Towards an Optimal Bound for the Interleaving Distance on Mapper Graphs

Ishika Ghosh

with Erin Wolf Chambers, Elizabeth Munch, Sarah Percival, Bei Wang

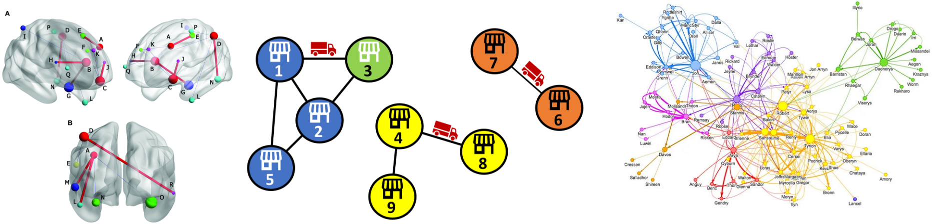
Department of Computational Mathematics, Science and Engineering
Michigan State University

5th Workshop on Computational Topology (ComPer 2025)

October 10, 2025

Graphs in the Wild

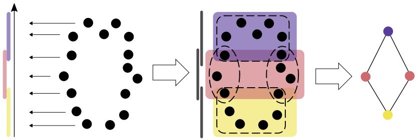
- Graphs with functions show up everywhere.
- We want to cluster and compare them.
- Need a meaningful and computable distance.
- But it's expensive to compute.

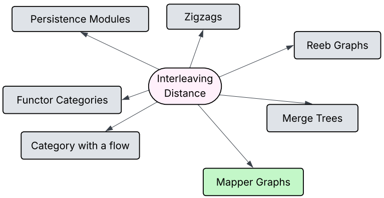


Real-world examples: brain activity, traffic flow, and social networks

Key Problem

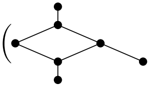
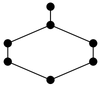
- Study graph-based topological signatures.
- Focus on Mapper graphs.
- Use interleaving distance.
- Computation is NP-hard.
- Loss function to upper bound.
- How to get the best upper bound?





TL;DR

We make Mapper graphs ML-friendly
by
optimizing a loss that bounds interleaving distance.

$d_I($  ,  $) \leq \quad n + Loss(\varphi, \psi)$

Mapper Graphs: Categorical Framework

Goal: Express mappers as cosheaves.

- Given:
  - Data $\mathbb{X}$ with function $f : \mathbb{X} \to \mathbb{R}$
  - Cover $\mathcal{U} = \{U_\alpha\}$ of $\mathbb{R}$
- Preserves connected components of $f^{-1}(U_\alpha)$
- Discretize using a grid on $\mathbb{R}$
- Represent f in cosheaf form
- Encoded as functor: $F : \mathbf{Open}(\mathcal{U}) \to \mathbf{Set}$
- Components stored in $\pi_0(f^{-1}(U))$.

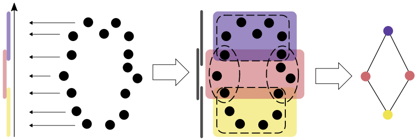
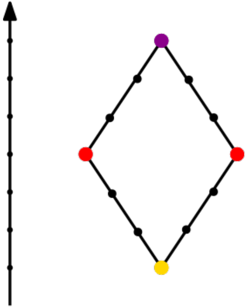


Image credit: Percival et al., 2024

Mapper Graphs: Categorical Framework

Goal: Express mappers as cosheaves.

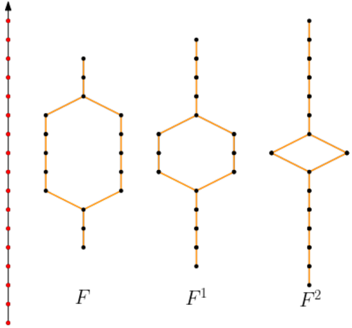
- Given:
  - Data $\mathbb{X}$ with function $f : \mathbb{X} \to \mathbb{R}$
  - Cover $\mathcal{U} = \{U_\alpha\}$ of $\mathbb{R}$
- Preserves connected components of $f^{-1}(U_\alpha)$
- Discretize using a grid on $\mathbb{R}$
- Represent f in cosheaf form
- Encoded as functor: $F : \textbf{Open}(\mathcal{U}) \to \textbf{Set}$
- Components stored in $\pi_0(f^{-1}(U))$.



Distance Between Mappers: Interleaving Distance

Goal: to compare $F, G : \mathbf{Open}(\mathcal{U}) \to \mathbf{Set}$.

- Define n-thickening of open sets.
  - Geometrically, $(i\delta, j\delta) \to ((i-n)\delta, ((j+n)\delta)$
- Thickening of functor $F^n := F \circ (-)^n$
  - Means, $F^n(S) = F(S^n)$
- n-interleaving is $\varphi : F \Rightarrow G^n$ and $\psi : G \Rightarrow F^n$.
- Must satisfy diagram commutativity.

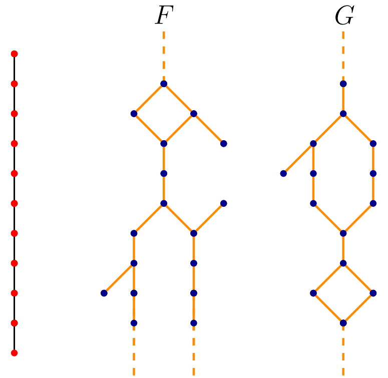


Distance Between Mappers: Interleaving Distance

- Diagrams to commute:

$$\begin{tikzcd} F(S) \arrow[rr, "{F[S\subseteq S^{2n}]}"] \arrow[dr, "\varphi_S"'] & & F(S^{2n}) \\ & G(S^n) \arrow[ur, "\psi_{S^n}"'] & \end{tikzcd} \qquad \begin{tikzcd} & F(S^n) \arrow[dr, "\varphi_{S^n}"] & \\ G(S) \arrow[ur, "\psi_S"] \arrow[rr, "{G[S\subseteq S^{2n}]}"'] & & G(S^{2n}) \end{tikzcd}$$

- Smallest n is the interleaving distance.
- NP hard.



Bounding Interleaving Distance: Assignment

- Assignment (φ, ψ): Natural transformation like maps without commutativity.
- Diagrams:
  - Triangle: Looks like interleaving
  - Parallelogram: Looks like natural transformation

$$
\begin{tikzcd}
F(S) \arrow[rr, "F[S \subseteq S^{2n}]"] \arrow[dr, "\varphi_S"'] & & F(S^{2n}) \\
& G(S^n) \arrow[ur, "\psi_{S^n}"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
& F(S^n) \arrow[dr, "\varphi_{S^n}"] & \\
G(S) \arrow[ur, "\psi_S"] \arrow[rr, "G[S \subseteq S^{2n}]"'] & & G(S^{2n})
\end{tikzcd}
$$

$$
\begin{tikzcd}
F(S) \arrow[r, "F[\subseteq]"] \arrow[dr, "\phi_S"'] & F(T) \arrow[dr, "\phi_T"] & \\
& G(S^n) \arrow[r, "G[\subseteq]"'] & G(T^n)
\end{tikzcd}
\qquad
\begin{tikzcd}
& F(S^n) \arrow[r, "F[\subseteq]"] & F(T^n) \\
G(S) \arrow[ur, "\psi_S"] \arrow[r, "G[\subseteq]"'] & G(T) \arrow[ur, "\psi_T"'] &
\end{tikzcd}
$$

Bounding Interleaving Distance: Loss Function

- Given n, how much to thicken (k), so that the diagrams commute for $n+k$?

$$\begin{tikzcd} F(S) \arrow[rr, "F[\subseteq]"] \arrow[dr, "\varphi_S"'] & & F(S^{2n}) \arrow[r] & F(S^{2(n+k)}) \\ & G(S^n) \arrow[ur, "\psi_{S^n}"'] & & \end{tikzcd}$$

$$\begin{tikzcd} F(S) \arrow[r, "F[\subseteq]"] \arrow[dr, "\varphi_S"'] & F(T) \arrow[dr, "\varphi_T"'] & \\ & G(S^n) \arrow[r, "G[\subseteq]"'] & G(T^n) \arrow[r] & G(T^{n+k}) \end{tikzcd}$$

- Diagram loss $L_{\square\!\!\!\diagdown}, L_{\square\!\!\!\diagup}, L_{\triangle}, L_{\bigtriangledown}$ is a quality measure.

- Basic loss $L_B(\varphi, \psi)$: Measures how far from commuting.

$$L_B(\phi, \psi) = \max_{\substack{\sigma<\tau\in K \\ \rho\in K}} \left\{ L_{\square\!\!\!\diagdown}^{S_\tau, S_\sigma}, L_{\square\!\!\!\diagup}^{S_\tau, S_\sigma}, L_{\square\!\!\!\diagdown}^{S_\rho, S_\rho^n}, L_{\square\!\!\!\diagup}^{S_\rho, S_\rho^n}, L_{\triangle}^{S_\rho}, L_{\bigtriangledown}^{S_\rho} \right\}$$

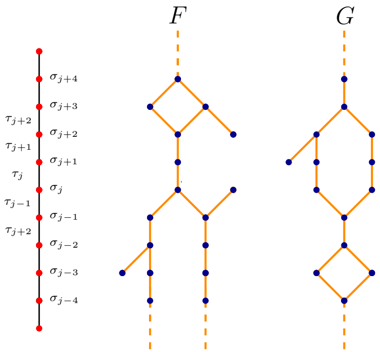
Theorem (Chambers et al. (2023))

For an n-assignment, $\varphi : F \Rightarrow G^n$ *and* $\psi : G \Rightarrow F^n$*,*

$$d_I(F, G) \leq n + L_B(\varphi, \psi).$$

Data Structure: Mapper as Graphs

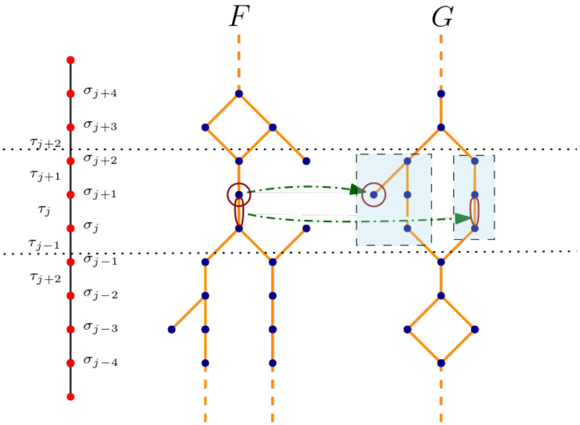
- Store $F : \textbf{Open}(\mathcal{U}) \to \textbf{Set}$ as graph $F \sim (V_F, E_F)$.
- Use grid structure on $\mathbb{R}$.
- Vertices: stored with height.
- *Assignment*: vertex and edge maps.
- Diagram commutes $\sim$ same connected component.



Commutativity and Connected Components

$$\begin{tikzcd}
F(S_\tau) \arrow[r, "F[\subseteq]"] \arrow[dr, "\varphi_{S_\tau}"'] & F(S_\sigma) \arrow[dr, "\varphi_{S_\sigma}"] & \\
& G^n(S_\tau) \arrow[r, "G[\subseteq]"'] & G^n(S_\sigma)
\end{tikzcd}$$

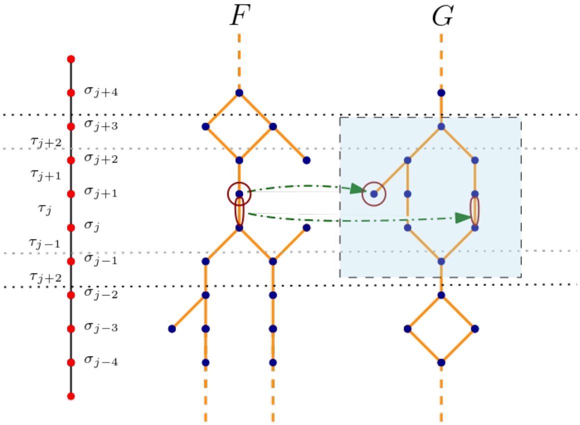
$$\begin{tikzcd}
e \arrow[r, mapsto] \arrow[dr, mapsto] & v \arrow[dr, mapsto] & \\
& {[e']} \arrow[r, mapsto] & \substack{[w] \\ {[e']}}
\end{tikzcd}$$



Commutativity and Connected Components

$$\begin{tikzcd}
F(S_\tau) \arrow[r, "F[\subseteq]"] \arrow[dr, "\varphi_{S_\tau}"'] & F(S_\sigma) \arrow[dr, "\varphi_{S_\sigma}"] & & \\
& G^n(S_\tau) \arrow[r, "G[\subseteq]"'] & G^n(S_\sigma) \arrow[r] & G(S_{\sigma_\ell}^{n+k})
\end{tikzcd}$$

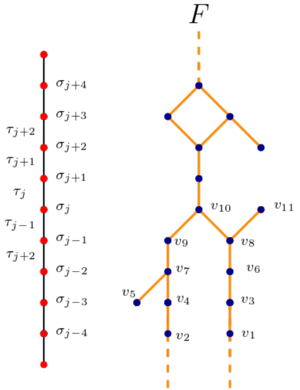
$$\begin{tikzcd}
e \arrow[r, mapsto] \arrow[dr, mapsto] & v \arrow[dr, mapsto] & & \\
& {[e']} \arrow[r, mapsto] & \begin{matrix}[w] \\ [e']\end{matrix} \arrow[r, mapsto] & \begin{matrix}[w] \\ [e']\end{matrix}
\end{tikzcd}$$



Data Structure: Maps as Matrices

Goal: Express diagram commutativity as matrix multiplication.

- Order vertices by increasing function values.
- Same for edges (lower vertex).
- Assignment: vertex and edge maps.
- Matrix whose rows and columns are these.
- Block Structure.

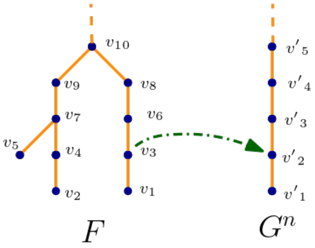


$$\{\sigma_{j-4} : (v_1, v_2), \sigma_{j-3} : (v_3, v_4), \cdots\}$$

Assignment Matrices

- Store maps like $\varphi : F \to G^n$.
- Place 1 if $\varphi(v) = v'$, 0 otherwise.
- For a valid map: only one 1 for each column.

$$
\begin{array}{cc}
 & F \\
 & \begin{array}{ccccc} v_1 & v_2 & v_3 & v_4 & \cdots \end{array} \\
G^n \begin{array}{c} v_1' \\ v_2' \\ v_3' \\ v_4' \\ \vdots \end{array} & \begin{bmatrix} & & & & \\ & & & & \\ & & 1 & & \\ & & & & \\ & & & & \end{bmatrix}
\end{array}
$$



- To have lower loss, need better assignment.

Loss Terms as Matrix Multiplication

- Diagram:

$$
\begin{tikzcd}
F(S_\tau) \arrow[r, "F[\subseteq]"] \arrow[dr, "\varphi_{S_\tau}"'] & F(S_\sigma) \arrow[dr, "\varphi_{S_\sigma}"] & \\
& G^n(S_\tau) \arrow[r, "G[\subseteq]"'] & G^n(S_\sigma)
\end{tikzcd}
\qquad
\begin{tikzcd}
e \arrow[r, mapsto] \arrow[dr, mapsto] & v \arrow[dr, mapsto] & \\
& {[e']} \arrow[r, mapsto] & \begin{smallmatrix}[w]\\ [e']\end{smallmatrix}
\end{tikzcd}
$$

- Top-right: $M_\phi^V \cdot B_F^\downarrow$
- Left-Down: $B_{G^n}^\downarrow \cdot M_\phi^E$
- In matrix terms:

$$\max_i L_{\square}^{S_{\tau_i}, S_{\sigma_i}} = \max \left\{ x \mid x \in D_{G^n}^V \left(M_\phi^V \cdot B_F^\downarrow - B_{G^n}^\downarrow \cdot M_\phi^E \right) \right\}$$

- computing loss $\rightarrow$ finding largest matrix element.

Loss in Matrix Terms

	Loss Term	Diagram	Matrix Multiplication	Eval.
Edge-vertex Parallelogram	$L^{S_\tau,S_\sigma}_{\text{▱}}$	$\begin{tikzcd} F(S_\tau) \arrow[r, "F[\subseteq]"] \arrow[dr, "\varphi_{S_\tau}"'] & F(S_\sigma) \arrow[dr, "\varphi_{S_\sigma}"] & \\ & G^n(S_\tau) \arrow[r, "G[\subseteq]"'] & G^n(S_\sigma) \end{tikzcd}$	$D^V_{G^n}\left(M^V_\varphi \cdot B^\uparrow_F - B^\uparrow_{G^n} \cdot M^E_\varphi\right)$ $D^V_{G^n}\left(M^V_\varphi \cdot B^\downarrow_F - B^\downarrow_{G^n} \cdot M^E_\varphi\right)$	$\max\limits_{x_{ij}\in A} x_{ij}$
	$L^{S_\tau,S_\sigma}_{\text{▱}}$	$\begin{tikzcd} & F^n(S_\tau) \arrow[r, "F[\subseteq]"] & F^n(S_\sigma) \\ G(S_\tau) \arrow[ur, "\psi_{S_\tau}"] \arrow[r, "G[\subseteq]"'] & G(S_\sigma) \arrow[ur, "\psi_{S_\sigma}"'] & \end{tikzcd}$	$D^V_{F^n}\left(M^V_\psi \cdot B^\uparrow_G - B^\uparrow_{F^n} \cdot M^E_\psi\right)$ $D^V_{F^n}\left(M^V_\psi \cdot B^\downarrow_G - B^\downarrow_{F^n} \cdot M^E_\psi\right)$	
Thickening Parallelogram	$L^{S_\rho,S_\rho^n}_{\text{▱}}$	$\begin{tikzcd} F(S_\rho) \arrow[r, "F[\subseteq]"] \arrow[dr, "\varphi_{S_\rho}"'] & F^n(S_\rho) \arrow[dr, "\varphi_{S_\rho^n}"] & \\ & G^n(S_\rho) \arrow[r, "G[\subseteq]"'] & G^{2n}(S_\rho) \end{tikzcd}$	$D^V_{G^n}\left(M^V_{\varphi^n} \cdot I^V_F - I^V_{G^n} \cdot M^V_\varphi\right)$ $D^E_{G^n}\left(M^E_{\varphi^n} \cdot I^E_F - I^E_{G^n} \cdot M^E_\varphi\right)$	
	$L^{S_\rho,S_\rho^n}_{\text{▱}}$	$\begin{tikzcd} & F^n(S_\rho) \arrow[r, "F[\subseteq]"] & F^{2n}(S_\rho) \\ G(S_\rho) \arrow[ur, "\psi_{S_\rho}"] \arrow[r, "G[\subseteq]"'] & G(S_\rho^n) \arrow[ur, "\psi_{S_\rho^n}"'] & \end{tikzcd}$	$D^V_{F^n}\left(M^V_{\psi^n} \cdot I^V_G - I^V_{F^n} \cdot M^V_\psi\right)$ $D^E_{F^n}\left(M^E_{\psi^n} \cdot I^E_G - I^E_{F^n} \cdot M^E_\psi\right)$	
Triangle	$L^{S_\rho}_{\triangledown}$	$\begin{tikzcd} F(S_\rho) \arrow[rr, "F[\subseteq]"] \arrow[dr, "\varphi_{S_\rho}"'] & & F^{2n}(S_\rho) \\ & G^n(S_\rho) \arrow[ur, "\psi_{S_\rho^n}"'] & \end{tikzcd}$	$D^V_{F^{2n}}\left(I^V_{F^n} \cdot I^V_F - M^V_{\psi^n} \cdot M^V_\varphi\right)$ $D^E_{F^{2n}}\left(I^E_{F^n} \cdot I^E_F - M^E_{\psi^n} \cdot M^E_\varphi\right)$	$\max\limits_{x_{ij}\in A} \left\lceil \frac{x_{ij}}{2} \right\rceil$
	$L^{S_\rho}_{\triangle}$	$\begin{tikzcd} & F^n(S_\rho) \arrow[dr, "\varphi_{S_\rho^n}"] & \\ G(S_\rho) \arrow[ur, "\psi_{S_\rho}"] \arrow[rr, "G[\subseteq]"'] & & G^{2n}(S_\rho) \end{tikzcd}$	$D^V_{G^{2n}}\left(I^V_{E^n} \cdot I^V_G - M^V_{\varphi^n} \cdot M^V_\psi\right)$ $D^E_{G^{2n}}\left(I^E_{G^n} \cdot I^E_G - M^E_{\varphi^n} \cdot M^E_\psi\right)$	

Implement Loss Optimization

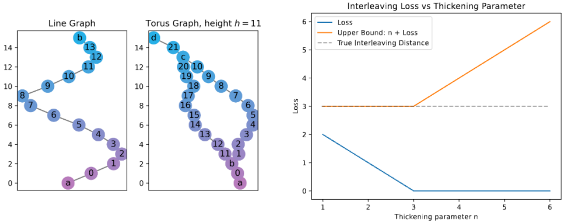
- Question: Can we formulate loss optimization as a linear program?
  - Yes! Both the objective function and constraints can be linearized.
- Discretized setup calls for integer linear programming.
- ILP is formulated as follows:

$$
\begin{aligned}
\text{Minimize} \quad & \ell \\
\text{Subject to} \quad & \ell \geq x_{ij}^{F,\uparrow} \quad \forall\, x_{ij}^{F,\uparrow} \in D_{G^n}^{V} \left(M_{\varphi}^{V} \cdot B_{F}^{\uparrow} - B_{G^n}^{\uparrow} \cdot M_{\varphi}^{E} \right) \\
& \sum_i x_{ij}^{\eta,A} = 1 \quad \forall x_{ij}^{\eta,A} \in M_{\eta}^{A},\ \eta \in \{\phi, \phi^n, \psi, \psi^n\},\ A \in \{V, E\} \\
& x_{ij} \geq 0 \\
& \vdots \quad \text{(more constraints)}
\end{aligned}
$$

- Nonlinearity in triangles (e.g., $M_{\psi^n}^{V} \cdot M_{\varphi}^{V}$) is linearized with additional variables.

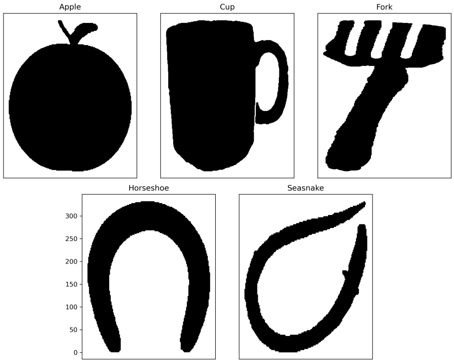
Experiments with Small Mappers

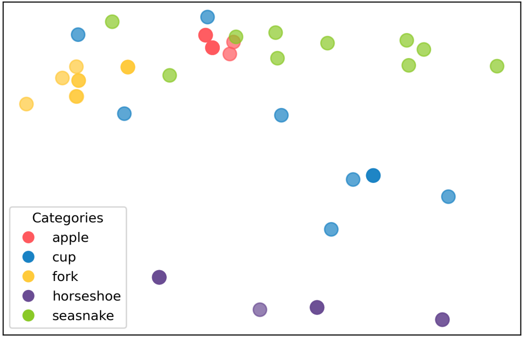
- Small mappers where interleaving distance can be computed.
- Line mapper: One vertex at every height.
- Torus mapper: A loop in the middle.
- Interleaving distance is $\lceil \frac{h}{4} \rceil$.



Experiments with Images

- Mappers of images from the MPEG7 image dataset.
- Pairwise distance with the optimized mapper loss.
- MDS to preserve pairwise dissimilarities.

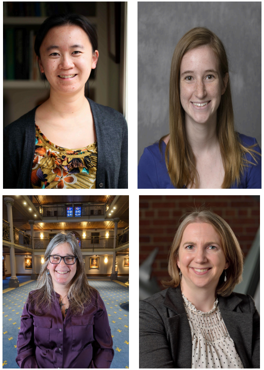




Summary and Future Work

- First available method to compute a bound for the interleaving distance.
- Mapper as graphs, maps as matrices.
- Loss optimization using ILP.
- Initial experiments support the approach.
- Future: test on more datasets, improve efficiency.
- Explore mapper construction and parameter selection.
- Integrate into ML pipelines.

Thank You!



Coauthors

MunchLab, Spring 2025

email: ghoshis3@msu.edu

Find the preprint: