x); g(x)) d(x) \right]$$

$$GA_M = O \left(C_L p \frac{1}{n} + \frac{p \overline{\log(1/\epsilon)}}{n} + \frac{\log n}{n} \right)^{\frac{1}{d}}$$



GNNs trained on sampled graphs with continuous lters can **generalize to the manifold**

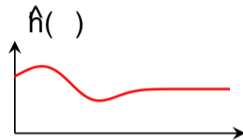
Z. Wang, J. Cerviao et al, A Manifold Perspective on the Statistical Generalization of Graph Neural Networks, arXiv:2406.05225

Generalization of GNNs is Robust to Model Mismatches

- Generalization gap between the **graph empirical risk** and the **mismatched manifold statistical risk**

$$GA = \sup_{H \neq \tilde{H}} \frac{1}{n} \sum_{i=1}^n \left(\ell(H; L_n; x_i; [y]_i) - \int_M \ell(H; L; f(x); g(x)) d\mu(x) \right)$$

$$GA = O \left(C_L \frac{p}{n} + \frac{p \log(1/\epsilon)}{n} + \frac{\log n}{n} \frac{1}{d} \right)$$

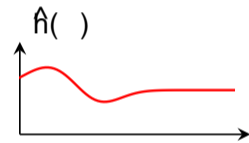


Generalization of GNNs is Robust to Model Mismatches

- Generalization gap between the **graph empirical risk** and the **mismatched manifold statistical risk**

$$GA = \sup_{H \neq H} \frac{1}{n} \sum_{i=1}^n \left(\mathbb{E}_{\mathcal{M}} \left[\ell(H; L_n; x_i; [y]_i) \right] - \mathbb{E}_{\mathcal{M}} \left[\ell(H; L; f(x); g(x)) \right] \right) d(x)$$

$$GA = O \left(C_L \frac{p}{n} + \frac{p \log(1/\epsilon)}{n} + \frac{\log n}{n} \right)^{\frac{1}{d}}$$



Generalization of GNNs is robust to **unseen graphs sampled from mismatched manifolds**

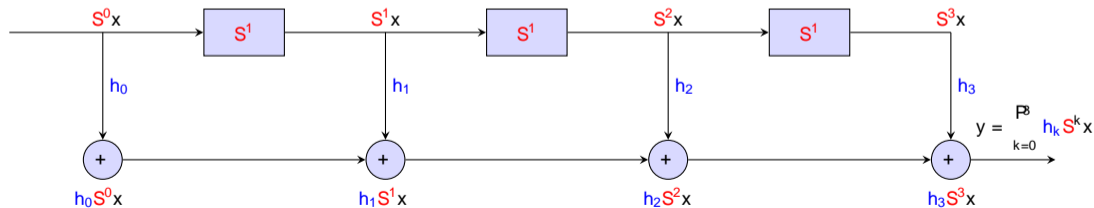
Z. Wang et al, Generalization of Graph Neural Networks is Robust to Model Mismatch, arXiv:2408.13878

Graph Filters and Neural Networks

Graph Convolutional Filters

- Graph G with matrix representation $S \in \mathbb{R}^{n \times n}$ { graph shift operator } and graph signal $x \in \mathbb{R}^n$
- Graph convolutional filter is defined as a summation of iterative graph data distributions

$$y = h_G(S)x = \sum_{k=0}^{P-1} h_k S^k x \quad \{ \text{Iter with coefficients } h_k \}$$



- The matrix S (which is symmetric) admits the **eigenvector decomposition** $S = V \Lambda V^H$

Spectral Representation of Graph Filters

Graph filter with **coefficients** h_k , **graph signal** x and the **filtered signal** y

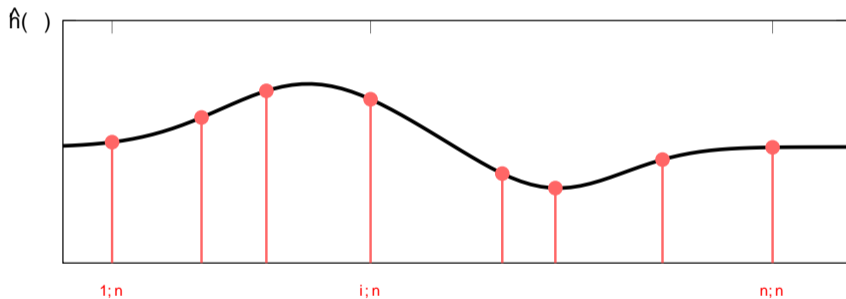
$$y = \sum_{k=0}^{K-1} h_k S^k x = h(S)x$$

The Graph Fourier Transforms (GFTs) $\hat{x} = V^H x$ and $\hat{y} = V^H y$ are related by

$$\hat{y} = \sum_{k=0}^{K-1} h_k \lambda^k \hat{x} = \hat{h}(\lambda) \hat{x}$$

Graph Convolutional Filters in Frequency Domain

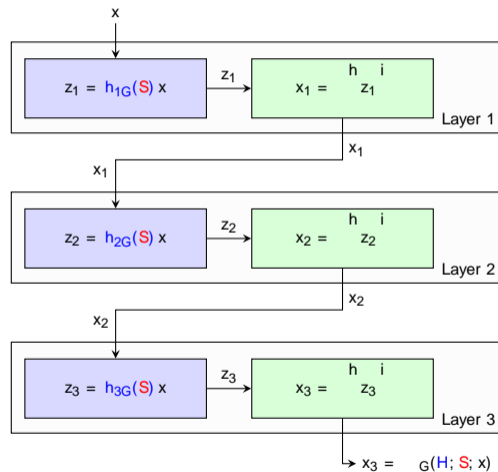
- The graph filter **frequency response** is point-wise on a scalar variable $\hat{h}(\omega) = \sum_{k=0}^{K-1} h_k \omega^k$



- A given graph instantiates the **frequency response** on its given specific eigenvalues
- Eigenvectors** do not appear in the frequency response. They determine **the meaning of frequencies**

Graph Neural Networks (GNNs)

- | Graph neural network is a **cascade of layers**
- | Each of the layers is composed of **graph convolutions** $h_G(\mathbf{S})$ and **pointwise nonlinearities**
- | Define the learnable parameter set \mathbf{H} as \mathbf{H}
- | GNN can be written as a map $\gamma = \gamma_G(\mathbf{H}; \mathbf{S}; x)$



Manifold Filters and Neural Networks

-) Graph convolutions; Spectral representation of graph filters; GNN architecture

Manifold Filters and Neural Networks

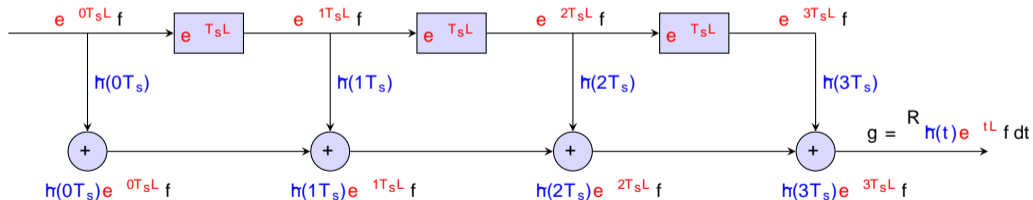
) Graph convolutions; Spectral representation of graph filters; GNN architecture

Manifold Convolutional Filters

| d-dimensional manifold M with Laplace-Beltrami (LB) operator L and manifold signal f

| A Manifold Filter with coefficients h is defined by the input-output relationship

$$g(x) = \int_0^1 h(t) e^{-tL} f(x) dt = h(L) f(x)$$



| d-dimensional manifold M with Laplace-Beltrami (LB) operator L and manifold signal f

| A Manifold Filter with coefficients h is defined by the input-output relationship

$$g(x) = \int_0^1 h(t) e^{-tL} f(x) dt = h(L) f(x)$$

| Manifold convolutions generalize graph convolutions and standard (time) convolutions

) Discretizing a manifold filter yields a graph filter with shift operator $e^{-T_s L_n}$

$$g = \sum_{k=0}^{\infty} h(kT_s) e^{-kT_s L_n} f = \sum_{k=0}^{\infty} h(kT_s) (I - T_s L_n)^k f$$

| d-dimensional manifold M with Laplace-Beltrami (LB) operator L and manifold signal f

| A Manifold Filter with coefficients h is defined by the input-output relationship

$$g(x) = \int_0^1 h(t) e^{-tL} f(x) dt = h(L) f(x)$$

| Manifold convolutions generalize graph convolutions and standard (time) convolutions

) Recover standard convolutions if we make the particular choice $dx = d = dx$

$$g(x) = \int_0^1 h(t) e^{-td} f(x) dt = \int_0^1 h(t) f(x-t) dt$$

- | L is self-adjoint and positive semi-definite, which leads to a discrete spectrum $\lambda_i; i \in \mathbb{N}^+$

Spectral Representation of Manifold Filters

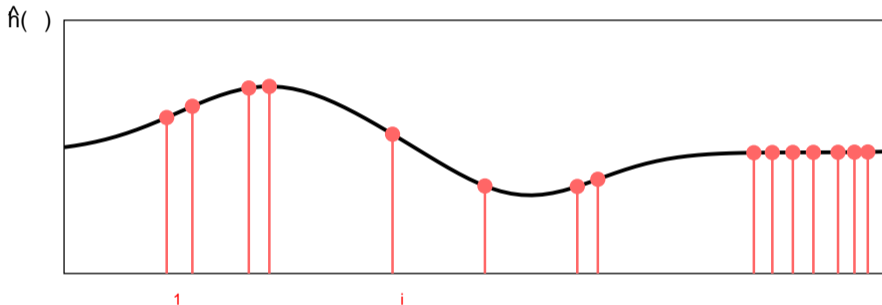
Manifold filter with filter function $h(t)$, manifold signal $f(x)$ and the filtered signal $g(x)$

$$g(x) = \int_0^1 h(t) e^{-tL} dt f(x) = \hat{h}(L) f(x)$$

The frequency components $\hat{f}(i) = \mathcal{H}f; i \in L^2(M)$ and $\hat{g}(i) = \mathcal{H}g; i \in L^2(M)$ are related by

$$\hat{g}(i) = \int_0^1 h(t) e^{-t \lambda_i} dt \hat{f}(i) = \hat{h}(\lambda_i) \hat{f}(i)$$

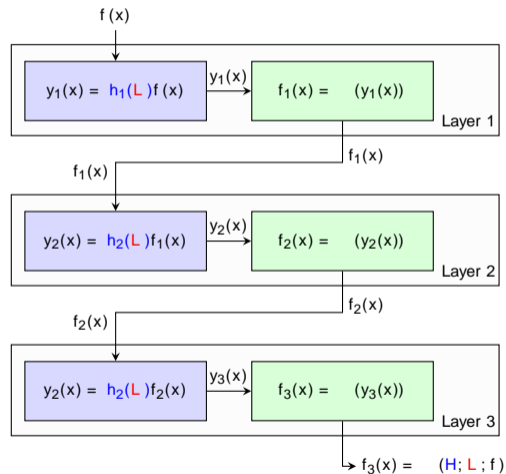
- The manifold filter **frequency response** is point-wise on a scalar variable $\hat{h}(\omega) = \int_0^{R_1} h(t) e^{-i\omega t} dt$



- A given manifold instantiates the **frequency response** on its given specific eigenvalues
- Laplace-Beltrami operator** possesses finite spectrum with $\lambda_i \sim i^{2=d}$ according to **Weyl's law**

Manifold Neural Networks (MNNs)

- Manifold neural network is a **cascade of layers**
- Each of the layers is composed of **manifold convolutions** $h(L)$ and **pointwise nonlinearities**
- Define the learnable parameter set **in** (L) as H
- MNN can be written as a map $y = (H; L; f)$



Transferability of Graph Neural Networks

-) Graph and manifold convolutions; Spectral representation of graph and manifold lters; GNN and MNN architectures

Transferability of Graph Neural Networks

-) Graph and manifold convolutions; Spectral representation of graph and manifold lters; GNN and MNN architectures

- | Geometric graph Iters and GNNs **converge** to their underlying manifold Iters and MNNs
 -) **Convergence** enables **transferability** of geometric GNNs from **small to large** graphs
- | Sample the manifold at $\{x_i\}_{i=1}^n$. Construct graph G_n with edge weights $w_{ij} = K \frac{k(x_i, x_j)}{k^2}$

Gaussian kernel-based graphs

-graphs

- Geometric graph filter is defined by replacing Laplace-Beltrami operator with graph Laplacian,

$$g = \int_0^1 h(t) e^{-tL_n} dt f = h(L_n) f; \quad [f]_i = f(x_i)$$

- Geometric graph neural networks (GNN) = cascading graph filters and non-linearities $(H; L_n; f)$



- Analyze the properties of GNNs and MNNs with the spectral structures of graphs and manifolds

- Geometric graph filter is defined by replacing Laplace-Beltrami operator with graph Laplacian, L_n

$$g = \int_0^1 h(t) e^{-tL_n} dt f = h(L_n) f; \quad [f]_i = f(x_i)$$

- Geometric graph neural networks (\mathbb{G}_n) cascading graph filters and non-linearities ($H; L_n; f$)



- Analyze the properties of GNNs and MNNs with the spectral structures of graphs and manifolds

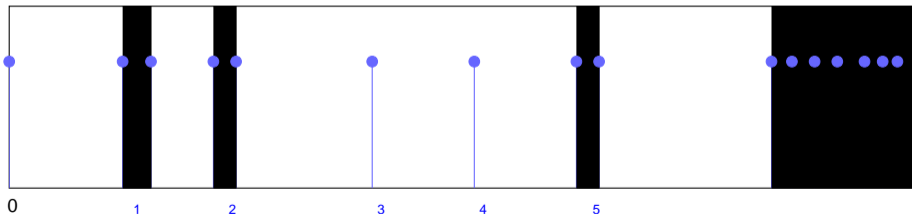
Lipschitz and Frequency Difference Threshold (FDT) Filters

- A filter is A_h -Lipschitz if its frequency response function $\hat{h}(\omega)$ is A_h -Lipschitz continuous

Definition (δ -separated spectrum)

The δ -separated spectrum of a LB operator L is defined as the partition $\{I_k\}_{k \in I}$ such that all $\omega_i \in I_k$ and $\omega_j \in I_l$, $k \neq l$, satisfy

$$|\omega_i - \omega_j| > \delta.$$



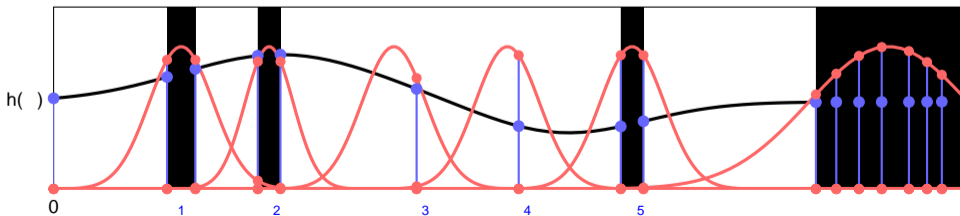
Lipschitz and Frequency Difference Threshold (FDT) Filters

- A filter is A_h -Lipschitz if its frequency response function $\hat{h}(\omega)$ is A_h -Lipschitz continuous

Definition (ϵ -FDT filter)

The frequency response of ϵ -frequency difference threshold (ϵ -FDT) filter $h(\omega)$ satisfies

$$|\hat{h}(\omega_i) - \hat{h}(\omega_j)| \leq \epsilon; \text{ for all } |\omega_i - \omega_j| \geq \delta(\epsilon)$$



Theorem (Convergence of Geometric GNNs)

If an L -layer GNN $(H; L_n; \cdot)$ on G_n and MNN $(H; L; \cdot)$ on M have normalized Lipschitz nonlinearities, it holds in high probability that

$$\| (H; L_n; P_n f) - P_n (H; L; f) \|_{L^2(G_n)} = O \left(\frac{N}{n} + A_h \rho + \frac{\log(n)}{n} \kappa f \kappa_{L^2(M)} \right)$$

with L_n that are ρ -FDT with $\rho = O(\rho)$ and A_h -Lipschitz continuous.

- The properties of large GNNs can be analyzed via MNN Transferability across graph scales
- The error bound shows trade-off between convergence and discriminability nonlinearities lift

Z. Wang et al, Geometric Graph Filters and Neural Networks: Limit Properties and Discriminability Trade-offs, IEEE Trans on Signal Processing

Point cloud Model Classification Convergence Result

- | We evaluate the implementations of GNNs with **ModelNet10 classification**

Z. Wu et al, 3D shapenets: A deep representation for volumetric shapes, IEEE CVPR 2015

- | Compare the **graph output differences** between trained small graphs and large graphs=(1000)

- | GNNs can **converge to** MNNs as more points are sampled; Lipschitz GNNs have smaller differences

- We verify the **transferability** by testing the trained GNNs on graphs with $n = 1000$

Baseline GNN			GF		GNN		Lipschitz GNN	
16:95	5:42	n = 300	21:97	4:17	10:10	1:40	8:60	2:95
13:11	4:97	n = 500	19:83	5:94	7:74	4:05	7:68	3:75
10:02	3:87	n = 700	16:62	2:38	7:92	3:14	8:02	2:77
6:83	3:96	n = 900	13:85	3:81	7:45	4:03	7:44	3:30

Table: Error rates tested on $n = 1000$

- Transferability** allows the GNNs trained on a small graph directly applied to a large graph

Stability of GNNs Implied by MNNs

-) Graph and manifold convolutions; Spectral representation of graph and manifold Iters; GNN and MNN architectures;
-) Transferability of GNNs across resolutions based on the convergence of GNNs to MNNs

Stability of GNNs Implied by MNNs

-) Graph and manifold convolutions; Spectral representation of graph and manifold Iters; GNN and MNN architectures;
-) Transferability of GNNs across resolutions based on the convergence of GNNs to MNNs

Manifold Deformations as Operator Perturbations

- | We investigate the **stability of MNNs** to the **manifold deformations**
 -) Consider **manifold signal** and a **deformation** $(x) \in M$ over the manifold (ϵ -small, ϵ -smooth)
$$p(x) = L^0 f(x) = L g(x) = L f(x)$$
 -) Translate **manifold signal perturbations** as **LB operator perturbations** (ϵ -small)

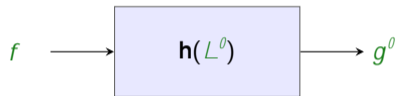
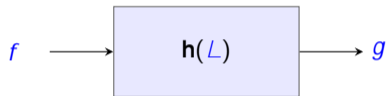
Theorem (Manifold deformations)

Let the **deformation** $(x) : M \rightarrow M$ satisfies $\text{dist}(x; (x)) \leq \epsilon$ and $J((x)) = I + A$ with $\|A\|_F \leq k$. If the gradient field is smooth, it holds that

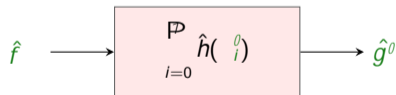
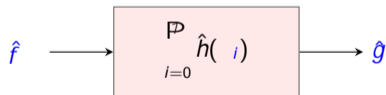
$$L - L^0 = EA + A;$$

where E and A satisfy $\|E\| = O(\epsilon)$ and $\|A\|_{\text{op}} = O(\epsilon)$.

- Manifold filters are parameterized by Laplace-Beltrami operator L and perturbed operator L^0
- In the spatial domain



- In the spectral domain



- Compare the difference of MNNs and perturbed MNNs with the spectral analysis of L and L^0

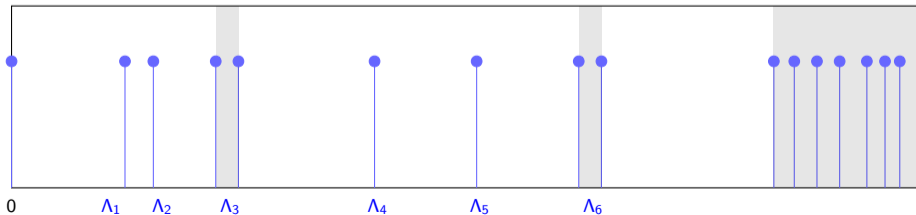
- A filter is B_h -Integral Lipschitz if its frequency response satisfies

$$|\hat{h}(a) - \hat{h}(b)| \leq \frac{B_h |a - b|}{(a + b)^2}; \quad \text{for all } a, b \geq (0; 1)$$

Definition (ϵ -separated spectrum)

The ϵ -separated spectrum of a LB operator L is defined as the partition $\Lambda_1(\epsilon) \cup \dots \cup \Lambda_M(\epsilon)$ such that all $\lambda_i \in \Lambda_k(\epsilon)$ and $\lambda_j \in \Lambda_l(\epsilon)$, $k \neq l$, satisfy

$$\frac{\lambda_i}{\lambda_j} \leq 1 + \epsilon.$$



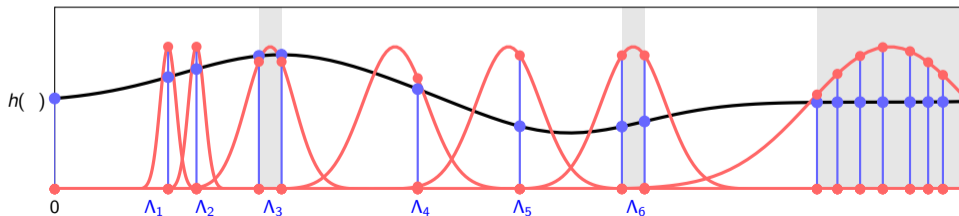
- A filter is B_h -Integral Lipschitz if its frequency response satisfies

$$|\hat{h}(a) - \hat{h}(b)| \leq \frac{B_h |a - b|}{(a + b)^2}; \quad \text{for all } a, b \geq (0; 1)$$

Definition (Λ_k -FRT Filter)

The frequency response of Λ_k -Frequency Ratio Threshold (Λ_k -FRT) filter $h(L)$ satisfies

$$|\hat{h}(i) - \hat{h}(j)| \leq R; \quad \text{for all } i, j \geq \Lambda_k(\cdot)$$



Theorem (Stability of MNNs to deformations)

An L -layer MNN $(\mathbf{H}; L; f)$ have normalized Lipschitz continuous nonlinearities. Let L^0 be the deformed LB operator with $\max_{j=1, \dots, L} f_j$, then

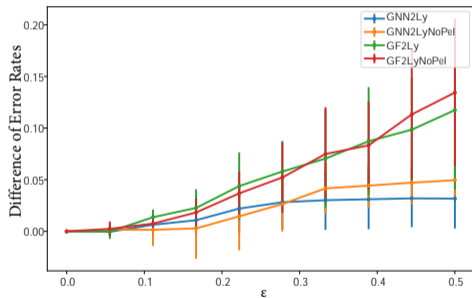
$$\|(\mathbf{H}; L; f) - (\mathbf{H}; L^0; f)\|_{L^2(\mathcal{M})} = O\left(\frac{N}{\epsilon} + A_h + \frac{M}{\epsilon} + B_h \|k\|_{L^2(\mathcal{M})}\right)$$

if the manifold filters are ϵ -FDT with $D = O(\epsilon)$, ϵ -FRT with $R = O(\epsilon)$, A_h -Lipschitz continuous and B_h -integral Lipschitz continuous.

- The difference bound shows a trade-off between stability and discriminability) nonlinearities lift

Z. Wang et al., *Stability to Deformations of Manifold Filters and Manifold Neural Networks*, IEEE Trans on Signal Processing

⌋ We verify the **stability** by comparing the performance on normal and deformed point clouds



Architecture	$\epsilon = 0.2$	$\epsilon = 0.4$
GNN2Ly	7.37% \pm 1.43%	7.71% \pm 3.96%
GF2Ly	13.76% \pm 6.82%	13.54% \pm 7.16%
Architecture	$\epsilon = 0.6$	$\epsilon = 0.8$
GNN2Ly	8.04% \pm 2.83%	11.01% \pm 6.33%
GF2Ly	14.76% \pm 5.67%	16.04% \pm 6.34%

Generalization of GNNs via a Manifold Perspective

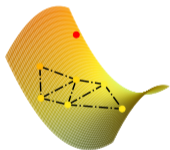
-) Graph and manifold convolutions; Spectral representation of graph and manifold filters; GNN and MNN architectures;
-) Transferability of GNNs across resolutions based on the convergence of GNNs to MNNs
-) Stability of large-scale GNNs implied by stability of MNNs

Generalization of GNNs via a Manifold Perspective

-) Graph and manifold convolutions; Spectral representation of graph and manifold filters; GNN and MNN architectures;
-) Transferability of GNNs across resolutions based on the convergence of GNNs to MNNs
-) Stability of large-scale GNNs implied by stability of MNNs

The generalization gap between **graph empirical risk** and **manifold statistical risk**

$$GA_M = \sup_{\mathcal{H} \geq \mathcal{H}} \frac{1}{n} \sum_{i=1}^n \ell(\mathcal{H}; \mathbf{L}_n; \mathbf{x}_i; [\mathbf{y}]_i) - \int_M \ell(\mathcal{H}; \mathcal{L}; f(x); g(x)) d(x)$$



Theorem (Generalization Gap of Geometric GNNs via Manifold)

If there are GNN $(\mathcal{H}; \mathcal{L}_n; \cdot)$ on \mathbf{G}_n and an MNN $(\mathcal{H}; \mathcal{L}; \cdot)$ on \mathcal{M} , it holds in high probability that

$$GA_M = O \left(C_L \rho_{\frac{1}{n}} + \frac{\rho_{\frac{1}{n}}}{n} + \frac{\log n}{n} \right)^{\frac{1}{d}}$$

with continuous filters and normalized Lipschitz nonlinearities.

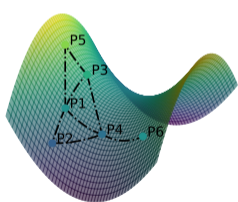
- | The conclusion can be extended to both **node-level** and **graph-level** tasks
- | The practical guidance – **restrictions on the filter continuity** help with the generalization abilities

Generative **model mismatch** between testing and training graphs is inevitable - **robust generalization**

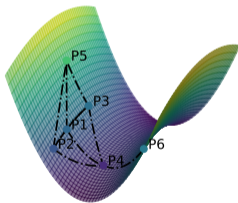
- | See the **manifold mismatches/deformations** as **perturbations on the generated graphs**
 -) **Laplacian operator** perturbations and **node feature** perturbations

$$x \in \mathcal{M} \quad Lf(x) = Lf^0(x); x \in \mathcal{M}$$

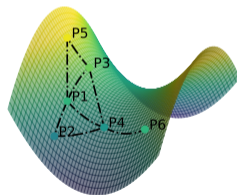
$$Lf(x) = Lf^0(x); x \in \mathcal{M}$$



Graph on \mathcal{M}
 Model
 Mismatch



Perturbed graph
 structure



Perturbed node
 features

- Generalization gap between the **graph empirical risk** and the **mismatched manifold statistical risk**

$$GA = \sup_{\mathcal{H} \subseteq \mathcal{H}} \frac{1}{n} \sum_{i=1}^n \ell(\mathcal{H}; \mathbf{L}_n; \mathbf{x})_i; [\mathbf{y}]_i - \int_{\mathcal{M}} \ell(\mathcal{H}; \mathcal{L}; f)(x); g(x) d \mu(x)$$

Theorem (Robust Generalization of GNNs to Model Mismatch)

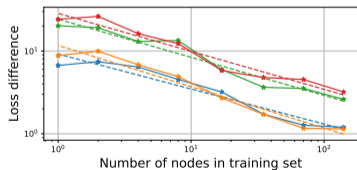
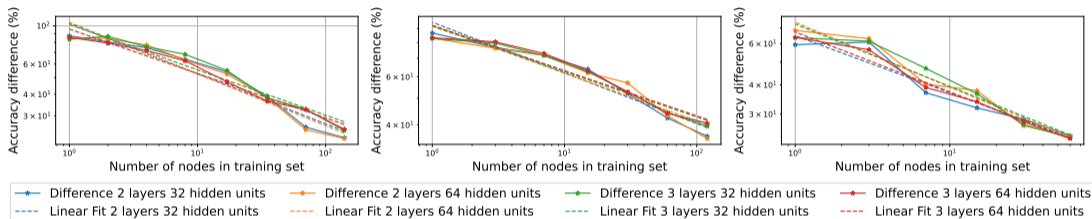
For GNN $(\mathcal{H}; \mathbf{L}_n; \cdot)$ and an $(\mathcal{H}; \mathcal{L}; \cdot)$, suppose the mismatch is **-small** and **-smooth**, then it holds in high probability that

$$GA = O \left(C_L \frac{\epsilon}{n} + \frac{\log(1/\delta)}{n} + \frac{\log n}{n} \right)^{\frac{1}{d}}$$

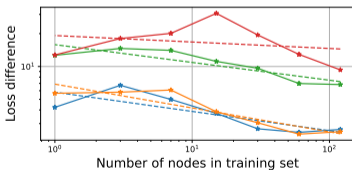
with continuous filters and normalized nonlinearities.

- | We compute the **generalization gap** with a **synthetic chair manifold** by fixing GNN weights

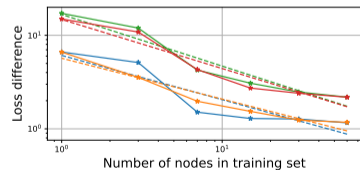
| Generalization gap w.r.t. the number of nodes in the training set for accuracy and loss



(a) Cora

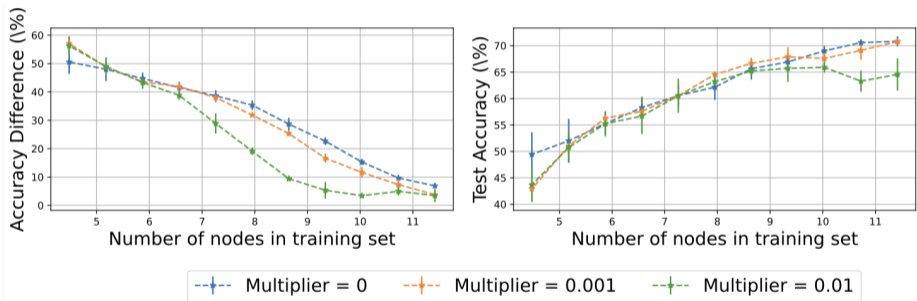


(b) CiteSeer

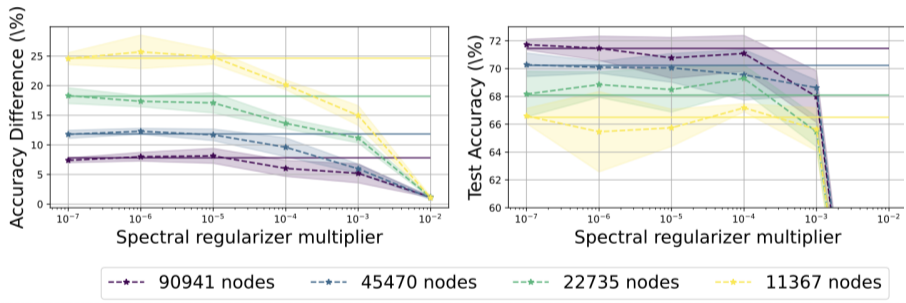


(c) PubMed

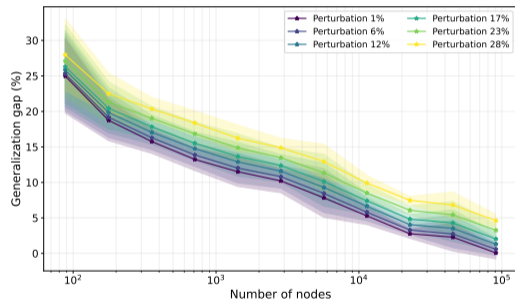
- | Generalization gap and test accuracy w.r.t. the continuity restriction on the filters on the citation network



- Generalization gap and test accuracy w.r.t. the continuity restriction on the filters on the citation network



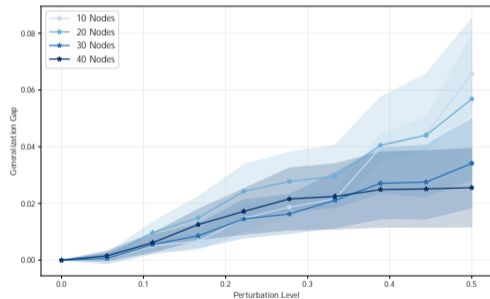
Generalization gap for edge and node perturbations for the Arxiv dataset for a 3 layered, 256 feature GNN



Node feature perturbations

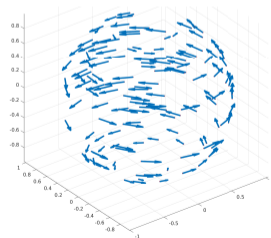
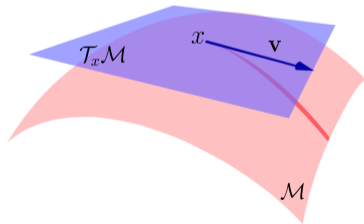
Edge removal perturbations

| Generalization gap w.r.t. the number of nodes and perturbation levels on ModelNet point cloud dataset



- Manifold filters and MNNs process scalar signals over the manifold without covering vector fields
- We define Tangent Bundle convolution with the Connection Laplacian $\Delta F = \sum_{i=1}^n \text{div}_i F_i$
- The tangent bundle filter with impulse response $\tilde{h}: \mathbb{R}^+ \rightarrow \mathbb{R}$ is given by

$$G(x) = \int_0^1 \tilde{h}(t) e^{t\Delta} F(x) dt = \mathbf{h}(\Delta) F(x):$$



Manifold filters and MNNs process scalar signals over the manifold without covering vector fields

We define Tangent Bundle convolution with the Connection Laplacian $\Delta F = \sum_{i=1}^n \text{div}_i F_i$

The tangent bundle filter with impulse response $\tilde{h}: \mathbb{R}^+ \rightarrow \mathbb{R}$ is given by

$$G(x) = \int_0^1 \tilde{h}(t) e^{t\Delta} F(x) dt = \mathbf{h}(\Delta) F(x)$$

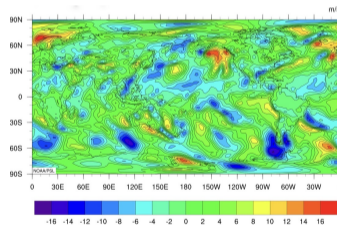
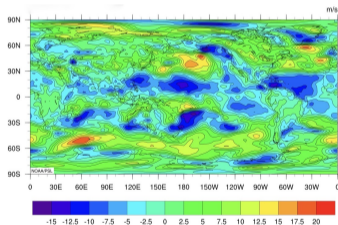
Tangent bundle Fourier Transform is the projections $\int_0^1 F_i = h F_i$

Frequency response of tangent bundle filter \mathbf{h} is $\hat{h}(\omega) = \int_0^1 \tilde{h}(t) e^{-t\omega} dt$

Theorem (Tangent bundle Filters in the Spectral Domain)

Tangent bundle filters are pointwise in the spectral domain $G_i = \hat{h}(\omega) F_i$

		$E_{fng} = 0.5n$		$E_{fng} = 0.3n$		$E_{fng} = 0.1n$	
$E_{fng} = 200$	DD-TNN	$1.99 \cdot 10^2$	$2.30 \cdot 10^3$	$1.18 \cdot 10^2$	$1.62 \cdot 10^3$	$3.67 \cdot 10^3$	$1.23 \cdot 10^3$
	MNN	$3.19 \cdot 10^2$	$1.31 \cdot 10^2$	$2.74 \cdot 10^2$	$1.55 \cdot 10^2$	$2.58 \cdot 10^2$	$1.82 \cdot 10^2$
	MLP	$2.03 \cdot 10^2$	$2.28 \cdot 10^3$	$1.20 \cdot 10^2$	$1.68 \cdot 10^3$	$3.69 \cdot 10^3$	$1.17 \cdot 10^3$
$E_{fng} = 300$	DD-TNN	$1.88 \cdot 10^2$	$1.72 \cdot 10^3$	$1.13 \cdot 10^2$	$1.54 \cdot 10^3$	$3.96 \cdot 10^3$	$1.00 \cdot 10^3$
	MNN	$2.68 \cdot 10^2$	$7.64 \cdot 10^3$	$2.41 \cdot 10^2$	$1.05 \cdot 10^2$	$2.09 \cdot 10^2$	$1.76 \cdot 10^2$
	MLP	$1.95 \cdot 10^2$	$1.74 \cdot 10^3$	$1.18 \cdot 10^2$	$1.56 \cdot 10^3$	$4.00 \cdot 10^3$	$8.85 \cdot 10^4$
$E_{fng} = 400$	DD-TNN	$1.95 \cdot 10^2$	$1.66 \cdot 10^3$	$1.14 \cdot 10^2$	$1.38 \cdot 10^3$	$3.70 \cdot 10^3$	$8.55 \cdot 10^4$
	MNN	$2.48 \cdot 10^2$	$6.55 \cdot 10^3$	$2.52 \cdot 10^2$	$1.20 \cdot 10^2$	$8.16 \cdot 10^2$	$1.87 \cdot 10^1$
	MLP	$2.01 \cdot 10^2$	$1.66 \cdot 10^3$	$1.19 \cdot 10^2$	$1.24 \cdot 10^3$	$3.81 \cdot 10^3$	$8.46 \cdot 10^4$



C. Battiloro, Z. Wang. et al., *Tangent bundle convolutional learning: from manifolds to cellular sheaves and back*, IEEE Trans on Signal Processing

- | We introduce **manifold neural networks** (MNNs) as the limits of graph neural networks
- | And study their fundamental properties:
 -) **Resolution:** GNNs converge to MNNs) the transferability of GNNs across scales
 -) **Deformation:** MNNs are stable to deformations) the stability of large-scale GNNs
 -) **Robust generalization:** GNNs can generalize robustly to unseen data over the manifold
- | Informs the **practical design of graph neural networks** for large-scale geometric graphs
 -) Point-cloud analysis, Wireless communications, Wind field reconstructions etc.

1. **Z. Wang**^y, J. Cerviño^y, and A. Ribeiro, “A Manifold Perspective on the Statistical Generalization of Graph Neural Networks,” *arXiv*, arXiv:2406.05225, 2024.
2. **Z. Wang**, J. Cerviño, and A. Ribeiro, “Generalization of Graph Neural Networks is Robust to Model Mismatch,” accepted at the 39th Annual AAAI Conference on Artificial Intelligence (AAAI 2025).
3. **Z. Wang**, L. Ruiz, and A. Ribeiro, “Geometric Graph Filters and Neural Networks: Limit Properties and Discriminability Trade-offs,” *IEEE Transactions on Signal Processing*, vol. 72, pp. 2244-2259, 2024.
4. C. Battiloro, **Z. Wang**, H. Riess, P. Di Lorenzo, and A. Ribeiro, “Tangent Bundle Convolutional Learning: from Manifolds to Cellular Sheaves and Back,” *IEEE Transactions on Signal Processing*, vol. 72, pp. 1892-1909, 2024.
5. **Z. Wang**, L. Ruiz, and A. Ribeiro, “Stability to Deformations of Manifold Filters and Manifold Neural Networks,” *IEEE Transactions on Signal Processing*, vol. 72, pp. 2130-2146, 2024.