Tangent Bundle Filters and Neural Networks: from Manifolds to Cellular Sheaves and Back

Claudio Battiloro[1,2], Zhiyang Wang[1], Hans Riess[3], Paolo Di Lorenzo[2], Alejandro Ribeiro[1]

[1] *ESE Dpt., University of Pennsylvania*

[2] *DIET Dpt., Sapienza University of Rome*

[3] *PSE, Duke University*

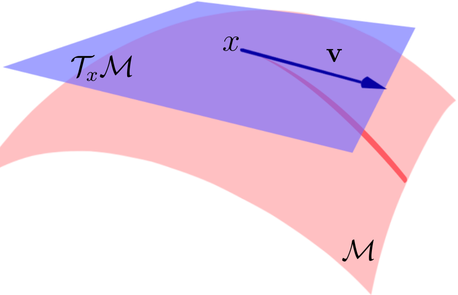
{clabat, zhiyangw}@seas.upenn.edu

GSP Workshop

June 12-14, 2023, Oxford, UK

- We introduce a convolution operation over the Tangent Bundle of Riemannian manifolds exploiting the Connection Laplacian operator
- We define Tangent Bundle Filters and Tangent Bundle Neural Networks (TNNs), novel architectures operating on tangent bundle signals, i.e. vector fields over manifolds
- We discretize TNNs both in space and time, showing that their discrete counterpart is a novel principled variant of the recently introduced Sheaf Neural Networks
- We prove that the discrete architecture converges to the underlying continuous TNN

- We consider a compact and smooth d-dimensional manifold $\mathcal{M}$ embedded in $\mathbb{R}^p$
- Each point $x \in \mathcal{M}$ is endowed with a $d-$dimensional tangent (vector) space $\mathcal{T}_x\mathcal{M} \cong \mathbb{R}^d$
- $\mathbf{v} \in \mathcal{T}_x\mathcal{M}$ is said to be a tangent vector at x
- Tangent vectors can be seen as the velocity vector of a curve over $\mathcal{M}$ passing through the point x

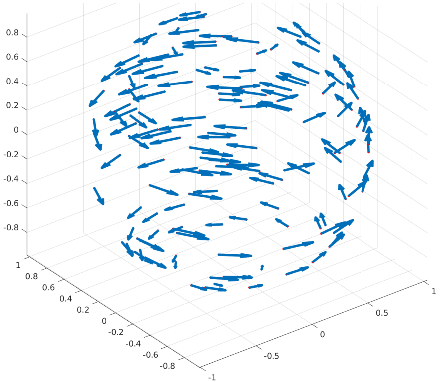


- We consider a compact and smooth d-dimensional manifold $\mathcal{M}$ embedded in $\mathbb{R}^p$
- Each point $x \in \mathcal{M}$ is endowed with a $d-$dimensional tangent (vector) space $\mathcal{T}_x\mathcal{M} \cong \mathbb{R}^d$
- $\mathbf{v} \in \mathcal{T}_x\mathcal{M}$ is said to be a tangent vector at x
- The disjoint union of the tangent spaces is called the tangent bundle $\mathcal{T}\mathcal{M} = \bigsqcup_{x\in\mathcal{M}} \mathcal{T}_x\mathcal{M}$
- Each tangent space $\mathcal{T}_x\mathcal{M}$ with a Riemann metric given, for each $\mathbf{v}$,$\mathbf{w} \in \mathcal{T}_x\mathcal{M}$, by

$$\langle \mathbf{v}, \mathbf{w} \rangle_{\mathcal{T}_x\mathcal{M}} = \boldsymbol{i}\mathbf{v} \cdot \boldsymbol{i}\mathbf{w},$$

  where $\boldsymbol{i}\mathbf{v} \in \mathcal{T}_x\mathbb{R}^p$ is the embedding of $\mathbf{v} \in \mathcal{T}_x\mathcal{M}$ in $\mathcal{T}_x\mathbb{R}^p \subset \mathbb{R}^p$ (the d-dimensional subspace of $\mathbb{R}^p$ which is the embedding of $\mathcal{T}_x\mathcal{M}$ in $\mathbb{R}^p$)
- The Riemann metric induces a probability measure μ over the manifold

- A Tangent Bundle Signal is a vector field over the manifold, thus a mapping $\mathbf{F} : \mathcal{M} \to \mathcal{T}\mathcal{M}$ that associates to each point of the manifold a vector in the corresponding tangent space



- A Tangent Bundle Signal is a vector field over the manifold, thus a mapping $\mathbf{F} : \mathcal{M} \to \mathcal{T}\mathcal{M}$ that associates to each point of the manifold a vector in the corresponding tangent space

- An inner product for tangent bundle signals $\mathbf{F}$ and $\mathbf{G}$ is

$$\langle \mathbf{F}, \mathbf{G} \rangle_{\mathcal{T}\mathcal{M}} = \int_{\mathcal{M}} \langle \mathbf{F}(x), \mathbf{G}(x) \rangle_{\mathcal{T}_x\mathcal{M}} \mathrm{d}\mu(x),$$

  and the induced norm is $||\mathbf{F}||^2_{\mathcal{T}\mathcal{M}} = \langle \mathbf{F}, \mathbf{F} \rangle_{\mathcal{T}\mathcal{M}}$

- We denote with $\mathcal{L}^2(\mathcal{T}\mathcal{M})$ the Hilbert Space of finite energy tangent bundle signals

- The Connection Laplacian is a (second-order) operator $\Delta : \mathcal{L}^2(\mathcal{T}\mathcal{M}) \to \mathcal{L}^2(\mathcal{T}\mathcal{M})$, given by the trace of the second covariant derivative defined via the Levi-Cita connection

- It is a means to diffuse vectors from one tangent space to another, because it encodes:
  - when tangent vectors are "parallel" (via the Connection)
  - how to "move" them keeping them parallel (via the induced Parallel Transport operator)

- A Tangent Bundle Signal is a vector field over the manifold, thus a mapping $\mathbf{F} : \mathcal{M} \rightarrow \mathcal{T}\mathcal{M}$ that associates to each point of the manifold a vector in the corresponding tangent space

- An inner product for tangent bundle signals $\mathbf{F}$ and $\mathbf{G}$ is

$$\langle \mathbf{F}, \mathbf{G} \rangle_{\mathcal{T}\mathcal{M}} = \int_{\mathcal{M}} \langle \mathbf{F}(x), \mathbf{G}(x) \rangle_{\mathcal{T}_x\mathcal{M}} \mathrm{d}\mu(x),$$

  and the induced norm is $||\mathbf{F}||^2_{\mathcal{T}\mathcal{M}} = \langle \mathbf{F}, \mathbf{F} \rangle_{\mathcal{T}\mathcal{M}}$

- We denote with $\mathcal{L}^2(\mathcal{T}\mathcal{M})$ the Hilbert Space of finite energy tangent bundle signals

- The Connection Laplacian is a (second-order) operator $\Delta : \mathcal{L}^2(\mathcal{T}\mathcal{M}) \rightarrow \mathcal{L}^2(\mathcal{T}\mathcal{M})$, given by the trace of the second covariant derivative defined via the Levi-Cita connection

- It is a means to diffuse vectors from one tangent space to another, because it encodes:
  - when tangent vectors are "parallel" (via the Connection)
  - how to "move" them keeping them parallel (via the induced Parallel Transport operator)

- The Connection Laplacian characterize the vector heat equation over manifolds, governing the diffusion of tangent vectors:

$$\frac{\partial \mathbf{U}(x,t)}{\partial t} - \Delta \mathbf{U}(x,t) = 0,$$

where $\mathbf{U} : \mathcal{M} \times \mathbb{R}_0^+ \rightarrow \mathcal{T}\mathcal{M}$ and $\mathbf{U}(\cdot, t) \in \mathcal{L}^2(\mathcal{T}\mathcal{M}) \, \forall t \in \mathbb{R}_0^+$

- With initial condition set as $\mathbf{U}(x,0) = \mathbf{F}(x)$, the solution is given by

$$\mathbf{U}(x,t) = e^{t\Delta}\mathbf{F}(x),$$

- The Connection Laplacian characterize the vector heat equation over manifolds, governing the diffusion of tangent vectors:

$$\frac{\partial \mathbf{U}(x,t)}{\partial t} - \Delta \mathbf{U}(x,t) = 0,$$

where $\mathbf{U} : \mathcal{M} \times \mathbb{R}_0^+ \rightarrow \mathcal{TM}$ and $\mathbf{U}(\cdot, t) \in \mathcal{L}^2(\mathcal{TM}) \, \forall t \in \mathbb{R}_0^+$

- With initial condition set as $\mathbf{U}(x,0) = \mathbf{F}(x)$, the solution is given by

$$\mathbf{U}(x,t) = e^{t\Delta}\mathbf{F}(x),$$

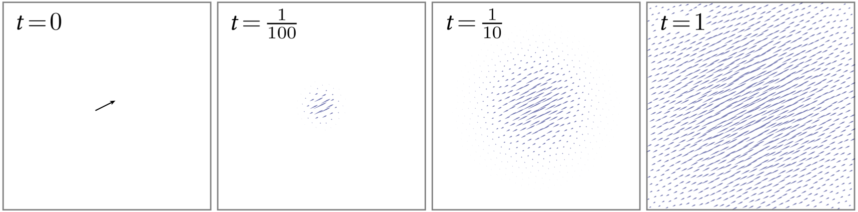


Figure: *"The Vector Heat Method", Sharp et al., ACM ToG, 2019*

- The Connection Laplacian characterize the vector heat equation over manifolds, governing the diffusion of tangent vectors:

$$\frac{\partial \mathbf{U}(x,t)}{\partial t} - \Delta \mathbf{U}(x,t) = 0,$$

where $\mathbf{U} : \mathcal{M} \times \mathbb{R}_0^+ \rightarrow \mathcal{TM}$ and $\mathbf{U}(\cdot, t) \in \mathcal{L}^2(\mathcal{TM}) \, \forall t \in \mathbb{R}_0^+$

- With initial condition set as $\mathbf{U}(x,0) = \mathbf{F}(x)$, the solution is given by

$$\mathbf{U}(x,t) = e^{t\Delta}\mathbf{F}(x)$$

- Δ is a negative semidefinite, self-adjoint and elliptic operator, and this leads to a discrete spectrum $\{-\lambda_i, \phi_i\}_{i=1}^{\infty}$, such that:

$$\Delta \mathbf{F} = \sum_{i=1}^{\infty} -\lambda_i \langle \mathbf{F}, \boldsymbol{\phi_i} \rangle_{\mathcal{TM}} \boldsymbol{\phi_i}$$

Definition (Tangent Bundle Convolutional Filter)

The tangent bundle filter with impulse response $\tilde{h} : \mathbb{R}^+ \to \mathbb{R}$, denoted as $\mathbf{h}$, is given by

$$\mathbf{G}(x) = (\mathbf{hF})(x) := (\tilde{h} \star_{\mathcal{TM}} \mathbf{F})(x) := \int_0^\infty \tilde{h}(t)\mathbf{U}(x,t)\mathrm{d}t,$$

where $\tilde{h} \star_{\mathcal{M}} \mathbf{F}$ is the manifold convolution of $\tilde{h}$ and $\mathbf{F}$, $\mathbf{U}(x,t)$ is the solution of the heat equation

Injecting the heat equation solution, we can express the convolution with a parametric map

$$\mathbf{G}(x) = (\mathbf{hF})(x) = \int_0^\infty \tilde{h}(t)e^{-t\Delta}\mathbf{F}(x)\mathrm{d}t = \mathbf{h}(\Delta)\mathbf{F}(x)$$

Definition (Tangent Bundle Convolutional Filter)

The tangent bundle filter with impulse response $\tilde{h} : \mathbb{R}^+ \to \mathbb{R}$, denoted as $\mathbf{h}$, is given by

$$\mathbf{G}(x) = (\mathbf{h}\mathbf{F})(x) := (\tilde{h} \star_{\mathcal{TM}} \mathbf{F})(x) := \int_0^\infty \tilde{h}(t)\mathbf{U}(x,t)\mathrm{d}t,$$

where $\tilde{h} \star_{\mathcal{M}} \mathbf{F}$ is the manifold convolution of $\tilde{h}$ and $\mathbf{F}$, $\mathbf{U}(x,t)$ is the solution of the heat equation

Injecting the heat equation solution, we can express the convolution with a parametric map

$$\mathbf{G}(x) = (\mathbf{h}\mathbf{F})(x) = \int_0^\infty \tilde{h}(t)e^{-t\Delta}\mathbf{F}(x)\mathrm{d}t = \mathbf{h}(\Delta)\mathbf{F}(x)$$

- We project the convolution input and output functions onto the eigenvectorfields ϕ_i

$$[\hat{G}]_i = \langle \mathbf{G}, \phi_i \rangle = \int_0^\infty \tilde{h}(t) e^{-t\lambda_i} \mathrm{d}t [\hat{F}]_i$$

Definition (Frequency Response)

Given a tangent bundle filter $\mathbf{h}(\Delta)$, the frequency response of this filter can be written as

$$\hat{h}(\lambda) = \int_0^\infty \tilde{h}(t) e^{-t\lambda} \mathrm{d}t$$

- The manifold filter $\mathbf{h}(\Delta)$ is pointwise in the frequency domain as $[\hat{G}]_i = \hat{h}(\lambda_i)[\hat{\mathbf{F}}]_i$

- We project the convolution input and output functions onto the eigenvectorfields ϕ_i

$$\left[\hat{G}\right]_i = \langle \mathbf{G}, \phi_i \rangle = \int_0^{\infty} \tilde{h}(t) e^{-t\lambda_i} \mathrm{d}t \left[\hat{F}\right]_i$$

Definition (Frequency Response)

Given a tangent bundle filter $\mathbf{h}(\Delta)$, the frequency response of this filter can be written as

$$\hat{h}(\lambda) = \int_0^{\infty} \tilde{h}(t) e^{-t\lambda} \mathrm{d}t$$

- The manifold filter $\mathbf{h}(\Delta)$ is pointwise in the frequency domain as $\left[\hat{G}\right]_i = \hat{h}(\lambda_i)\left[\hat{\mathbf{F}}\right]_i$

- We project the convolution input and output functions onto the eigenvectorfields $\boldsymbol{\phi_i}$

$$\left[\hat{G}\right]_i = \langle \mathbf{G}, \phi_i \rangle = \int_0^\infty \tilde{h}(t) e^{-t\lambda_i} \mathrm{d}t \left[\hat{F}\right]_i$$

Definition (Frequency Response)

Given a tangent bundle filter $\mathbf{h}(\Delta)$, the frequency response of this filter can be written as

$$\hat{h}(\lambda) = \int_0^\infty \tilde{h}(t) e^{-t\lambda} \mathrm{d}t$$

- The manifold filter $\mathbf{h}(\Delta)$ is pointwise in the frequency domain as $\left[\hat{G}\right]_i = \hat{h}(\lambda_i)\left[\hat{\mathbf{F}}\right]_i$

- We project the convolution input and output functions onto the eigenvectorfields $\boldsymbol{\phi_i}$

$$\left[\hat{G}\right]_i = \langle \mathbf{G}, \phi_i \rangle = \int_0^\infty \tilde{h}(t) e^{-t\lambda_i} \mathrm{d}t \left[\hat{F}\right]_i$$

Definition (Bandlimited tangent bundle signal)

A tangent bundle signal is defined as λ_M-bandlimitd with $\lambda_M > 0$ if $\left[\hat{F}\right]_i = 0$ for all i such that $\lambda_i > \lambda_M$.

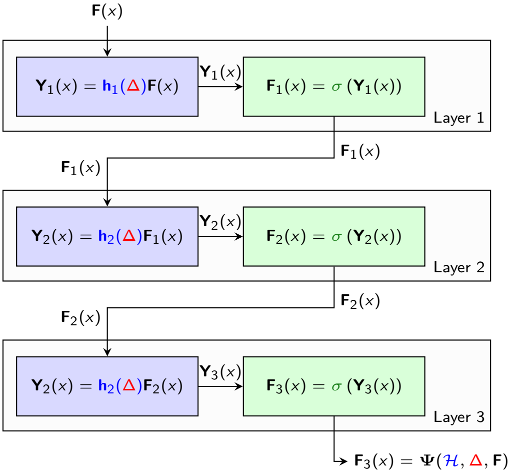
- The manifold filter $\mathbf{h}(\Delta)$ is pointwise in the frequency domain as $\left[\hat{G}\right]_i = \hat{h}(\lambda_i)\left[\hat{\mathbf{F}}\right]_i$

- A Tangent Bundle Neural Network (TNN) is a cascade of L layers
- Each of the layer is composed of
  - Tangent Bundle convolutions $\mathbf{h}(\Delta)$
  - Pointwise nonlinearities σ
- Define the learnable parameter set in $\mathbf{h}(\Delta)$ as $\mathcal{H}$
- TNN can be written as a map $\mathbf{Y} = \Psi(\mathcal{H}, \Delta, \mathbf{F})$

- A Tangent Bundle Neural Network (TNN) is a cascade of L layers
- Each of the layer is composed of
  - Tangent Bundle convolutions $\mathbf{h}(\Delta)$
  - Pointwise nonlinearities σ
- Define the learnable parameter set in $\mathbf{h}(\Delta)$ as $\mathcal{H}$
- TNN can be written as a map $\mathbf{Y} = \Psi(\mathcal{H}, \Delta, \mathbf{F})$

- A Tangent Bundle Neural Network (TNN) is a cascade of L layers
- Each of the layer is composed of
  - Tangent Bundle convolutions $\mathbf{h}(\Delta)$
  - Pointwise nonlinearities σ
- Define the learnable parameter set in $\mathbf{h}(\Delta)$ as $\mathcal{H}$
- TNN can be written as a map $\mathbf{Y} = \Psi(\mathcal{H}, \Delta, \mathbf{F})$

- A Tangent Bundle Neural Network (TNN) is a cascade of L layers
- Each of the layer is composed of
  - Tangent Bundle convolutions $\mathbf{h}(\Delta)$
  - Pointwise nonlinearities σ
- Define the learnable parameter set in $\mathbf{h}(\Delta)$ as $\mathcal{H}$
- TNN can be written as a map $\mathbf{Y} = \boldsymbol{\Psi}(\mathcal{H}, \Delta, \mathbf{F})$



Cellular Sheaves in a (too tiny) Nutshell

- A cellular sheaf over an undirected graph consists of an assignment of a vector space to each node and edge in the graph and a map between these spaces for each incident node-edge pair

- Given an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $|\mathcal{V}| = n$, a cellular sheaf $\mathcal{TM}_n = (\mathcal{G}, \mathcal{F})$ is:
  - A vector space $\mathcal{F}(v)$ for each $v \in \mathcal{V}$. We refer to these vector spaces as nodes stalks
  - A vector space $\mathcal{F}(e)$ for each $e \in \mathcal{E}$. We refer to these vector spaces as edges stalks
  - A linear mapping $\mathbf{V}_{v,e}^T : \mathcal{F}(v) \to \mathcal{F}(e)$ for each incident $v \trianglelefteq e$ node-edge pair. We refer to these mappings as restriction maps

- All the spaces associated with the nodes of the graph form the space of sheaf signals $\mathcal{L}^2(\mathcal{TM}_n)$

- The Sheaf Laplacian of a sheaf $\mathcal{TM}_n$ is a linear mapping $\Delta_n : \mathcal{L}^2(\mathcal{TM}_n) \to \mathcal{L}^2(\mathcal{TM}_n)$ defined node-wise. In particular, given a sheaf signal $\mathbf{f}_n$, it holds:

$$(\Delta_n \mathbf{f}_n)(v) = \sum_{v,u \trianglelefteq e} \mathbf{V}_{v,e}^T (\mathbf{V}_{v,e} \mathbf{f}_n(v) - \mathbf{V}_{u,e} \mathbf{f}_n(u))$$

- In this work, we focus on orthogonal cellular sheaves, i.e. sheaves with orthogonal restriction maps and stalks with same dimension d

- Orthogonal Cellular Sheaves connecting the points can capture the geometric structure $\rightarrow$ they can be seen as discretized manifolds and approximated tangent bundles
- $\mathcal{X} = \{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^p$ are n points sampled uniformly over the manifold $\mathcal{M}$
- We first build a geometric weighted graph connecting points x_i and x_j with weigts:

$$w_{i,j} = \exp\left(\frac{||x_i - x_j||^2}{\sqrt{\epsilon}}\right)\mathbb{I}\left(0 < ||x_i - x_j||^2 \leq \sqrt{\epsilon}\right)$$

- The graph is not sufficient to correctly approximate the manifold and its tangent bundle $\rightarrow$ we need to equip it with nodes stalks, edge stalks and restriction maps

- Orthogonal Cellular Sheaves connecting the points can capture the geometric structure $\rightarrow$ they can be seen as discretized manifolds and approximated tangent bundles

- $\mathcal{X} = \{x_1, x_2, \dots, x_n\} \subset \mathbb{R}^p$ are n points sampled uniformly over the manifold $\mathcal{M}$

- We first build a geometric weighted graph connecting points x_i and x_j with weigts:

$$w_{i,j} = \exp\left(\frac{||x_i - x_j||^2}{\sqrt{\epsilon}}\right)\mathbb{I}\left(0 < ||x_i - x_j||^2 \leq \sqrt{\epsilon}\right)$$

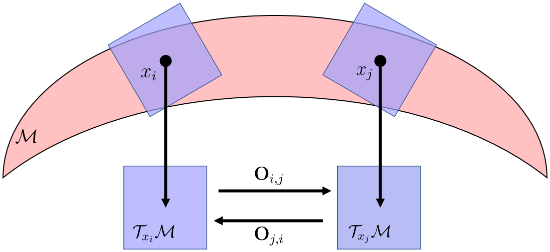
- The graph is not sufficient to correctly approximate the manifold and its tangent bundle $\rightarrow$ we need to equip it with nodes stalks, edge stalks and restriction maps

Discretization in the Space Domain 2

- We assign to each node i an orthogonal transformation $\mathbf{O}_i \in \mathbb{R}^{p \times d}$ computed via a local PCA procedure (from *"Vector diffusion maps and the Connection Laplacian", Singer, Wu, 2012*)
- $\mathbf{O}_i$ is a basis of the i-th node stalk and represents an approximation of a basis of the tangent space $\mathcal{T}_{x_i}\mathcal{M}$
- The restriction maps of the edge (i, j) are given by the SVD $\mathbf{M}_{i,j}$ and right $\mathbf{V}_{i,j}^T$ of $\mathbf{O}_i{}^T\mathbf{O}_j$
- $\mathbf{O}_{i,j} = \mathbf{M}_{i,j}\mathbf{V}_{i,j}^T$ represents an approximated transport operator from x_i to x_j

- We assign to each node i an orthogonal transformation $\mathbf{O}_i \in \mathbb{R}^{p \times d}$ computed via a local PCA procedure (from *"Vector diffusion maps and the Connection Laplacian", Singer, Wu, 2012*)
- $\mathbf{O}_i$ is a basis of the i-th node stalk and represents an approximation of a basis of the tangent space $\mathcal{T}_{x_i}\mathcal{M}$
- The restriction maps of the edge (i,j) are given by the SVD $\mathbf{M}_{i,j}$ and right $\mathbf{V}_{i,j}^T$ of $\mathbf{O}_i{}^T\mathbf{O}_j$
- $\mathbf{O}_{i,j} = \mathbf{M}_{i,j}\mathbf{V}_{i,j}^T$ represents an approximated transport operator from x_i to x_j

- We assign to each node i an orthogonal transformation $\mathbf{O}_i \in \mathbb{R}^{p\times d}$ computed via a local PCA procedure (from *"Vector diffusion maps and the Connection Laplacian", Singer, Wu, 2012*)
- $\mathbf{O}_i$ is a basis of the i-th node stalk and represents an approximation of a basis of the tangent space $\mathcal{T}_{x_i}\mathcal{M}$
- The restriction maps of the edge (i,j) are given by the SVD $\mathbf{M}_{i,j}$ and right $\mathbf{V}_{i,j}^T$ of $\mathbf{O}_i{}^T\mathbf{O}_j$
- $\mathbf{O}_{i,j} = \mathbf{M}_{i,j}\mathbf{V}_{i,j}^T$ represents an approximated transport operator from x_i to x_j



- We build block matrix $\mathbf{S} \in \mathbb{R}^{nd \times nd}$ and a diagonal block matrix $\mathbf{D} \in \mathbb{R}^{nd \times nd}$ with blocks defined as

$$\mathbf{S}_{i,j} = w_{i,j}\widetilde{\mathbf{D}}_i^{-1}\mathbf{O}_{i,j}\widetilde{\mathbf{D}}_j^{-1}, \quad \mathbf{D}_{i,i} = \mathrm{ndeg(i)}\mathbf{I}_d,$$

where $\widetilde{\mathbf{D}}_i = \mathrm{deg}(i)\mathbf{I}_d$, $\mathrm{deg}(i) = \sum_j w_{i,j}$, and $\mathrm{ndeg}(i) = \sum_j w_{i,j}/(\mathrm{deg}(i)\mathrm{deg}(j))$

- Finally, the (normalized) Sheaf Laplacian is the following block matrix

$$\Delta_n = \epsilon^{-1}\left(\mathbf{D}^{-1}\mathbf{S} - \mathbf{I}\right) \in \mathbb{R}^{nd \times nd}$$

- The Sheaf Laplacians Δ_n is proved to converge to the Connection Laplacian of manifold $\mathcal{M}$

- We build block matrix $\mathbf{S} \in \mathbb{R}^{nd \times nd}$ and a diagonal block matrix $\mathbf{D} \in \mathbb{R}^{nd \times nd}$ with blocks defined as

$$\mathbf{S}_{i,j} = w_{i,j}\widetilde{\mathbf{D}}_i^{-1}\mathbf{O}_{i,j}\widetilde{\mathbf{D}}_j^{-1}, \quad \mathbf{D}_{i,i} = \mathrm{ndeg(i)}\mathbf{I}_d,$$

where $\widetilde{\mathbf{D}}_i = \mathrm{deg}(i)\mathbf{I}_d$, $\mathrm{deg}(i) = \sum_j w_{i,j}$, and $\mathrm{ndeg}(i) = \sum_j w_{i,j}/(\mathrm{deg}(i)\mathrm{deg}(j))$

- Finally, the (normalized) Sheaf Laplacian is the following block matrix

$$\Delta_n = \epsilon^{-1}\left(\mathbf{D}^{-1}\mathbf{S} - \mathbf{I}\right) \in \mathbb{R}^{nd \times nd}$$

- The Sheaf Laplacians Δ_n is proved to converge to the Connection Laplacian of manifold $\mathcal{M}$

- In our context, a sheaf signal $\mathbf{f}_n$ is defined as a sampled version of a tangent bundle signal $\mathbf{F}$

$$\mathbf{f}_n = \mathbf{\Omega}_n^{\mathcal{X}} \mathbf{F} \in \mathbb{R}^{nd},$$

$$\mathbf{f}_n(x_i) := [\mathbf{f}_n]_{((i-1)d+1):(i+1)d} = \mathbf{O}_i{}^T i\mathbf{F}(x_i), x_i \in \mathcal{X}$$

- We can define a discrete tangent bundle filter as

$$\mathbf{g}_n = \int_0^{\infty} \widetilde{h}(t) e^{t\Delta_n} \mathrm{d}t \mathbf{f}_n = \mathbf{h}(\Delta_n)\mathbf{f}_n \in \mathbb{R}^{nd}$$

- We can define a discretized space tangent bundle neural network (D-TNN) as the stack of L layers:

$$\mathbf{y}_n = \sigma\left(\mathbf{h}(\Delta_n)\mathbf{f}_n\right)$$

- D-TNN can be written as a map $\mathbf{y}_n = \Psi(\mathcal{H}, \Delta_n, \mathbf{f}_n)$

- In our context, a sheaf signal $\mathbf{f}_n$ is defined as a sampled version of a tangent bundle signal $\mathbf{F}$

$$\mathbf{f}_n = \mathbf{\Omega}_n^{\mathcal{X}} \mathbf{F} \in \mathbb{R}^{nd},$$
$$\mathbf{f}_n(x_i) := [\mathbf{f}_n]_{((i-1)d+1):(i+1)d} = \mathbf{O}_i{}^T i\mathbf{F}(x_i), x_i \in \mathcal{X}$$

- We can define a discrete tangent bundle filter as

$$\mathbf{g}_n = \int_0^\infty \widetilde{h}(t) e^{t\Delta_n} \mathrm{d}t \mathbf{f}_n = \mathbf{h}(\Delta_n)\mathbf{f}_n \in \mathbb{R}^{nd}$$

- We can define a discretized space tangent bundle neural network (D-TNN) as the stack of L layers:

$$\mathbf{y}_n = \sigma\left(\mathbf{h}(\Delta_n)\mathbf{f}_n\right)$$

- D-TNN can be written as a map $\mathbf{y}_n = \Psi(\mathcal{H}, \Delta_n, \mathbf{f}_n)$

- In our context, a sheaf signal $\mathbf{f}_n$ is defined as a sampled version of a tangent bundle signal $\mathbf{F}$

$$\mathbf{f}_n = \mathbf{\Omega}_n^{\mathcal{X}} \mathbf{F} \in \mathbb{R}^{nd},$$

$$\mathbf{f}_n(x_i) := [\mathbf{f}_n]_{((i-1)d+1):(i+1)d} = \mathbf{O}_i{}^T i\mathbf{F}(x_i), x_i \in \mathcal{X}$$

- We can define a discrete tangent bundle filter as

$$\mathbf{g}_n = \int_0^\infty \widetilde{h}(t) e^{t\Delta_n} \mathrm{d}t \mathbf{f}_n = \mathbf{h}(\Delta_n)\mathbf{f}_n \in \mathbb{R}^{nd}$$

- We can define a discretized space tangent bundle neural network (D-TNN) as the stack of L layers:

$$\mathbf{y}_n = \sigma\left(\mathbf{h}(\Delta_n)\mathbf{f}_n\right)$$

- D-TNN can be written as a map $\mathbf{y}_n = \mathbf{\Psi}(\mathcal{H}, \Delta_n, \mathbf{f}_n)$

Theorem (Convergence of D-TNN to TNN)

Let $\boldsymbol{\Psi}(\mathcal{H},\cdot,\cdot)$ be the output of a neural network with L layers parameterized by the operator Δ of $\mathcal{TM}$ (TNN) or by the discrete operator Δ_n of $\mathcal{TM}_n$ (D-TNN). If:

- the frequency response of filters in $\mathcal{H}$ are non-amplifying and Lipschitz continuous
- $\mathbf{F}$ and $\boldsymbol{\Omega}_n^{\mathcal{X}}\mathbf{F}$ are bandlimited tangent bundle and sheaf signals, respectively
- The kernel scale $\epsilon = n^{-2/(d+4)}$

then it holds that:

$$\lim_{n\to\infty} ||\boldsymbol{\Psi}(\mathcal{H},\Delta_n,\mathbf{f}_n) - \boldsymbol{\Omega}_n^{\mathcal{X}}\boldsymbol{\Psi}(\mathcal{H},\Delta,\mathbf{F})||_{\mathcal{TM}_n} = 0 \text{ in probability.}$$

Discretization in the Time Domain

- Discretize function $\tilde{h}(t)$ in the continuous time domain with a fixed sampling interval T_s
- Replace the filter response function with a series of coefficients $h_k = \tilde{h}(kT_s)$, $k = 0, 1, 2 \dots$.
- Fix a finite number of K samples over the time horizon $\mathbf{h}(\Delta)\mathbf{F}(x) = \sum_{k=0}^{K-1} h_k e^{-k\Delta}\mathbf{F}(x)$
- Inject the time discretized filter on the discretized manifold:

$$\mathbf{g}_n = \mathbf{h}(\Delta_n)\mathbf{f}_n = \sum_{k=0}^{K-1} h_k e^{-k\Delta_n}\mathbf{f}_n$$

- The discretized space-time TNN (DD-TNN) is then given by (suppose multiple inputs/outputs):

$$\mathbf{Y}_n = \sigma\left(\sum_{k=1}^{K} \left(e^{\Delta_n}\right)^k \mathbf{F}_n \mathbf{H}_k\right)$$

- DD-TNN are a novel principled variant of Sheaf Neural Networks!

- Discretize function $\tilde{h}(t)$ in the continuous time domain with a fixed sampling interval T_s
- Replace the filter response function with a series of coefficients $h_k = \tilde{h}(kT_s),\ k = 0, 1, 2 \ldots.$
- Fix a finite number of K samples over the time horizon $\mathbf{h}(\Delta)\mathbf{F}(x) = \sum_{k=0}^{K-1} h_k e^{-k\Delta}\mathbf{F}(x)$
- Inject the time discretized filter on the discretized manifold:

$$\mathbf{g}_n = \mathbf{h}(\Delta_n)\mathbf{f}_n = \sum_{k=0}^{K-1} h_k e^{-k\Delta_n}\mathbf{f}_n$$

- The discretized space-time TNN (DD-TNN) is then given by (suppose multiple inputs/outputs):

$$\mathbf{Y}_n = \sigma\left(\sum_{k=1}^{K} \left(e^{\Delta_n}\right)^k \mathbf{F}_n \mathbf{H}_k\right)$$

- DD-TNN are a novel principled variant of Sheaf Neural Networks!

- Discretize function $\tilde{h}(t)$ in the continuous time domain with a fixed sampling interval T_s
- Replace the filter response function with a series of coefficients $h_k = \tilde{h}(kT_s)$, $k = 0, 1, 2 \ldots$.
- Fix a finite number of K samples over the time horizon $\mathbf{h}(\Delta)\mathbf{F}(x) = \sum_{k=0}^{K-1} h_k e^{-k\Delta}\mathbf{F}(x)$
- Inject the time discretized filter on the discretized manifold:

$$\mathbf{g}_n = \mathbf{h}(\Delta_n)\mathbf{f}_n = \sum_{k=0}^{K-1} h_k e^{-k\Delta_n}\mathbf{f}_n$$

- The discretized space-time TNN (DD-TNN) is then given by (suppose multiple inputs/outputs):

$$\mathbf{Y}_n = \sigma\left(\sum_{k=1}^{K} \left(e^{\Delta_n}\right)^k \mathbf{F}_n \mathbf{H}_k\right)$$

- DD-TNN are a novel principled variant of Sheaf Neural Networks!

- Discretize function $\tilde{h}(t)$ in the continuous time domain with a fixed sampling interval T_s
- Replace the filter response function with a series of coefficients $h_k = \tilde{h}(kT_s)$, $k = 0, 1, 2 \ldots$.
- Fix a finite number of K samples over the time horizon $\mathbf{h}(\Delta)\mathbf{F}(x) = \sum_{k=0}^{K-1} h_k e^{-k\Delta}\mathbf{F}(x)$
- Inject the time discretized filter on the discretized manifold:

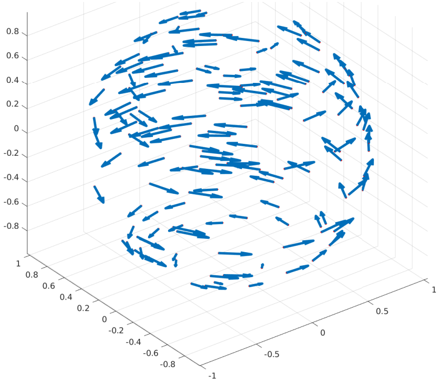
$$\mathbf{g}_n = \mathbf{h}(\Delta_n)\mathbf{f}_n = \sum_{k=0}^{K-1} h_k e^{-k\Delta_n}\mathbf{f}_n$$

- The discretized space-time TNN (DD-TNN) is then given by (suppose multiple inputs/outputs):

$$\mathbf{Y}_n = \sigma\left(\sum_{k=1}^{K} \left(e^{\Delta_n}\right)^k \mathbf{F}_n \mathbf{H}_k\right)$$

- DD-TNN are a novel principled variant of Sheaf Neural Networks!

- We assess the consistency of the proposed framework by designing a denoising task of a tangent vector field over the unit 2-sphere



- We assess the consistency of the proposed framework by designing a denoising task of a tangent vector field over the unit 2-sphere
- We uniformly sample the sphere on n points $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$
- We add AWGN with variance τ^2 obtaining a noisy field
- We compare DD-TNNs and Manifold Neural Networks (MNNs), obtained by discretizations of the Laplace-Beltrami operator (thus, without taking into account the bundle structure)

- We assess the consistency of the proposed framework by designing a denoising task of a tangent vector field over the unit 2-sphere
- We uniformly sample the sphere on n points $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$
- We add AWGN with variance τ^2 obtaining a noisy field

		$\tau = 10^{-2}$	$\tau = 5 \cdot 10^{-2}$	$\tau = 1 \cdot 10^{-1}$
$n = 200$	DD-TNN	$\mathbf{2 \cdot 10^{-4}} \pm 1.6 \cdot 10^{-5}$	$\mathbf{4.9 \cdot 10^{-3}} \pm 2.4 \cdot 10^{-4}$	$\mathbf{1.9 \cdot 10^{-2}} \pm 1.3 \cdot 10^{-3}$
	MNN	$2.9 \cdot 10^{-4} \pm 1.5 \cdot 10^{-5}$	$7 \cdot 10^{-3} \pm 2.8 \cdot 10^{-4}$	$2.9 \cdot 10^{-2} \pm 1.5 \cdot 10^{-3}$
$n = 800$	DD-TNN	$\mathbf{2 \cdot 10^{-4}} \pm 5.7 \cdot 10^{-6}$	$\mathbf{5 \cdot 10^{-3}} \pm 1.2 \cdot 10^{-4}$	$\mathbf{1.9 \cdot 10^{-2}} \pm 4.6 \cdot 10^{-4}$
	MNN	$2.8 \cdot 10^{-4} \pm 8.7 \cdot 10^{-6}$	$7.3 \cdot 10^{-3} \pm 1.7 \cdot 10^{-4}$	$2.9 \cdot 10^{-2} \pm 6.9 \cdot 10^{-4}$

Table: MSE on the denoising task

- This is the first work to introduce a signal processing framework for signals defined on tangent bundles of Riemann manifolds via the Connection Laplacian
- The presented discretization procedure and convergencence result explicitly link the manifold domain with cellular sheaves
- In future work, we will investigate more general classes of cellular sheaves that approximate unions of manifolds
- We believe our perspective on TNNs could shed further light on challenging problems in graph neural networks such as heterophily, over-squashing, or transferability

- The preprint of the journal version of this paper is online! More theory, more experiments, more FUN (?): `https://arxiv.org/abs/2303.11323`

- My Linkedin `https://www.linkedin.com/in/claudio-battiloro-b4390b175/` and Twitter `https://twitter.com/ClaBat9`: