Bounding the Interleaving Distance for Mapper Graphs with a Loss Function *

Erin Chambers[1], Elizabeth Munch[2], Sarah Percival[2], and Bei Wang[3]

[1]St. Louis University
[2]Michigan State University
[3]University of Utah

Abstract

Data consisting of a graph with a function mapping into $\mathbb{R}^d$ arise in many data applications, encompassing structures such as Reeb graphs, geometric graphs, and knot embeddings. As such, the ability to compare and cluster such objects is required in a data analysis pipeline, leading to a need for distances between them. In this work, we study the interleaving distance on discretization of these objects, $\mathbb{R}^d$-mapper graphs, where functor representations of the data can be compared by finding pairs of natural transformations between them. However, in many cases, computation of the interleaving distance is NP-hard. For this reason, we take inspiration from recent work by Robinson to find quality measures for families of maps that do not rise to the level of a natural transformation, called assignments. We then endow the functor images with the extra structure of a metric space and define a loss function which measures how far an assignment is from making the required diagrams of an interleaving commute. Finally we show that the computation of the loss function is polynomial with a given assignment. We believe this idea is both powerful and translatable, with the potential to provide approximations and bounds on interleavings in a broad array of contexts.

1 Introduction

Graphs with additional geometric information arise in many contexts in data analysis. For instance, a *geometric graph* is generally defined as an abstract graph along with a well-behaved embedding of the graph into $\mathbb{R}^2$, while a graph with a well-behaved map into $\mathbb{R}$ is called a *Reeb graph*. In particular from the viewpoint of the Reeb graph, these types of input data can arise by studying connected component structures from more general input $\mathbb{R}^d$-spaces, meaning a topological space $\mathbb{X}$ with a function $f : \mathbb{X} \to \mathbb{R}^d$. Such graphs are a fundamental object used to model a wide range of data sets, ranging from maps and trajectories, to commodity networks (e.g. electrical grids) and shape skeletons for object recognition. The input data can be quite noisy, so the ability to compare, cluster, and simplify such representative objects is essential in a data analysis pipeline, leading to a need for theoretically motivated and computable distances. In this paper, we study a distance for a discretization of the input data, that is, an $\mathbb{R}^d$-*mapper graph* [34]. That is, starting from a topological space $\mathbb{X}$ with a function $f : \mathbb{X} \to \mathbb{R}^d$ (resp. a point cloud P with a function $f : P \to \mathbb{R}^d$), the mapper graph is an encoding of the connected components (resp. clusters) of $f^{-1}(U_\alpha)$ for some cover $\mathcal{U} = \{U_\alpha\}$ of $\mathbb{R}^d$.

There has been extensive work on metrics for general graphs, geometric graphs, and Reeb graphs (see surveys [13, 16, 17, 35], [9], and [4, 36] respectively). In this paper, we will draw inspiration from the interleaving distance; specifically, we develop a natural extension of the interleaving distance on Reeb graphs [15]

*The authors would like to thank Vidit Nanda for helpful discussions related to non-commutative diagrams. This work was funded in part by the National Science Foundation through grants CCF-1907591, CCF-1907612, CCF-2106578, CCF-2142713, CCF-2106672, DMS-2301361, and IIS-2145499.

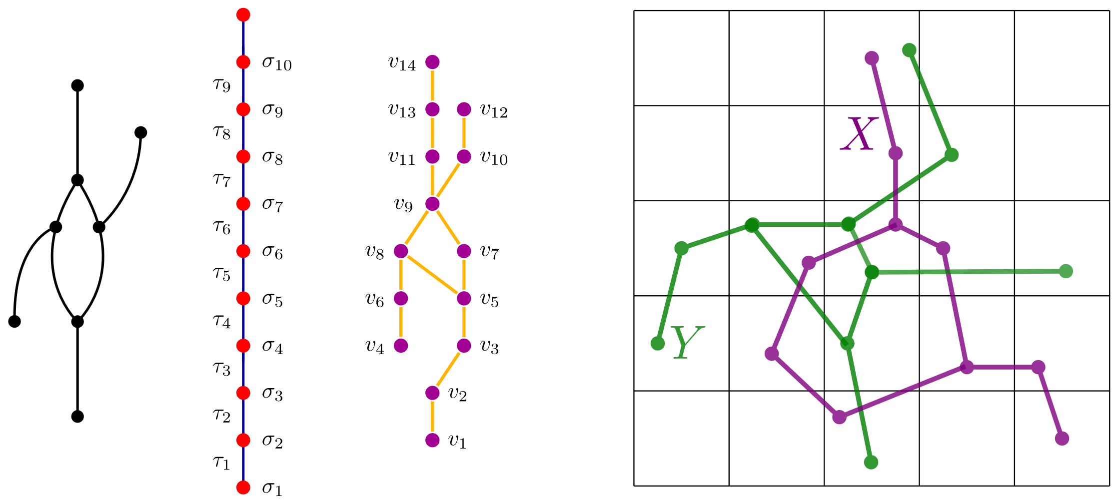


Figure 1: (Left) An example of a Reeb graph ($d = 1$) where the discretization of $\mathbb{R}$ is used to generate a mapper graph. (Right) An example of a pair of geometric graphs ($d = 2$) which we study using the connected components restricted to the grid.

to the setting of $\mathbb{R}^d$-mapper graphs. Interleaving distances arose in the context of generalizing the bottleneck distance for persistence modules [12] and were subsequently translated to more general categorical frameworks in [8, 15]. With the exception of 1-parameter persistence [24], the interleaving distance is NP-hard in many contexts including multi-parameter persistence [2, 3], and Reeb graphs [15]. However, some additional structural information can give better algorithms such as FPT computation for merge trees [18], and polynomial time for formigrams [22] and labeled merge trees [19, 27]. See [2] for a recent summary of interleaving distance complexity results.

When $d = 1$, there is already work using the interleaving distance to relate the Reeb graph and its mapper graph [5, 6, 10, 11, 28]. We will encode the structure of our more general $\mathbb{R}^d$-mapper graphs in a discretized setting by imposing a grid structure K on $\mathbb{R}^d$. Then we can represent the input data $f : X \to \mathbb{R}^d$ as a cosheaf of the form $F : \mathbf{Open}(K) \to \mathbf{Set}$ where we store the connected components of inverse images of open sets $\pi_0(f^{-1}(U))$. The idea of the interleaving distance, in this context, is to compare two cosheaves $F, G : \mathbf{Open}(K) \to \mathbf{Set}$ using a pair of natural transformations $\varphi : F \Rightarrow G^n$ and $\psi : G \Rightarrow F^n$ mapping into relaxations of the original inputs. The complexity of computing this distance then relies on finding the smallest n with available φ and ψ maps, which in our setting immediately connects to hard underlying problems such as graph isomorphism.

The ideas building this distance are rooted in previous work that study interleavings in related contexts. In some sense, we can view this distance as a discretized cosheaf version of the continuous sheaf version of the interleaving distance that was previously studied [14]. It can fit into the more general framework of an interleaving distance arising from a category with a flow [33], or as an interleaving distance on generalized persistence modules with a family of translations [8], but prior work in these areas focused on theoretical properties and did not address computational aspects, as the more general framework makes such study incredibly difficult. Perhaps the closest example of this distance is mentioned in the $d = 1$ case as an example in prior work [5]; however, that example keeps the thickening of the open sets structure tightly bound to thickening in $\mathbb{R}^d$, whereas we choose to define the distance entirely over the combinatorial structures.

While all this prior work is powerful in theory, the computational complexity of the construction in more general settings has meant a lack of the use of the interleaving distance in practice. To circumvent these issues, we take inspiration from recent work of Robinson [32] to define quality measures for families of maps that do not rise to the level of a natural transformation, in order to allow for non-optimal maps φ and ψ in this framework. We then apply these quality measures to $\mathbb{R}^d$-mapper graphs, in the hopes of utilizing algorithms from geometry and graph theory to make computation more feasible.

In particular, in [32] the object of study is a single input assignment of data of the form $P : \mathbf{Open}(X) \to \mathbf{Set}$ and, with the added structure of a pseudometric for each set $P(U)$, provides a measurement for how far the input data is from having a global section. In our work, we instead work with a pair of functors $F, G : \mathbf{Open}(K) \to \mathbf{Set}$ as input, and study collections of maps $\varphi = \{\varphi_U : F(U) \to G(U^n) \mid U\}$ and $\psi = \{\psi_U : G(U) \to F(U^n) \mid U\}$ which we call an *assignment* when they do not necessarily form a true interleaving. We then endow the image with the extra structure of a metric space, so that we have pairs $(F(U), d_U)$ for every open set U. Using this metric structure, we define a loss function $L(\varphi, \psi)$ which measures how far the required diagrams of an interleaving are from commuting, given any input assignment (Thm. 3.8). We modify this bound by only focusing on the loss function computed for a basis of the topology, $L_B(\varphi, \psi)$ (Thm. 3.17), which not only improves the computational complexity but also improves the bound. Then, we show that the computation of the bound is polynomial, opening up the potential for algorithmic approximation of the interleaving distance. Throughout, we show examples encoding the data of a *geometric graph* (i.e. a graph G with a straight line embedding $f : G \to \mathbb{R}^2$) or a Reeb graph (a graph G with a straight line map to $\mathbb{R}$), however, this kind of input is not a requirement for our framework; see Fig. 1 for an illustration.

We note that this paper is the first step in a larger project. That is, the paper here presents a loss function that can be computed given an input n-assignment φ, ψ and results in a bound on the distance by explicitly constructing an $(n + L_B(\varphi, \psi))$-interleaving. As with many garbage-in-garbage-out settings, this bound is only as good as the input, but this study seeks to determine the most general possible bound with no guarantees on the input at all. In future work, we plan to include this loss function as part of an optimization strategy to update a given assignment in order to find better bounds as well as provide further study on how close to optimal is possible. We suspect that tools from computational geometry may provide particularly nice starting maps and strategies to improve our assignment.

Outline. In Sec. 2 we provide the necessary technical background to set up the interleaving distance for $\mathbb{R}^d$-mapper graph inputs. In Sec. 3, we define the loss functions and bounds. We discuss algorithmic requirements of the bound in Sec. 4. Next, we show how this loss function can be used to similarly bound the Reeb graph interleaving distance by approximating the Reeb graph with a mapper graph in Sec. 5. We include all technical proofs in Sec. 6. Finally, we discuss broader implications of this work in Sec. 7.

2 Technical Background

We will assume several example types of inputs in this paper. All are tied together by having data of the form $f : G \to \mathbb{R}^d$, where G is a finite graph. We view each input as a topological graph by treating the graph as a 1-dimensional CW complex. We call this input an $\mathbb{R}^d$-graph. Prior to discussing the specifics of $\mathbb{R}^d$-graphs in our use cases, we give the necessary categorical framework in order to formulate a precise notion of the interleaving distance.

2.1 Functors and Cosheaves

We give basic definitions for the category theoretic notions required in this paper, and direct the interested reader to [14, 30] for further details. A *category* $\mathcal{C}$ consists of a collection of objects $X, Y, Z, \ldots$ and morphisms $f, g, h, \ldots$ with the following requirement: morphisms $f : X \to Y$ have designated domain X and codomain Y; every object has a designated identity morphism $\mathbb{1}_X : X \to X$, and any pair of morphisms $f : X \to Y$ and $g : Y \to Z$ has a composite morphism $g \circ f : X \to Z$. These objects and morphisms satisfy an identity axiom, where $f : X \to Y$ is the same as the $\mathbb{1}_Y \circ f$ and $f \circ \mathbb{1}_X$; and composition (denoted by $\circ$ but often dropped when unnecessary) is associative, so $h \circ (g \circ f) = (h \circ g) \circ f$. Some example categories are **Set** where objects are finite sets and morphisms are set maps; **Top** where objects are topological spaces and morphisms are continuous functions; and $\mathbf{Open}(K)$ for a given topological space K, where the objects are open sets and morphisms are given by inclusion.

A *functor* $F : \mathcal{C} \to \mathcal{D}$ is a map between categories preserving the relevant structures. Specifically, for every object $X \in \mathcal{C}$ there is a an object $F(X) \in \mathcal{D}$, and for every morphism $f : X \to Y$, there is a morphism

$F[f] : F(X) \to F(Y)$. To be a functor, F must further satisfy that for any $X \in \mathcal{C}$, $F[\mathbb{1}_X] = \mathbb{1}_{F(X)}$ and for any composable pair $f, g \in \mathcal{C}$, we have $F[gf] = F[g]F[f]$. Given functors $F, G : \mathcal{C} \to \mathcal{D}$, a *natural transformation* $\eta : F \Rightarrow G$ consists of a map $\eta_X : F(X) \to G(X)$ for every $X \in \mathcal{C}$ (called the components) so that for any morphism $f : X \to Y$ in $\mathcal{C}$, the following diagram

$$\begin{tikzcd}
X \arrow[d, "f"] & F(X) \arrow[r, "\eta_X"] \arrow[d, "F[f]"'] & G(X) \arrow[d, "G[f]"] \\
Y & F(Y) \arrow[r, "\eta_Y"] & G(Y)
\end{tikzcd}$$

commutes. One example is $\pi_0 : \mathbf{Top} \to \mathbf{Set}$, where $\pi_0(\mathbb{X})$ is the set of path-connected components of $\mathbb{X}$, and morphisms are set maps $\pi_0[f] : \pi_0(\mathbb{X}) \to \pi_0(\mathbb{Y})$ sending a connected component A in $\mathbb{X}$ to the connected component of $f(A)$ in $\mathbb{Y}$.

A diagram is a functor $F : J \to \mathcal{C}$ where J is a small category. In essence, this construction picks out a collection of objects $F(j)$ and a collection of morphisms $F(j) \to F(k)$. A cocone on a given diagram is a natural transformation $\lambda : F \to c$ where we abuse notation to write that $c : J \to \mathcal{C}$ is the constant functor returning the object $c(j) = c \in \mathcal{C}$ for all $j \in J$. We often call the components $\lambda : F(j) \to c$ the *legs*, and note that this requirement says that for any $f : j \to k$ in J, $\lambda_k \circ F[f] = \lambda_j$. A cocone $\lambda : F \to c$ is called a colimit if for any other cocone $\lambda' : F \to c'$, there is a unique morphism $u : c \to c'$ such that

$$\begin{tikzcd}
F(j) \arrow[rr, "F[f]"] \arrow[dr, "\lambda_j"] \arrow[ddr, "\lambda'_j"'] & & F(k) \arrow[dl, "\lambda_k"'] \arrow[ddl, "\lambda'_k"] \\
& c \arrow[d, dashed, "u"] & \\
& c' &
\end{tikzcd}$$

commutes for all $f : j \to k$ in J.

We will be particularly interested in functors of the form $F : \mathbf{Open}(X) \to \mathbf{Set}$, which can also be called *pre-cosheaves*. A pre-cosheaf is a *cosheaf* if it satisfies a gluing axiom, meaning that $F(U)$ is entirely determined by $F(U_\alpha)$ for any cover $\{U_\alpha\}_\alpha$. Specifically, given an open set U and a cover $\{U_\alpha \mid \alpha \in A\}$ of U, define a category $\mathcal{U} = \{U_\alpha \cap U_{\alpha'} \mid \alpha, \alpha' \in A\}$ with morphisms given by inclusion. Then we have a diagram $F : \mathcal{U} \to \mathbf{Set}$, and as such can consider its colimit $\lambda : F \to L$. If the unique map $L \to F(U)$ given by the colimit definition is an isomorphism, then F is called a *cosheaf*.

2.2 Functorial Representation of Embedded Data

We start by defining a cubical complex on $[-B, B]^d$, where $[-B, B]^d$ will be the bounding box for the image of the graphs $f : X \to \mathbb{R}^d$ and $g : Y \to \mathbb{R}^d$ to be compared. Following [21], we define a cubical complex given by diameter δ, with a modification to result in open sets. For the sake of simplicity, assume that B is a multiple of δ, so that the bounding box can be written as $[-L\delta, L\delta]^d$. An *elementary interval* is an open interval in $\mathbb{R}$ of the form $(\ell\delta, (\ell + 1)\delta)$ or a single point viewed as a degenerate interval $[\ell] := [\ell\delta, \ell\delta]$ for $\ell \in [-L, \cdots, L] \subset \mathbb{Z}$. These are called non-degenerate and degenerate intervals, respectively. An elementary cube Q is a finite product of d elementary intervals, i.e. $\sigma = I_1 \times I_2 \times \cdots \times I_d \subset [-B, B]^d$. The dimension of a cube σ is given by the number of intervals used which are non-degenerate. This means that 0-cubes are vertices at grid locations $(i\delta, j\delta, \ldots, k\delta) \in \delta \cdot \mathbb{Z}^d$, 1-cubes are edges (not including their endpoints), 2-cubes are squares (not including their boundaries), 3-cubes are voxels, etc. The collection of elementary cubes discretizing $[-B, B]^d$ is a finite cubical complex K. This construction comes with a face relation which gives a poset structure, where we write $\sigma \leq \tau$ iff $\sigma \subseteq \overline{\tau}$, where $\overline{\tau}$ denotes the closure of the set. In order to differentiate between the combintorial and continuous settings, we write $|\sigma|$ for the geometric realization in $\mathbb{R}^d$ of a combinatorial object $\sigma \in K$.

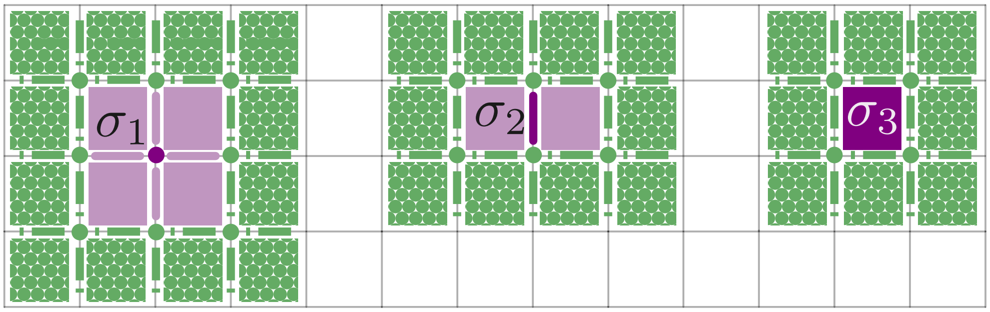


Figure 2: Examples of open sets for $d = 2$. The basic open (upset) is given in light purple for each cell type shown in dark purple (vertex σ_1, edge σ_2 and square σ_3). The 1-thickening of the open sets are given by including the green portion.

We next endow the complex K with the Alexandroff topology, following [1]. Given the poset $(K, \leq)$, for any set $S \subseteq X$, the up-set is $S^{\uparrow} = \{x \mid x \geq y,\, y \in S\}$ and the down-set is $S^{\downarrow} = \{x \mid x \leq y,\, y \in S\}$. We give K the Alexandroff topology[1] $\mathbf{Open}(K)$, where a set $U \subseteq K$ is open iff the following holds: for any $x \in U$ and any $y \geq x$, $y \in U$. Equivalently, this means that U is its own up-set, i.e. $U = U^{\uparrow}$. It can be checked that this topology has a basis given by the collection $\{U_\sigma\}_{\sigma \in K}$ where $U_\sigma := \{\sigma\}^{\uparrow} = \{\tau \mid \tau \geq \sigma\}$. This complex is constructed so that for any open set $U \subset K$, the geometric realization $|U| \subseteq \mathbb{R}^d$ is open in the usual sense.

The example inputs here are given by $f : G \to \mathbb{R}^d$ where G is a finite topological graph; however the graph assumption is needed only for simplicity of drawing examples. For our purpose, we only require that for any open set U, the set of connected components $\pi_0 f^{-1}(|U|)$ is finite. Then we can encode f in a functor $F : \mathbf{Open}(K) \to \mathbf{Set}$ given by

$$\begin{array}{rccc} F: & \mathbf{Open}(K) & \to & \mathbf{Set} \\ & U & \mapsto & \pi_0 f^{-1}(|U|). \end{array}$$

Functoriality of π_0 means that for $U \subseteq V$, there is an induced map

$$F[U \subseteq V] \colon \pi_0 f^{-1}(|U|) \to \pi_0 f^{-1}(|V|),$$

so that F satisfies the requirements of a functor. Indeed, this functor is actually a cosheaf. When the sets involves are obvious in the notation, we will write the induced map as $F[\subseteq] : F(U) \to F(V)$.

2.3 Thickenings

Given any set $U \in \mathbf{Open}(K)$, the 1-thickening is defined by taking the upset of the downset of U, written as $U^1 = (U^{\downarrow})^{\uparrow}$. This operation can be thought of as taking the star of the closure of the set; see Fig. 2 for examples. The n-thickening is defined to be n repetitions of the process given recursively as

$$U^n = \begin{cases} U & n = 0 \\ (U^{n-1})^{\downarrow\uparrow} & n \geq 1. \end{cases}$$

Each U^n is itself an open set in $\mathbf{Open}(K)$, and that if $U \subseteq V$, then $U^n \subseteq V^n$. Thus we can view this operation as a functor on the category $\mathbf{Open}(K)$ with morphisms given by inclusion:

$$\begin{array}{rccc} (-)^n: & \mathbf{Open}(K) & \to & \mathbf{Open}(K) \\ & U & \mapsto & U^n. \end{array}$$

[1]In the case of finite posets, either down- or up-sets can be used to define the topology; while the Alexandroff topology is defined using the down-sets as opens in the textbook we follow [1]. However, we choose to use the upsets as opens since we are trying to avoid using the opposite poset as much as possible to alleviate notation woes, and the correspondence with stars in this setting is useful for our purpose.

In Sec. 6, we show that $(-)^n$ is a functor. Another useful property of this thickening, proved in Sec. 6, is described in Lem. 2.1.

Lemma 2.1. *For any $n, n' \geq 0$ and $U \in \mathbf{Open}(K)$,*

$$(U^n)^{n'} = U^{n+n'}.$$

We can use this thickening to build an interleaving distance on functors of the form $F : \mathbf{Open}(K) \to \mathbf{Set}$. The first necessary ingredient is the composition of functors $F \circ (-)^n : \mathbf{Open}(K) \to \mathbf{Set}$, which we denote by F^n. This means $F^n(U) = F(U^n)$, followed by a similar setup for G^n. With this notation, an interleaving is a pair of natural transformations $\varphi : F \Rightarrow G^n$ and $\psi : G \Rightarrow F^n$, so a component of φ is a set-map $\varphi_U : F(U) \to G(U^n)$. There is another component at U^n, $\varphi_{U^n} : F(U^n) \to G(U^{2n})$, which can also be viewed as a component of a different natural transformation $\varphi^n : F^n \Rightarrow G^{2n}$. For this reason, we use the notation φ_{U^n} and φ^n_U interchangeably when φ is indeed a natural transformation.[2]

We are now ready to introduce our notion of interleaving distance.

Definition 2.2. *Given cosheaves $F, G : \mathbf{Open}(K) \to \mathbf{Set}$ and $n \geq 0$, an n-*interleaving *is a pair of natural transformations $\varphi : F \Rightarrow G^n$ and $\psi : G \Rightarrow F^n$ such that the diagrams*

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

commute for all $U \in \mathbf{Open}(K)$. The interleaving distance is given by

$$d_I(F, G) = \inf\{n \geq 0 \mid \textit{there exists an } n\textit{-interleaving}\},$$

and is set to be $d(F, G) = \infty$ if there is no interleaving for any n.

As shown in Sec. 6, this definition fits in the framework built by Bubenik et al. [8] and thus it is an extended pseudometric.

3 Loss Function and Bounds

In this section, we introduce a loss function for interleavings on $\mathbb{R}^d$-mapper graphs. We give the definition of the loss function (Defn. 3.4) in Sec. 3.1, and present our first version of the bound as Thm. 3.8 in Sec. 3.2. However, this version of the bound requires checking diagrams for all possible open sets $U \in \mathbf{Open}(K)$ which creates a combinatorial explosion that is counterproductive in practice. Thus, in Sec. 3.3, we prove this loss function can be replaced with an improved loss function which only needs to check the open sets for a basis of $\mathbf{Open}(K)$.

3.1 Loss Function Definition

We start by turning each $F(U)$ (similarly $G(U)$) into a metric space, as follows.

Definition 3.1. *Define the distance $d^F_U(A, B)$ for $A, B \in F(U)$ to be the smallest n such that A and B represent the same connected component when included into U^n. Specifically,*

$$d^F_U(A, B) = \min\{n \geq 0 \mid F[U \subset U^n](A) = F[U \subset U^n](B)\}.$$

If no such n exists, then we set $d^F_U(A, B) = \infty$.

[2] We are implicitly using Lem. 2.1 to write the maps this way.

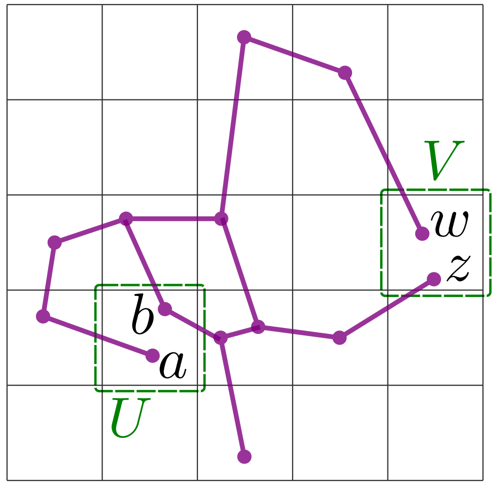


Figure 3: An example geometric graph used to calculate example distances in d_U^F and d_V^F.

Consider the example of Fig. 3 with a single input graph encoded by a cosheaf $F : \mathbf{Open}(K) \to \mathbf{Set}$. The set $F(U)$ has two elements, which we denote by A and B as they represent the connected components containing the points a and b respectively. Then $d_U^F(A, B) = 1$, since thickening the set U by 1 puts a and b in the same connected component. Likewise, denoting the elements of $F(V)$ by W and Z, we see that $d_V^F(W, Z) = 2$ since we must expand the set V twice before w and z are in the same connected component.

As a first useful property of this distance, thickening a set implies that the distance between components will only decrease. For an example, consider $W, Z \in F(V)$ representing points w and z in Fig. 3. As noted previously, $d_V^F(W, Z) = 2$. However, if the elements $W', Z' \in F(V^1)$ represent the connected components in the 1-thickening of V, then $d_{V^1}^F(W', Z') = 1$, and in particular, $d_V^F(W, Z) \geq d_{V^1}^F(W', Z')$. This idea is formalized in the following lemma:

Lemma 3.2. *Fix $k \geq 0$ and any $A, B \in F(U)$ with images $A' = F[U \subseteq U^k](A)$ and $B' = F[U \subseteq U^k](B)$ in $F(U^k)$. Then $d_U^F(A, B) \geq d_{U^k}^F(A', B')$.*

Proof. Let $n = d_U^F(A, B)$, so that we know the image of A and B in $F(U^n)$ is the same. If $k \geq n$, then we use the functor maps $F(U) \to F(U^n) \to F(U^k)$ to see that the images of A and B are the same in $F(U^n)$ so they are the same in $F(U^k)$. Then $d_{U^k}^F(A', B') = 0 \leq d_U^F(A, B)$. If $k < n$, then we have the maps $F(U) \to F(U^k) \to F(U^n)$. This means that $d_{U^k}^F(A', B') \leq n - k \leq n = d_U^F(A, B)$, completing the proof. ☐

We use this framework as follows: first, assume we are given F and G but our attempts at finding an interleaving do not necessarily satisfy the requirements of a natural transformation. Normally, a natural transformation $\eta : H \Rightarrow H'$ is a collection of component morphisms $\eta : H(U) \to H'(U)$ which commute with the inclusions:

$$\begin{tikzcd}
H(U) \arrow[r, "H'[\subseteq]"] \arrow[d, "\eta_u"'] & H(V) \arrow[d, "\eta_V"] \\
H'(U) \arrow[r, "H[\subseteq]"] & H'(V).
\end{tikzcd}$$

The following definitions, inspired by [32] and [29], give names to collections of component morphisms used to define an interleaving where the square might not commute.

Definition 3.3. *Given functors $H, H' : \mathbf{Open}(K) \to Set$, an* unnatural transformation[3] $\eta : H \to H'$ *is a collection of maps $\eta_U : H(U) \to H'(U)$ with no additional promise of commutativity.*

For a fixed $n \geq 0$ and cosheaves F and G, an assignment*, or more specifically an n-*assignment*, is a pair of unnatural transformations $\varphi : F \Rightarrow G^n$ and $\psi : G \Rightarrow F^n$.*

In order to clarify notation, for the remainder of the paper, we will be using n-assignments to build $(n+k)$-interleavings, which by definition will be required to be natural transformations. When the n-assignment might not commute, we denote its maps by lower case φ and ψ; for $(n+k)$-assignments which are constructed to be natural transformations, we denote them by Φ and Ψ.

In the spirit of [32], we measure the quality of a choice of an n-assignment φ, ψ using the collections of distances $\{d_U^F \mid U \in \mathbf{Open}(K)\}$ and $\{d_U^G \mid U \in \mathbf{Open}(K)\}$. First, note that checking that φ and ψ are natural transformations means ensuring the diagrams

$$\begin{tikzcd}
F(U) \arrow[r, "F[\subseteq]"] \arrow[dr, "\varphi_U"'] & F(V) \arrow[dr, "\varphi_{U^n}"'] & \\
& G(U^n) \arrow[r, "G[\subseteq]"'] & G(V^n)
\end{tikzcd}
\qquad
\begin{tikzcd}
& F(U^n) \arrow[r, "F[\subseteq]"] & F(V^n) \\
G(U) \arrow[ur, "\psi_U"] \arrow[r, "G[\subseteq]"'] & G(V) \arrow[ur, "\psi_{U^n}"] &
\end{tikzcd}$$

commute. As we use them repeatedly, we will denote these diagrams by $\boxbslash_\varphi(U,V)$ and $\boxslash_\psi(U,V)$, dropping the subscript when it is clear from context. Then checking whether the pair constitutes an interleaving involves checking commutativity of the diagrams

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

which we denote by $\triangledown_{\varphi,\psi}(U)$ and $\triangle_{\varphi,\psi}(U)$ respectively, again dropping the subscripts when unnecessary. We measure quality of the given assignments by checking how far these four diagrams are from commuting in the sense of the distances defined at the terminal vertex of the shape.

Definition 3.4. *Fix an n-assignment (φ, ψ). We define four* diagram loss functions*:*

$$\begin{aligned}
L_{\boxbslash}^{U,V}(\varphi) &= \max_{\alpha \in F(U)} d_{V^n}^G(\varphi_U^n \circ F[U \subseteq V](\alpha), G[U^n \subseteq V^n] \circ \varphi_U(\alpha)) \\
L_{\boxslash}^{U,V}(\psi) &= \max_{\alpha \in G(U)} d_{V^n}^F(\psi_U^n \circ G[U \subseteq V](\alpha), F[U^n \subseteq V^n] \circ \psi_U(\alpha)) \\
L_{\triangledown}^{U}(\varphi, \psi) &= \max_{\alpha \in F(U)} \left\lceil \tfrac{1}{2} \cdot d_{U^{2n}}^F(F[U \subseteq U^{2n}](\alpha), \psi_{U^n} \circ \varphi_U(\alpha)) \right\rceil \\
L_{\triangle}^{U}(\varphi, \psi) &= \max_{\alpha \in G(U)} \left\lceil \tfrac{1}{2} \cdot d_{U^{2n}}^G(G[U \subseteq U^{2n}](\alpha), \varphi_{U^n} \circ \psi_U(\alpha)) \right\rceil.
\end{aligned}$$

Then the loss for the given assignment is defined to be

$$L(\varphi, \psi) = \max_{U \subseteq V} \left\{ L_{\boxbslash}^{U,V}, L_{\boxslash}^{U,V}, L_{\triangle}^{U}, L_{\triangledown}^{U} \right\}.$$

These loss functions are defined in a way so that while the diagram in question might not commute, pushing n forward by the loss value will send the elements to the same place. For example, if $L_{\boxbslash}^{U,V}(\varphi) = k$,

[3] A natural transformation is an unnatural transformation which just happens to follow commutativity properties. In other words, natural and unnatural transformations are not mutually exclusive. This vocabulary follows from [29] so please do not blame us for the linguistic implications.

then in the diagram

$$
\begin{tikzcd}
F(U) \arrow[r, "F[\subseteq]"] \arrow[dr, "\varphi_U"'] & F(V) \arrow[dr, "\varphi_{U^n}"'] & & \\
& G(U^n) \arrow[r, "G[\subseteq]"'] & G(V^n) \arrow[r] & G(V^{n+k})
\end{tikzcd}
\tag{3.5}
$$

the image of a point from $F(U)$ is the same in $G(V^{n+k})$ following both paths. Similarly, if $L^U_{\bigtriangledown}(\varphi,\psi) = k$, then in the diagram

$$
\begin{tikzcd}
F(U) \arrow[rr, "F[\subseteq]"] \arrow[dr, "\varphi_U"'] & & F(U^{2n}) \arrow[r] & F(U^{2(n+k)}) \\
& G(U^n) \arrow[ur, "\psi_{U^n}"'] & &
\end{tikzcd}
\tag{3.6}
$$

the image of a point in $F(U)$ is the same (following both paths) in $F(U^{2(n+k)})$ even if not in $F(U^{2n})$.

An Example: Consider Fig. 4 and fix $n = 1$. Denote the connected component of the point a in $F(U)$, $F(U^1)$, and $F(U^2)$ by A, A', and A'', respectively. Similarly, the connected component of the point b is denoted by $B'' \in G(U^2)$. Follow the same form for the connected components of points w and z in G. The interleaving diagrams can be collected together as

$$
\begin{tikzcd}
\{A\} \arrow[r, "F[U\subseteq U^1]"] \arrow[dr, "\varphi_U"'] & \{A'\} \arrow[r, "F[U^1\subseteq U^2]"] \arrow[dr, "\varphi_{U^n}"'] & \{A'', B''\} \\
\{W, Z\} \arrow[r, "G[U\subseteq U^1]"'] \arrow[ur, "\psi_U"] & \{W', Z'\} \arrow[r, "G[U^1\subseteq U^2]"'] \arrow[ur, "\psi_{U^1}"] & \{W'', Z''\}
\end{tikzcd}
\tag{3.7}
$$

noting that the horizontal maps are determined by sending a letter to the same letter with an additional prime. The distances between the points in their respective sets are

$$
d^F_{U^2}(A'', B'') = 1;
$$

$$
d^G_U(W, Z) = 3, \quad d^G_{U^1}(W', Z') = 2, \quad d^G_{U^2}(W'', Z'') = 1.
$$

Consider the following example assignment:

$$
\begin{aligned}
\varphi_U &: A \mapsto W', & \psi_U &: W, Z \mapsto A', \\
\varphi_{U^1} &: A' \mapsto W'', & \psi_{U^1} &: \substack{W' \mapsto A'' \\ Z' \mapsto B''}.
\end{aligned}
$$

In this case, we then have that $L^{U,U^n}_{\boxbslash} = 0$, $L^{U,U^n}_{\varparallelogram} = 1$, $L^U_{\bigtriangledown} = 0$, and $L^U_{\bigtriangleup} = 1$, so again $L(\varphi, \psi) \geq 1$. For this particular example, no $n = 1$ interleaving is possible so any choice of assignment will have a non-zero loss (the easiest check is to see that any choice of assignment will force $L^U_{\bigtriangleup} = 1$).

3.2 Bounding the Interleaving Distance

We now use the loss function to give an upper bound for the interleaving distance.

Theorem 3.8. *For an n-assignment, $\varphi\colon F \Rightarrow G^n$ and $\psi\colon G \Rightarrow F^n$,*

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

To prove this, we require the following technical lemma, proved in Sec. 6.

Lemma 3.9. *Assume we are given an n-assignment $\varphi : F \Rightarrow G^n$ and $\psi : G \Rightarrow F^n$. For a fixed k, define $(n+k)$-assignments $\Phi_U = G[U^n \subseteq U^{n+k}] \circ \varphi_U$ and $\Psi_U = F[U^n \subseteq U^{n+k}] \circ \psi_U$ for all $U \in$ **Open**(K). Then the following hold:*

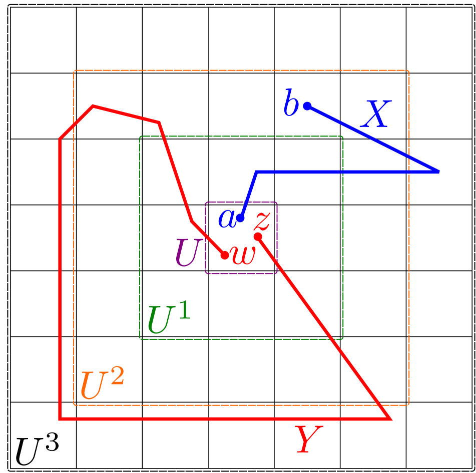


Figure 4: An example of two input geometric graphs, X and Y.

1. *$L^{U,V}_{\diagdown}(\varphi) \le k$ implies $\diagdown_\Phi(U,V)$ commutes, and thus $L^{U,V}_{\diagdown}(\Phi) = 0$.*
2. *$L^{U,V}_{\diagup}(\psi) \le k$ implies $\diagup_\Psi(U,V)$ commutes, and thus $L^{U,V}_{\diagup}(\Psi) = 0$.*
3. *$L^{U}_{\triangledown}(\varphi,\psi) \le k$ and $L^{U^n,U^{n+k}}_{\diagup}(\psi) \le k$ imply $\triangledown_{\Phi,\Psi}(U)$ commutes, and thus $L^{U}_{\triangledown}(\Phi,\Psi) = 0$.*
4. *$L^{U}_{\triangle}(\varphi,\psi) \le k$ and $L^{U^n,U^{n+k}}_{\diagdown}(\varphi) \le k$ imply $\triangle_{\Phi,\Psi}(U)$ commutes, and thus $L^{U}_{\triangle}(\Phi,\Psi) = 0$.*

In particular, if φ and ψ have $L(\varphi,\psi) = 0$, then φ and ψ constitute an interleaving, and so $d_I(F,G) \le n$.

Proof of Thm. 3.8 . Set $k = L(\varphi,\psi)$, so by definition, $L^{U,V}_{\diagdown}(\varphi) \le k$, $L^{U,V}_{\diagup}(\psi) \le k$, $L^{U}_{\triangledown}(\varphi,\psi) \le k$, and $L^{U}_{\triangle}(\varphi,\psi) \le k$. As in Lem. 3.9, construct two $(n+k)$-assignments: Φ given by $\Phi_U = G[U^n \subseteq U^{n+k}] \circ \varphi$, and Ψ given by $\Psi_U = F[U^n \subseteq U^{n+k}] \circ \psi$. By Lem. 3.9, this means the diagrams $\diagdown_\Phi(U,V)$, $\diagup_\Psi(U,V)$, $\triangledown_{\Phi,\Psi}(U)$, and $\triangle_{\Phi,\Psi}(U)$ commute for all pairs $U \subseteq V$. This implies that Φ and Ψ are an $(n+k)$-interleaving, giving the theorem. $\square$

First, notice that this proof works by explicitly constructing an interleaving from a given n-assignment. Second, we have no reason to believe that this bound is tight. In particular, in Sec. 3.3 we improve the bound by way of restricting the computation to the basis for the topology of K but even that is depending on input quality and gives no guarantee.

We include one additional note about when this loss function can be promised to be finite. Define the diameter of a metric space to be the largest distance between points, which we denote by $\mathrm{diam}(X,d) = \sup\{d(a,b) \mid a,b \in X\}$, and note that here, the sup can be replaced with a max since we are working in finite metric spaces.

Lemma 3.10. *The loss function for an n-assignment (φ,ψ) is bounded above; specifically,*

$$L(\varphi,\psi) \le \max\Bigg(\Big\{ \mathrm{diam}(F(U^k), d_F^{U^k}) \mid U \in \mathbf{Open}(K), k \in \{n, 2n\} \Big\} \cup \Big\{ \mathrm{diam}(G(U^k), d_G^{U^k}) \mid U \in \mathbf{Open}(K), k \in \{n, 2n\} \Big\} \Bigg).$$

In particular, if the original graphs are each connected, then $L(\varphi,\psi)$ is finite.

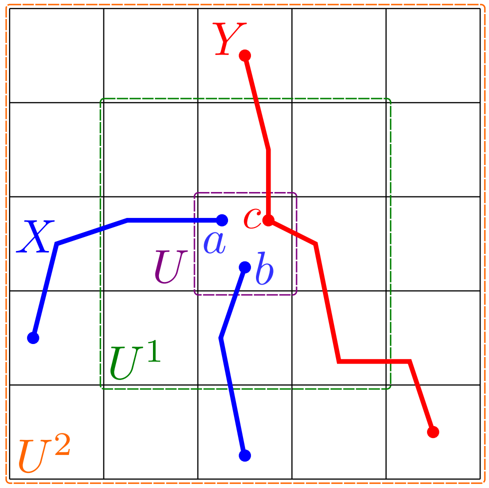


Figure 5: An example of the comparison of two geometric graphs with different numbers of connected components. In this case, because X has one connected component and Y has two, the loss function will be infinite for any assignment.

Proof. The parallelogram portions of the loss function $L_{\diagdown}$ and $L_{\diagup}$ take values from distances in $F(U^n)$ and $G(U^n)$, . The triangle portions $L_{\bigtriangledown}$ and $L_{\bigtriangleup}$ take values from distances in $F(U^{2n})$ and $G(U^{2n})$. So, the maximum for the loss function must be attained on one of these sets, giving the inequality. For the second statement, if the input graphs each have a single connected component, then any pair of elements $a, b \in F(U)$ map to the same element under the inclusion $F(U) \to F(U^K)$ for a large enough K. This in turn implies that the diameter of d_U^F is finite for every U. $\square$

Consider the example in Fig. 5. Let $\{A, B\}$, $\{A', B'\}$, and $\{A'', B''\}$ be the representatives of the connected components of the points a and b in $F(U)$, $F(U^1)$ and $F(U^2)$ respectively. Because there is no n for which the two points are the same connected component of X, the distance between A and B is ∞ in all three sets. Then no matter the choice of 1-assignment, $L_{\bigtriangledown} = \infty$, making the loss function infinite.

3.3 Restriction to Basis Elements

We have so far measured the loss function by studying all possible open sets U. While this is helpful for definitions, it does not make for a reasonable computational setting. To that end, we now focus on a basis of the topology, and prove that this basis suffices.

Definition 3.11. *An open set defined by the upset of a cell $\sigma \in K$ (that is, a vertex, edge, square, etc), $U_\sigma = \{\tau \mid \tau \geq \sigma\}$, is called a* basic open set.

Note that $\{U_\sigma \mid \sigma \in K\}$ is a basis for the Alexandroff topology. Also, this is an order reversing process, as for $\sigma \leq \tau$, $U_\tau \subseteq U_\sigma$. We next give a name to the case where we are only given n-assignment information for basis elements, or equivalently, if we are given a full assignment but ignore the maps for non-basis open sets.

Definition 3.12. *A* basis unnatural transformation *for functors H and H' is a collection of maps η_{U_σ} : $H(U_\sigma) \to H'(U_\sigma)$ for all basis elements U_σ from $\sigma \in K$. A* basis n-assignment *(or simply a basis assignment) is a pair of basis unnatural transformations*

$$\{\varphi_{U_\sigma} : F(U_\sigma) \to G(U_\sigma^n) \mid \sigma \in K\} \qquad \textit{and} \qquad \{\psi_{U_\sigma} : G(U_\sigma) \to F(U_\sigma^n) \mid \sigma \in K\}$$

In this section, we prove that we can focus our loss function efforts on only those diagrams associated to basic opens, and the solution can be extended to any open set.

Definition 3.13. *The* basis loss function *is defined to be*

$$L_B(\varphi, \psi) = \max_{\sigma \leq \tau} \left\{ L_{\boxbslash}^{U_\tau, U_\sigma}, L_{\boxslash}^{U_\tau, U_\sigma}, L_{\triangle}^{U_\sigma}, L_{\bigtriangledown}^{U_\sigma} \right\}.$$

It is immediate from the definitions that $L_B \leq L$ as the L_B maximum is taken over a subset of those used to determine L. This means, in particular, that if $L = 0$ then $L_B = 0$. These values are not always equal; for instance, we might have chosen a basis assignment for which every diagram commutes (making $L_B = 0$), but φ_V defined on non-basis elements causes a non-zero loss function so $L > 0$. However in the special case where $L_B = 0$, and thus the basis open diagrams are commutative, we do have the ability to extend the information checked to a full interleaving. This can be seen in the following lemma, proved in Sec. 6.

Lemma 3.14. *Given a basis unnatural transformation*

$$\{\Phi_{U_\sigma} : F(U_\sigma) \to G(U_\sigma^N) \mid \sigma \in K\}$$

with $L_{\boxbslash}^{U_\tau, U_\sigma} = 0$ *for all* $\sigma \leq \tau$*, we can extend this to a full natural transformation* Φ*; i.e. we can define* Φ_U *for all* U *such that* $L_{\boxbslash}^{U,V} = 0$ *for all* $U \subseteq V$*.*

Note that the symmetric version extending a basis unnatural transformation Ψ to a natural transformation $\Psi : G \Rightarrow F^N$ is obtained in exactly the same way. Next, we can take these natural transformations and ensure the triangles commute (thus giving an interleaving) by only checking the basis set triangles, again proved in Sec. 6.

Lemma 3.15. *Given natural transformations* $\Phi : F \Rightarrow G^N$ *and* $\Psi : G^N \Rightarrow F$ *such that* $L_{\bigtriangledown}^{U_\sigma} = 0$ *for all* $\sigma \in K$*, then* $L_{\bigtriangledown}^{U} = 0$ *for all open sets* U*.*

Taken together, we immediately have the following proposition.

Proposition 3.16. *Fix a basis* N*-assignment* (Φ, Ψ)*. If* $L_B(\Phi, \Psi) = 0$*, then* Φ *and* Ψ *can be extended to natural transformations with* $L(\Phi, \Psi) = 0$*, and thus constitute an interleaving.*

Finally, we arrive at our main theorem, where we can use the provided basis n-assignment and the calculated loss function to give a bound for the interleaving distance.

Theorem 3.17. *Given a basis* n*-assignment*

$$\varphi = \{\varphi_{U_\sigma} \mid \sigma \in K\} \text{ and } \psi = \{\psi_{U_\sigma} \mid \sigma \in K\},$$

we have

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

Proof. This proof proceeds in the same way as that of Thm. 3.8 with some minor modifications of input assumptions. First, let $k = L_B(\varphi, \psi)$; and define a basis $(n + k)$-assignment by

$$\{\Phi_{U_\sigma} = G[\subseteq] \circ \varphi_{U_\sigma} \mid \sigma \in K\} \qquad \text{and} \qquad \{\Psi_{U_\sigma} = F[\subseteq] \circ \psi_{U_\sigma} \mid \sigma \in K\}.$$

By Lem. 3.9, we know that $L_{\boxbslash}^{U_\tau, U_\sigma}(\Phi) = 0$ and $L_{\boxslash}^{U_\tau, U_\sigma}(\Psi) = 0$ for all $\tau \leq \sigma$. Then by Lem. 3.14, we can extend Φ and Ψ to full natural transformations defined for all $U \in \mathbf{Open}(K)$.

To show that Φ and Ψ constitute an $(n+k)$-interleaving, we must check triangles; i.e. ensure that $L_{\bigtriangledown}^{U}(\Phi,\Psi) = L_{\bigtriangleup}^{U}(\Phi,\Psi) = 0$. With the goal of using part 3 of Lem. 3.9, first note that $L_{\bigtriangledown}^{U_\sigma}(\varphi,\psi) \leq k$ for basis elements. We can see that $L_{\varparallelogram}^{U_\sigma^n, U_\sigma^{n+k}} \leq k$ by using the (non-commutative) diagram

$$\begin{tikzcd}
& F(U_\sigma^{2n}) \arrow[rr, "F[\subseteq]"] & & F(U_\sigma^{2n+k}) \arrow[r, "F[\subseteq]"] & F\left(U_\sigma^{2(n+k)}\right) \\
G(U_\sigma^{n}) \arrow[ur, "\psi_\bullet"] \arrow[urrr, "\Psi_\bullet"] \arrow[rr, "G[\subseteq]"'] & & G(U_\sigma^{n+k}). \arrow[ur, "\psi_\bullet"] \arrow[urr, "\Psi_\bullet"'] & &
\end{tikzcd}$$

The leftmost and rightmost triangles commute by definition of Ψ, and the orange parallelogram commutes because Ψ is a natural transformation. Then chasing any $x \in G(U_\sigma^n)$ up to the top right $F\left(U_\sigma^{2(n+k)}\right)$ results in the same element, giving the required bound on $L_{\varparallelogram}^{U_\sigma^n, U_\sigma^{n+k}}$. Using Lem. 3.15 for Φ and Ψ, $L_{\bigtriangledown}^{U}(\Phi,\Psi) = 0$ for all open sets U. The proof that $L_{\bigtriangleup}^{U}(\Phi,\Psi) = 0$ is similar. Therefore Φ and Ψ are an $(n+k)$-interleaving, giving the bound. $\square$

What is surprising about this bound is that despite checking fewer open sets, the loss function for L_B is actually lower than that found using L. One reason for this is that when we work with the smaller set of input maps, we extend the collection to a "better" full assignment, potentially getting rid of some of the causes of a nonzero loss function in the first place. For example, a full assignment would be required to provide a map φ_U for a U with multiple connected components, say $U = V_1 \cup V_2$. Since no requirements were made of this map based on the φ_{V_1} and φ_{V_2} maps, there is a reasonable chance that the loss function contribution from the $L_{\bigtriangledown}^{V_1,U}$ is higher than necessary. However, in the basis version, we can build the best possible φ_V given the information over φ_{U_1} and φ_{U_2}, providing a potentially better, but certainly no worse, bound.

4 Computation

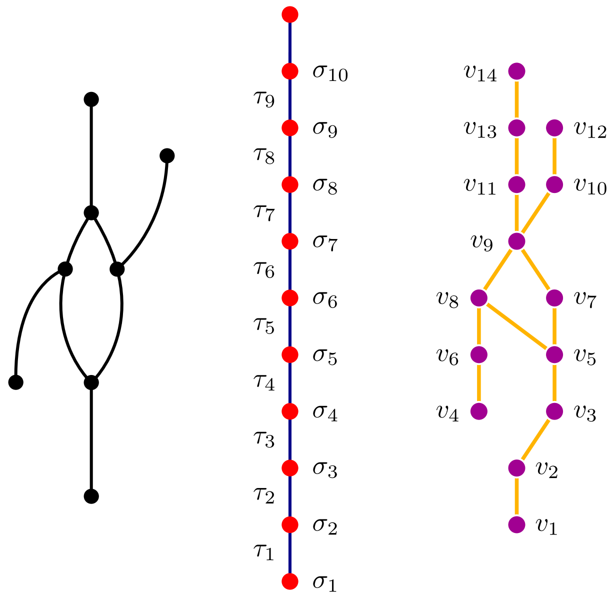
In this section, we show that given an n-assignment φ and ψ, we can compute the loss function $L_B(\varphi,\psi)$ in polynomial time. For simplicity, we describe the algorithm explicitly in the case where $d=1$ for clarity of exposition, before addressing the run time in higher dimensions.

4.1 Data Structures

In this section, we describe the encoding of the data structures for a pair of input functors F, G and a given n-assignment φ and ψ. We will start with the case $d=1$, and follow the example of Fig. 6 to illustrate our construction. At a high level, we construct graphs for F and G, which we denote by (V_F, E_F) and (V_G, E_G). Then we build data structures to encode the natural transformations φ and ψ. For clarity, we use `phi` and `psi` to denote the data structures that store information for φ and ψ, respectively. These encode set maps `phi` $: (V_F, E_F) \to (V_G, E_G)$ and `psi` $: (V_G, E_G) \to (V_F, E_F)$, which map each vertex to a vertex in the other graph and each edge to an edge in the other graph.

When $d=1$, recall that K, i.e., the discretization of $\mathbb{R}$, consists of vertices $\sigma_{-L}, \cdots, \sigma_L$ with heights in our bounding box $[-L\delta, L\delta]$, and with edges $\tau_j = (\sigma_j, \sigma_{j+1})$. Then we construct the graph for $F : \mathbf{Open}(K) \to \mathbf{Set}$ by generating a vertex for every object in every $F(\sigma_i)$ and connect them using the morphisms of the functor. This results in a vertex set $V_F = \coprod_{i=1}^{B} F(U_{\sigma_i})$, and an edge for every object in every $F(U_{\tau_i})$, giving edge set $E_F = \coprod_{i=1}^{B-1} F(U_{\tau_i})$. The endpoints of any edge $e \in F(U_{\tau_i}) \subseteq E_F$ can be found via the attaching maps:

$$F[U_{\tau_i} \subseteq U_{\sigma_i}](e) \in F(U_{\sigma_i}) \text{ and}$$
$$F[U_{\tau_i} \subseteq U_{\sigma_{i+1}}](e) \in F(U_{\sigma_{i+1}}).$$



$v_1 : \{v_2\}, \quad \sigma_2$
$v_2 : \{v_3, v_1\}, \quad \sigma_3$
$v_3 : \{v_5, v_2\}, \quad \sigma_4$
$v_4 : \{v_6\}, \quad \sigma_4$
$v_5 : \{v_3, v_7, v_8\}, \quad \sigma_5$
$v_6 : \{v_4, v_8\}, \quad \sigma_5$
$v_7 : \{v_5, v_9\}, \quad \sigma_6$
$v_8 : \{v_5, v_6, v_9\}, \quad \sigma_6$
$v_9 : \{v_7, v_8, v_{10}, v_{11}\}, \quad \sigma_7$
$v_{10} : \{v_9, v_{12}\}, \quad \sigma_8$
$v_{11} : \{v_9, v_{13}\}, \quad \sigma_8$
$v_{12} : \{v_{10}\}, \quad \sigma_9$
$v_{13} : \{v_{11}, v_{14}\}, \quad \sigma_9$
$v_{14} : \{v_{13}\}, \quad \sigma_{10}$

Figure 6: From left to right: an example input Reeb graph ($d = 1$), a discretization of $\mathbb{R}$, the generated mapper graph, and the data structure encoding the mapper graph.

For example, $e = (v_4, v_6) \in F(U_{\tau_4})$ in Fig. 6 has endpoints $v_6 \in F(U_{\sigma_4})$ and $v_4 \in F(U_{\sigma_5})$. We store this data in a standard adjacency list. In addition, each vertex also keeps track of its height, so a vertex $v \in F(U_{\sigma_i})$ also stores the value i as a representation of its height.

Next, we encode the information for an assignment (φ, ψ) between $F, G : \textbf{Open}(K) \to \textbf{Set}$ by constructing the maps `phi` and `psi` using the graphs (V_F, E_F) and (V_G, E_G). Specifically, for every $v \in V_F$, we store a vertex `phi`$(v) \in V_G$ with the requirement that if $v \in F(U_{\sigma_i})$ and `phi`$(v) \in G(U_{\sigma_j})$, then $|i - j| \le n$. In addition, for every $e \in E_F$, we store an edge `phi`$(e) \in E_G$ again with the requirement that if $e \in F(U_{\tau_i})$ and `phi`$(e) \in G(U_{\tau_j})$ then $|i - j| \le n$. The symmetric situation is setup for `psi`.

To see how these maps arise from an input assignment φ and ψ, we start by focusing on the vertex set. For this, we need to encode the map $\varphi_{U_{\sigma_i}} : F(U_{\sigma_i}) \to G(U^n_{\sigma_i})$. The elements of $F(U_{\sigma_i})$ are already encoded as vertices, however the elements of $G(U^n_{\sigma_i})$ are not. But, because of the cosheaf structure of G, the elements of $G(U^n_{\sigma_i})$ can be seen as the connected components of particular subgraphs. Let

$$V_{G,\sigma_i,n} = \{v \mid v \in G(\sigma_j), j \in [i - n, i + n]\}$$

and

$$E_{G,\sigma_i,n} = \{e \mid e \in G(\tau_j), j \in [i - n - 1, i + n]\}.$$

The elements of $G(U^n_{\sigma_i})$ are the connected components of the subset of the graph $(V_G, E_G)_{\sigma_i,n} := (V_{G,\sigma_i,n}, E_{G,\sigma_i,n})$. Note that because of the endpoints, this is not an induced subgraph; see Fig. 7 for examples. Similarly for the edges of K, we can define

$$V_{G,\tau_i,n} = \{v \mid v \in G(\sigma_j), j \in [i - n + 1, i + n]$$

and

$$E_{G,\tau_i,n} = \{e \mid e \in G(\tau_j), j \in [i - n, i + n]\}$$

so that the connected components of $(V_G, E_G)_{\tau_i,n} := (V_{G,\tau_i,n}, E_{G,\tau_i,n})$ are the elements of $G(U^n_{\tau_i})$.

So, for each $v \in F(U_{\sigma_i})$, we store a vertex `phi`$(v) \in V_{G,\sigma_i,n}$, where `phi`(v) is in the connected component of $(V_G, E_G)_{\sigma_i,n}$ represented by $\varphi_{U_{\sigma_i}}(v) \in G(U^n_{\sigma_i})$. For instance, consider the example of Fig. 7 where we

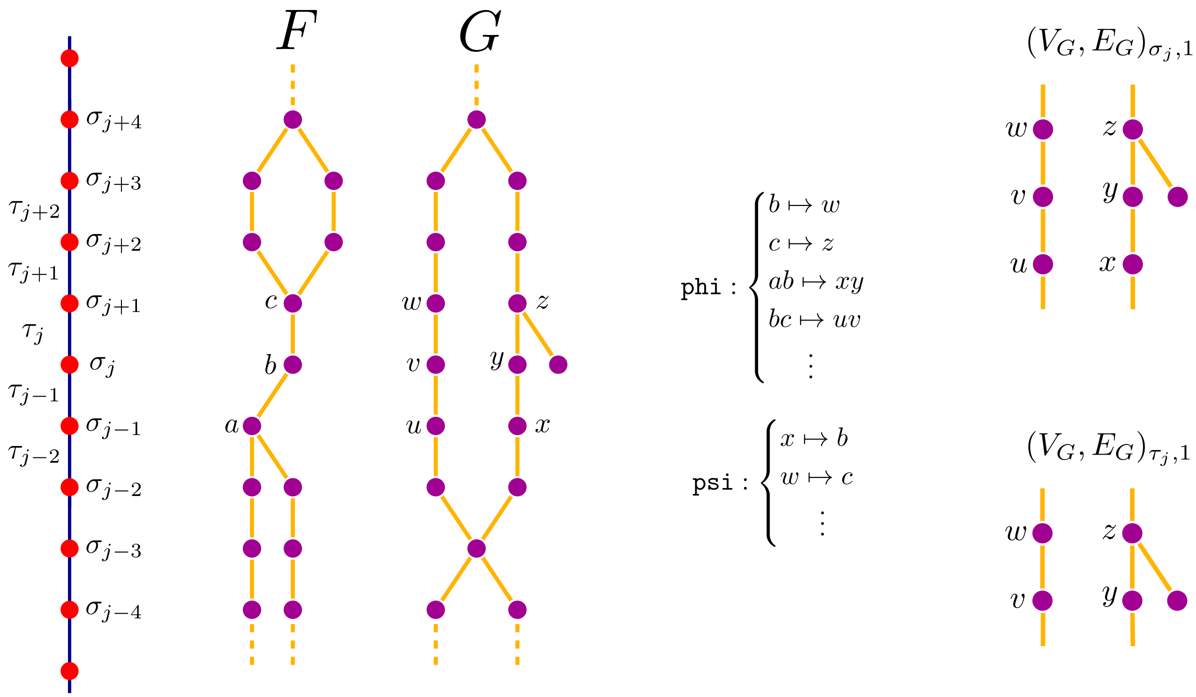


Figure 7: At left are two example input mapper graphs. In the middle are a subset of an example `phi` and `psi` input assignment used throughout the text. At right are two example slices of the graphs used when checking connectivity.

assume $n = 1$. If $\texttt{phi}(b) = w$, then $\varphi_{U_{\sigma_i}}(b)$ is the connected component that includes w of $(V_G, E_G)_{\sigma_i,1}$ as shown on the right. We can similarly find the edge map $\texttt{phi}(e)$ for $e \in F(U_{\tau_i})$ by setting it to be an edge in $E_{G,\tau_i,n}$ representing the connected component of $\varphi_{U_{\tau_i}}(e) \in G(U^n_{\tau_i})$ in $(V_G, E_G)_{\tau_i,n}$. So, for example, in Fig. 7 where $n = 1$, the input data might have $\texttt{phi}(ab) = (xy) \in E_G$ and $\texttt{phi}(bc) = (uv) \in E_G$.

4.2 Algorithm and Complexity

In this section, we discuss the complexity of determining $L_B(\varphi, \psi)$ given `phi` and `psi`. First, we will proceed using a binary search on $k \in [0, \cdots, 2L]$ where the maximum is determined by the diameter of the bounding box. For a fixed k, we will determine if $L_B(\varphi, \psi) \le k$. We will describe the cases for $L^{U_\tau,U_\sigma}_{\diagdown}$ and $L^{U_\sigma}_{\triangledown}$, as $L^{U_\tau,U_\sigma}_{\diagup}$ and $L^{U_\sigma}_{\vartriangle}$ are symmetric.

Start with $L^{U_\tau,U_\sigma}_{\diagdown}$ and note that in the case where $d = 1$, there are two pairs necessary to check for each edge: τ_j, σ_j and τ_j, σ_{j+1}. Fixing σ_ℓ to be either σ_j or σ_{j+1}, for each edge $e \in F(U_{\tau_j})$, we need to check if the two possible images in $G(U^{n+k}_{\sigma_\ell})$ under the diagram

$$
\begin{tikzcd}
F(U_{\tau_i}) \arrow[r, "F[\subseteq]"] \arrow[dr, "\varphi_{U_{\tau_i}}"'] & F(U_{\sigma_\ell}) \arrow[dr, "\varphi_{U^n_{\tau_i}}"] & & \\
& G(U^n_{\tau_i}) \arrow[r, "G[\subseteq]"'] & G(U^n_{\sigma_\ell}) \arrow[r] & G(U^{n+k}_{\sigma_\ell})
\end{tikzcd}
\qquad
\begin{tikzcd}
e \arrow[r, mapsto] \arrow[dr, mapsto] & v \arrow[dr, mapsto] & & \\
& {[e']} \arrow[r, mapsto] & \begin{smallmatrix}[w]\\ [e']\end{smallmatrix} \arrow[r, Mapsto] & \begin{smallmatrix}[w]\\ [e']\end{smallmatrix}
\end{tikzcd}
\tag{4.1}
$$

are the same. Note that we use $[-]$ to note that the elements represent the connected component in the relevant sliced graph containing that edge or vertex. Following the top of diagram Eq. (4.1), we know that e has a unique endpoint vertex $v \in F(U_{\sigma_\ell})$, and that vertex has an image under $\varphi_{U^n_{\tau_i}}$, which is a connected component represented by $\texttt{phi}(v) = w \in V_G$. Following down, the edge e has an edge image $\texttt{phi}(e) = e' \in E_G$.

So the question becomes: are e' and w in the same connected component of $(V_G, E_G)_{\sigma_\ell, n+k}$? This can be done by filtering through the adjacency lists, keeping only vertices and edges in the correct strip, and then checking for connectivity using a standard graph traversal like breadth or depth first search. As we do this once per grid element, we get a total time (when $d = 1$) of $O(V_G + E_G)$ time per parallelogram. If $d > 1$, then the correct "strip" for $\sigma_{\overrightarrow{\ell}}$ with indices $\overrightarrow{\ell} \in \mathbb{Z}^d$ involves checking a portion of the graph with indices in a d-dimensional box $[\ell_1 - (n+k), \ell_1 + (n+k)] \times \cdots \times [\ell_d - (n+k), \ell_d + (n+k)]$ and hence takes $O(d(V_G + E_G))$ time.

In the example of Fig. 7, assume $n = k = 1$ and assume the given input `phi` is as noted. Then for the diagram of Eq. (4.1) with $\ell = j$ and chasing $bc \in F(U_{\tau_j})$, this comes down to checking if the connected component of $\texttt{phi}(b) = w$ and $\texttt{phi}(bc) = xy$ are the same in the portion of $(V_G, E_G)_{\sigma_j, 2}$. In this particular example, there are two connected components in this slice and the images are not in the same component. Then we know that $L_{\diagdown}^{U_{\tau_j}, U_{\sigma_j}} > k$ so we would immediately move on in our binary search. If it were the case that the two images were in the same connected component, then $L_{\diagdown}^{U_{\tau_j}, U_{\sigma_j}} \le k$ and thus we would move on to the next commutative diagram check.

Checking if $L_{\bigtriangledown}^{U_\tau} \le k$ is similar so we briefly highlight the differences. First, there are two types of basis elements in our case where $d = 1$, so we need to check $L_{\bigtriangledown}^{U_{\sigma_i}} \le k$ (meaning checking vertices) and $L_{\bigtriangledown}^{U_{\tau_i}} \le k$ (meaning checking edges). We focus on the case of vertices since the edge version is similar. For any vertex element $v \in U_{\sigma_i}$, we need to chase it around the diagram

$$
\begin{tikzcd}
F(U_{\sigma_i}) \arrow[rr, "F{[U_{\sigma_i} \subseteq U_{\sigma_i}^{2n}]}"] \arrow[dr, "\varphi_{U_{\sigma_i}}"'] & & F(U_{\sigma_i}^{2n}) \arrow[r] & F(U_{\sigma_i}^{2(n+k)}). \\
& G(U_{\sigma_i}^{n}) \arrow[ur, "\psi_{U_{\sigma_i}^{n}}"'] & &
\end{tikzcd}
\qquad
\begin{tikzcd}
v \arrow[rr, mapsto] \arrow[dr, mapsto] & & {[v]} \atop {[v']} \arrow[r, Mapsto] & {[v]} \atop {[v']} \\
& w \arrow[ur, mapsto] & &
\end{tikzcd}
\tag{4.2}
$$

If `phi` : $v \mapsto w$, and `psi` : $w \mapsto v'$, the question again becomes: are v and v' in the same connected component of $(V_G, E_G)_{\sigma_j, 2(n+k)}$? Similar to the parallelogram case, we take a strip of the graph and check this connectivity question in $O(d(V_G + E_G))$ time. As before, either the elements checked are in the same connected component of the relevant slice of the graph, in which case we move to the next diagram; or it does not, and we move to a different k in our binary search. In our example case of Fig. 7 with $n = k = 1$, we have $2(n + k) = 4$. Then chasing b, we need to check that b and $\texttt{psi} \circ \texttt{phi}(b) = c$ are in the same connected component of $(V_G, E_G)_{\sigma_j, 4}$. As this slice has one connected component, this triangle commutes. We can check another triangle $L_{\bigtriangleup}^{U_{\sigma_j}} \le k$ chasing w. In this case, we must check if w and $\texttt{phi} \circ \texttt{psi}(w) = z$ are in the same component of $(V_G, E_G)_{\sigma_i, 4}$, which again, they both are. In either case, if they were not, we would know the loss function is at least k and continue in the binary search.

To count the number of (complected) diagram checks, a vertex $v \in F(U_{\sigma_i})$ is checked for one triangle loss $L_{\bigtriangledown}^{U_{\sigma_i}}$; and a vertex $w \in G(U_{\sigma_i})$ is checked for one triangle loss $L_{\bigtriangleup}^{U_{\sigma_i}}$. An edge $e \in F(U_{\tau_i})$ is checked for one triangle loss $L_{\bigtriangledown}^{U_{\tau_i}}$ and for two parallelograms: $L_{\diagdown}^{(U_{\tau_i}, U_{\sigma_i})}$ and $L_{\diagdown}^{(U_{\tau_i}, U_{\sigma_{i+1}})}$. Likewise, an edge $e' \in G(U_{\tau_i})$ is checked in diagrams $L_{\bigtriangleup}^{U_{\tau_i}}$, $L_{\diagup}^{(U_{\tau_i}, U_{\sigma_i})}$ and $L_{\diagup}^{(U_{\tau_i}, U_{\sigma_{i+1}})}$. This means that if the graph representations of F and G are (V_F, E_F) and (V_G, E_G) respectively, the time for computing the loss function is

$$
O\Bigg([(V_F + V_G) + 3(E_F + E_G)] \cdot \max\{(V_F + E_F), (V_G + E_G)\} \Bigg).
$$

In d-dimensions, a similar construction holds, except that our σ cells are now indexed by B^d, giving an extra multiplicative factor of B^d in the run time.

5 Extension to Reeb Graphs

We now take a brief diversion into understanding how the loss function framework can be used to approximate the Reeb graph interleaving distance. In this case, we consider a 1-dimensional mapper graph to be an approximation of the Reeb graph [5, 6, 10, 11, 28]. We show that in order to bound the Reeb graph interleaving distance, we can compute the mapper graph for a resolution δ, and then use the loss function to provide a similar bound.

5.1 Definitions

Given input data $f : \mathbb{X} \to \mathbb{R}$, the Reeb graph of $(\mathbb{X}, f)$ is computed as follows. Define an equivalence relation by setting $x \sim y$ iff x and y are in the same path-connected component of the levelset $f^{-1}(a)$. With enough restrictions on the space and function (for example, a Morse function on a manifold), the resulting Reeb graph is a topological graph; i.e. a 1-dimensional stratified space. Similar to the vantage taken for the mapper graphs in this paper, the data of a Reeb graph can be stored in a cosheaf.

Definition 5.1. *For a given $(\mathbb{X}, f)$, the associated Reeb cosheaf is given by*

$$\begin{array}{rccc} \widetilde{F} : & \mathbf{Int} & \to & \mathbf{Set} \\ & I & \mapsto & \pi_0 f^{-1}(I) \\ & \text{\rotatebox{90}{$\subseteq$}} & & \downarrow \pi_0[\subseteq] \\ & J & \mapsto & \pi_0 f^{-1}(J) \end{array}$$

where morphisms are induced by the π_0 functor.

For clarity, we write the Reeb cosheaf with a tilde to distinguish it from the mapper cosheaf without a tilde. Given this input, we have the Reeb graph interleaving distance [33], given as follows.

Definition 5.2. *Define the functor $(-)^\varepsilon : \mathbf{Int} \to \mathbf{Int}$ by $(a, b) \mapsto (a - \varepsilon, b + \varepsilon)$ with morphisms induced by inclusion. Then $\widetilde{F}_\varepsilon : \mathbf{Int} \to \mathbf{Set}$ is given by $\widetilde{F}_\varepsilon(J) = \widetilde{F}(J^\varepsilon)$.*

For given $\widetilde{F}, \widetilde{G} : \mathbf{Int} \to \mathbf{Set}$, an ε-interleaving is a pair of natural transformations $\widetilde{\varphi} : \widetilde{F} \Rightarrow \widetilde{G}_\varepsilon$ and $\widetilde{\psi} : \widetilde{G} \Rightarrow \widetilde{F}_\varepsilon$ such that

$$\begin{tikzcd} \widetilde{F}(I) \arrow[rr, "\widetilde{F}[I\subseteq I^{2n}]"] \arrow[dr, "\widetilde{\varphi}_I"'] & & \widetilde{F}(I^{2n}) \\ & \widetilde{G}(I^n) \arrow[ur, "\widetilde{\psi}_{I^n}"'] & \end{tikzcd} \qquad \begin{tikzcd} & \widetilde{F}(I^n) \arrow[dr, "\widetilde{\varphi}_{I^n}"] & \\ \widetilde{G}(I) \arrow[ur, "\widetilde{\psi}_I"] \arrow[rr, "\widetilde{G}[I\subseteq I^{2n}]"'] & & \widetilde{G}(I^{2n}) \end{tikzcd}$$

commute for all $I \in \mathbf{Int}$. The (categorical) Reeb graph interleaving distance is given by

$$d_R(\widetilde{F}, \widetilde{G}) = \inf\{\varepsilon \geq 0 \mid \text{ there exists an } \varepsilon\text{-interleaving}\}.$$

Fix a δ. Following Sec. 2, denote the vertices of K by $\{\sigma_{-L}, \cdots, \sigma_L\}$ where σ_i is at the point $i\delta \in \mathbb{R}$. Denote the edges by $\tau_i = (i\delta, (i+1)\delta)$ which has faces σ_i and σ_j. Given some input data $f : \mathbb{X} \to \mathbb{R}$, we can either construct its Reeb cosheaf $\widetilde{F} : \mathbf{Int} \to \mathbf{Set}$, or by fixing some choice of δ, we can construct its mapper cosheaf $F : \mathbf{Open}(K) \to \mathbf{Set}$.

We next show that the loss function we have computed here on the mapper version d_I can be used to similarly bound the Reeb interleaving distance d_R. We do this by showing that d_I is an approximation of d_R, which can be viewed as a special case of [5, Thm. 5.15]; however, for clarity, we include a direct proof in Sec. 6 as our setting allows a proof with considerably less use of category theoretic machinery.

Proposition 5.3. *For inputs $f : \mathbb{X} \to \mathbb{R}$ and $g : \mathbb{Y} \to \mathbb{R}$, denote the respective Reeb cosheaves as $\widetilde{F}, \widetilde{G} : \mathbf{Int} \to \mathbf{Set}$, and the respective mapper cosheaves as $F, G : \mathbf{Open}(K) \to \mathbf{Set}$. Then*

$$d_R(\widetilde{F}, \widetilde{G}) \leq (d_I(F, G) + 1)\,\delta.$$

Given this bound, we combine Prop. 5.3 with Thm. 3.17 to show that the loss function for the mapper graph discretization bounds the Reeb graph interleaving as well and that, in particular, this bound is controlled by the diameter δ chosen for K.

Corollary 5.4. *Given a basis n-assignment $\varphi = \{\varphi_{U_\sigma} \mid \sigma \in K\}$ and $\psi = \{\psi_{U_\sigma} \mid \sigma \in K\}$ for $F, G : \mathbf{Open}(K) \to \mathbf{Set}$, we have that*

$$d_R(\widetilde{F}, \widetilde{G}) \leq \delta(d_I(F,G) + 1) \leq \delta(n + L_B(\varphi, \psi) + 1).$$

6 Technical Proofs

In this section, we include the technical proofs from the previous sections.

6.1 Proofs from Sec. 2

Lemma 6.1. *$(-)^n$ is a functor.*

Proof. First, we check that the images of morphisms are well defined, which is to say that if $U \subseteq V$, then $U^n \subseteq V^n$. The statement is clear if $n = 0$, so by induction, we assume that $U^{n-1} \subseteq V^{n-1}$. Given an arbitrary $\sigma \in U^n$, the statement is immediate if $\sigma \in U^{n-1} \subseteq U^n$, so we assume $\sigma \in U^n \setminus U^{n-1}$. For this to happen, there must be a $\gamma \in U^{n-1}$ and $\tau \in K$ with $\gamma \geq \tau \leq \sigma$. But as $\gamma \in U^{n-1} \subseteq V^{n-1}$, this sequence also implies that $\sigma \in V^n$, finishing the well-defined check.

To ensure this is a functor, we need to check that the identity morphism is sent to the identity, and that composition holds. For the former, we see that $U \subseteq U$ gets sent to $U^n \subseteq U^n$, and each is an identity. The latter is immediate from the property that $\mathbf{Open}(K)$ is a poset category, meaning that there is at most one morphism between any pair of objects. ☐

One property of this construction that will be useful is as follows. For any $\sigma \in U^n$, there is a $\tau \in U$ and a sequence of cells

$$\tau \geq \gamma_1 \leq \tau_1 \geq \gamma_2 \leq \tau_2 \geq \cdots \geq \gamma_n \leq \sigma. \tag{6.2}$$

Further, given such a sequence with $\tau \in U$, we know that $\sigma \in U^n$. Two examples of this can be seen in Fig. 8, where σ and σ' from U^3 are given, along with a path satisfying Eq. (6.2). Of course, the choice of sequence for Eq. (6.2) is not unique, so other options are possible.

Next we show that the distance of Eq. (2.2) is indeed a distance using the super-linear family of translations framework of [8]. This construction can be generalized to the concept of a category with a flow [15], but the added generality is not needed here.

Definition 6.3 ([8])**.** *Let $P = (P, \leq)$ be a preordered set. A* translation *on P is a functor $\Gamma : P \to P$ along with a natural transformation $\eta : \mathbb{1}_P \Rightarrow \Gamma$. A* super-linear family of translations *is a collection $\{\Gamma_\varepsilon\}_{\varepsilon \geq 0}$ such that $\Gamma_\varepsilon \Gamma_{\varepsilon'}(p) \leq \Gamma_{\varepsilon+\varepsilon'}(p)$ for all $p \in P$, and $\varepsilon, \varepsilon' \geq 0$.*

Lemma 2.1. *For any $n, n' \geq 0$ and $U \in \mathbf{Open}(K)$,*

$$(U^n)^{n'} = U^{n+n'}.$$

Proof. First, we check that $(-)^n$ is indeed a translation using the above terminology. In particular, we define $\gamma^n : \mathbb{1}_{\mathbf{Open}(K)} \Rightarrow (-)^n$ to have components $\gamma^n_U : U \to U^n$ as simply the inclusion, and we can easily check that this satisfies the naturality requirements.

Fix $U \in \mathbf{Open}(K)$. We need to show that $(U^n)^{n'} = U^{n+n'}$. Let $\sigma \in (U^n)^{n'}$. By previous remarks, this is true if and only if there is a sequence

$$\tau \geq \gamma_1 \leq \tau_1 \geq \gamma_2 \leq \tau_2 \geq \cdots \geq \gamma_{n'} \leq \sigma$$

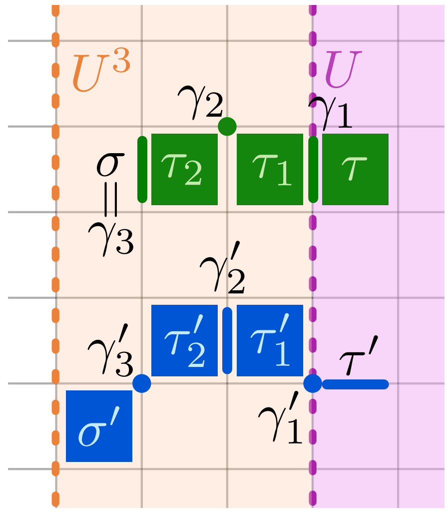


Figure 8: Given the purple set U, we have an edge σ and a square σ' which are elements of U^3. Then we provide an example sequence for each satisfying Eq. (6.2) leading to τ and τ' in U.

with $\tau \in U^n$. But this property of τ happens iff there is also a sequence

$$\tau' \geq \gamma_1' \leq \tau_1' \geq \gamma_2' \leq \tau_2' \geq \cdots \geq \gamma_n' \leq \tau$$

with $\tau' \in U$. Concatenating the two sequences gives a sequence of length $(n+n')$ from $\tau \in U$ to σ. Thus $\sigma \in U^{n+n'}$ iff $\sigma \in (U^n)^{n'}$, and hence $(U^n)^{n'} = U^{n+n'}$. □

Theorem 6.4. *The interleaving distance of Defn. 2.2 is an extended pseudometric.*

Proof. Because Lem. (2.1) is a stronger requirement than needed for Defn. 6.3, the collection $\{(-)^n\}_{n\geq 0}$ forms a super-linear family of translations. Then the result is immediate from [8, Theorem 3.21]. □

6.2 Proofs from Sec. 3

Lemma 3.9. *Assume we are given an n-assignment $\varphi : F \Rightarrow G^n$ and $\psi : G \Rightarrow F^n$. For a fixed k, define $(n+k)$-assignments $\Phi_U = G[U^n \subseteq U^{n+k}] \circ \varphi_U$ and $\Psi_U = F[U^n \subseteq U^{n+k}] \circ \psi_U$ for all $U \in$ **Open**(K). Then the following hold:*

1. $L^{U,V}_{\diagdown}(\varphi) \leq k$ *implies* $\diagdown_\Phi(U,V)$ *commutes, and thus* $L^{U,V}_{\diagdown}(\Phi) = 0$.
2. $L^{U,V}_{\varparallelogram}(\psi) \leq k$ *implies* $\varparallelogram_\Psi(U,V)$ *commutes, and thus* $L^{U,V}_{\varparallelogram}(\Psi) = 0$.
3. $L^{U}_{\bigtriangledown}(\varphi,\psi) \leq k$ *and* $L^{U^n,U^{n+k}}_{\varparallelogram}(\psi) \leq k$ *imply* $\bigtriangledown_{\Phi,\Psi}(U)$ *commutes, and thus* $L^{U}_{\bigtriangledown}(\Phi,\Psi) = 0$.
4. $L^{U}_{\bigtriangleup}(\varphi,\psi) \leq k$ *and* $L^{U^n,U^{n+k}}_{\diagdown}(\varphi) \leq k$ *imply* $\bigtriangleup_{\Phi,\Psi}(U)$ *commutes, and thus* $L^{U}_{\bigtriangleup}(\Phi,\Psi) = 0$.

In particular, if φ and ψ have $L(\varphi,\psi) = 0$, then φ and ψ constitute an interleaving, and so $d_I(F,G) \leq n$.

Proof of Lem. 3.9. We prove the lemma for the first and third entries only as the other arguments are symmetric. Assume $L_{\diagdown}^{U,V}(\varphi) \leq k$ and consider the diagram

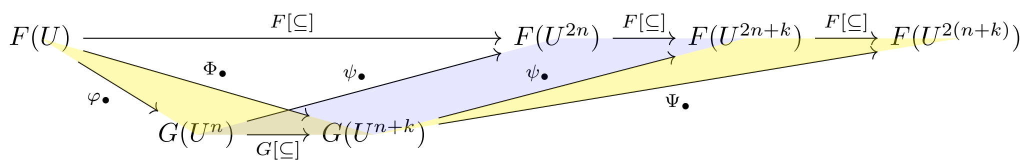
$$\begin{tikzcd}
F(U) \arrow[rr, "F[\subseteq]"] \arrow[dr, "\varphi_U"] \arrow[ddr, "\Phi_U"'] & & F(V) \arrow[dr, "\varphi_V"] \arrow[ddr, "\Phi_V"'] & \\
 & G(U^n) \arrow[rr, "G[\subseteq]"'] \arrow[d, "G[\subseteq]"] & & G(V^n) \arrow[d, "G[\subseteq]"] \\
 & G(U^{n+k}) \arrow[rr, "G[\subseteq]"] & & G(V^{n+k}).
\end{tikzcd}$$

Note that the top of the diagram does not necessarily commute in the case that $k \geq 1$, and the bottom of the diagram is $\diagdown_\Phi(U, V)$, for which we wish to check for commutativity. For any $x \in F(U)$, following around the top square gives

$$\begin{tikzcd}
x \arrow[r, mapsto] \arrow[d, mapsto] & x' \arrow[d, mapsto] \\
a \arrow[r, mapsto] & \substack{b' \\ a'}
\end{tikzcd}$$

with $d^G_{V^n}(a', b') \leq k$. By definition, the image of a' and b' is the same under the map $G(V^n) \to G(V^{n+k})$. Then since the front square commutes by functoriality of G, and the side triangles commute by definition of Φ, we have that the image of x under either direction of the back square commutes, proving claim (1).

Turning to claim (3), consider the noncommutative diagram



The two yellow triangles commute by definition of Φ and Ψ. The blue parallelogram is the diagram $\varparallelogram_\psi(U^n, U^{n+k})$ which also has loss function bounded by k, thus elements of $G(U^n)$ are not necessarily the same in the image of $F(U^{2n+k})$ following the parallelogram, but are the same in $F(U^{2(n+k)})$.

Checking that $\bigtriangledown_{\Phi,\Psi}(U)$ commutes amounts to a diagram chase. For an arbitrary $\alpha \in F(U)$, consider the following elements aligning with the diagram above,

$$\begin{tikzcd}
\alpha \arrow[rrr, mapsto] \arrow[dr] \arrow[drr] & & & \substack{a \\ a'} \arrow[r, mapsto] & \left.\substack{b \\ b' \\ b''}\right\} \arrow[r, mapsto] & c \\
 & x \arrow[r, mapsto] \arrow[urr, mapsto] & x' \arrow[urr, mapsto] \arrow[urrr, mapsto, bend right] & & &
\end{tikzcd}$$

Both α and x map to x' because of the yellow triangle commuting, and both b'' and x' map to the same c for the same reason. Even if α and x map to different elements in $F(U^{2n})$, they must map to the same element in $F(U^{2(n+k)})$, and this element must be c, since both b' and b'' map to the same element by the bound on the blue parallelogram. As this was done for an arbitrary α, we have that $\bigtriangledown_{\Phi,\Psi}(U)$ commutes.

Claims (2) and (4) are similar with appropriate choices of diagrams. The final statement is immediate since $L(\varphi, \psi) = 0$ implies all diagrams needed for an interleaving commute. $\square$

Lemma 3.14. *Given a basis unnatural transformation*

$$\{\Phi_{U_\sigma} : F(U_\sigma) \to G(U_\sigma^N) \mid \sigma \in K\}$$

with $L_{\diagdown}^{U_\tau, U_\sigma} = 0$ *for all* $\sigma \leq \tau$*, we can extend this to a full natural transformation* Φ*; i.e. we can define* Φ_U *for all* U *such that* $L_{\diagdown}^{U,V} = 0$ *for all* $U \subseteq V$.

Proof of Lem. 3.14. We start by defining Φ_U for arbitrary open sets. Note that since $L_{\diagdown}^{U_\tau, U_\sigma} = 0$, for any $\sigma \leq \tau$, the diagram of the form

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

commutes.

For an arbitrary open U, define the cover $\mathcal{U} = \{U_\sigma \mid \sigma \in U\}$. It is straightforward to check that $U = \bigcup_{U_\sigma \in \mathcal{U}} U_\sigma$ and that any nonempty intersection $U_\sigma \cap U_\tau$ is also an element of $\mathcal{U}$. Then we use the fact that F is a cosheaf, and in particular this means that $F(U)$ is the coequalizer of the diagram

$$\coprod_{\sigma, \sigma'} F(U_\sigma \cap U_{\sigma'}) \underset{F[U_\sigma \cap U_{\sigma'} \subseteq U_{\sigma'}]}{\overset{F[U_\sigma \cap U_{\sigma'} \subseteq U_\sigma]}{\rightrightarrows}} \coprod_\tau F(U_\tau).$$

Rephrased, this means that for any set S with maps $F(U_\sigma) \to S$ such that the solid arrow diagrams of the form

$$\begin{tikzcd}
F(U_\sigma \cap U_{\sigma'}) \arrow[r, "F[\subseteq]"] \arrow[d, "F[\subseteq]"'] & F(U_\sigma) \arrow[d, "F[\subseteq]"] \arrow[ddr, bend left] & \\
F(U_{\sigma'}) \arrow[r, "F[\subseteq]"] \arrow[drr, bend right] & F(U) \arrow[dr, dashed, "\exists!"] & \\
& & S
\end{tikzcd}$$

commute for any σ, σ', then there is a unique map $F(U) \to S$ whose addition still has all diagrams commute. In our case, set $S = G(U^n)$, and define the legs of the cocone to be $G[\subseteq] \circ \Phi_{U_\sigma}$ as seen in the bold purple arrows of the diagram

$$\begin{tikzcd}
F(U_\sigma \cap U_{\sigma'}) \arrow[rr, "\Phi_\bullet"] \arrow[dr, "F[\subseteq]"] \arrow[dd, "F[\subseteq]"'] & & G((U_\sigma \cap U_{\sigma'})^n) \arrow[dr, "G[\subseteq]"] \arrow[dd, "G[\subseteq]"'] & \\
& F(U_{\sigma'}) \arrow[rr, "\Phi_\bullet"] \arrow[dd, "F[\subseteq]"'] & & G(U_{\sigma'}^n) \arrow[dd, "G[\subseteq]"] \\
F(U_\sigma) \arrow[rr, "\Phi_\bullet"] \arrow[dr, "F[\subseteq]"'] & & G(U_{\sigma'}^n) \arrow[dr, "G[\subseteq]"] & \\
& F(U) \arrow[rr, dashed, "\exists!\ \Phi_U"] & & G(U^n)
\end{tikzcd} \tag{6.5}$$

where $\Phi_\bullet$ means Φ_V for the appropriate set V, but is dropped to simplify the notation. Note that the diagram prior to the inclusion of the dotted line commutes, since we can check the relevant faces as follows. The left and right squares commute because F and G are functors. The back and top panels commute because they involve only basis opens; equivalently, because we assumed $L_{\diagdown}^{U_\sigma \cap U_{\sigma'}, U_\sigma} = L_{\diagdown}^{U_\sigma \cap U_{\sigma'}, U_{\sigma'}} = 0$. Then, because $F(U)$ is a colimit of the diagram, there exists a unique map $\Phi_U : F(U) \to G(U)$ as noted, making any diagram of this form commute.

To ensure that the resulting Φ_U maps make diagrams of the form

$$\begin{tikzcd}
F(U) \arrow[r] \arrow[d] & G(U^n) \arrow[d] \\
F(V) \arrow[r] & G(V^n)
\end{tikzcd}$$

commute for arbitrary $U \subseteq V$, fix such a pair and an $x \in F(U)$. Because $F(U)$ is the colimit, there is a σ and an $x_\sigma \in F(U_\sigma)$ such that $x_\sigma \mapsto x$. In this case we have the diagram

$$\begin{tikzcd}
F(U_\sigma) \arrow[r, "\Phi_\bullet"] \arrow[dr] \arrow[ddr] & G(U_\sigma)^n \arrow[dr] \arrow[ddr] & \\
& F(U) \arrow[r, "\Phi_\bullet"] \arrow[d] & G(U^n) \arrow[d] \\
& F(V) \arrow[r, "\Phi_\bullet"] & G(V^n)
\end{tikzcd}$$

The top and back squares commute because they are the front of the cube of the diagram in Eq. (6.5). The left and right triangles commute since F and G are functors. Thus the front square, and in particular the element $x \in F(U)$, commute. This means the resulting Φ is a natural transformation, and thus $L_{\triangledown}^{U,V} = 0$. $\square$

Lemma 3.15. *Given natural transformations $\Phi : F \Rightarrow G^N$ and $\Psi : G^N \Rightarrow F$ such that $L_{\triangledown}^{U_\sigma} = 0$ for all $\sigma \in K$, then $L_{\triangledown}^{U} = 0$ for all open sets U.*

Proof of Lem. 3.15. Because $L_{\triangledown}^{U_\sigma} = 0$ for all basis elements, diagrams of the form

$$\begin{tikzcd}
F(U_\sigma) \arrow[rr, "F[\subseteq]"] \arrow[dr, "\Phi_{U_\sigma}"'] & & F(U_\sigma^{2n}) \\
& G(U_\sigma^n) \arrow[ur, "\Psi_{U_\sigma^n}"'] &
\end{tikzcd}$$

commute for any $\sigma \in K$. Given an arbitrary open set U, let $x \in F(U)$ be given. As in the proof of Lem. 3.14, there is a σ and an $x_\sigma \in F(U_\sigma)$ with $x_\sigma \mapsto x$. Then consider the diagram

$$\begin{tikzcd}
F(U_\sigma) \arrow[rr, "F[\subseteq]"] \arrow[dr, "F[\subseteq]"] \arrow[ddr, "\Phi_\bullet"'] & & F(U_\sigma^{2n}) \arrow[dr, "F[\subseteq]"] & \\
& F(U) \arrow[rr, "F[\subseteq]"] \arrow[ddr, "\Phi_\bullet"] & & F(U^{2n}) \\
& G(U_\sigma^n) \arrow[uur, "\Psi_\bullet"] \arrow[dr, "G[\subseteq]"'] & & \\
& & G(U^n) \arrow[uur, "\Psi_\bullet"'] &
\end{tikzcd}$$

where we again replace map subscripts with $\bullet$ to simplify notation. The top square commutes because F is a functor. The back triangle commutes by this lemma's assumption. The left and right squares commute because Φ and Ψ are natural transformations. Taken together, this means that the diagram commutes and in particular, the image of $x \in F(U)$ chased around the front triangle commutes. $\square$

6.3 Proof from Sec. 5

Proposition 5.3. *For inputs $f : \mathbb{X} \to \mathbb{R}$ and $g : \mathbb{Y} \to \mathbb{R}$, denote the respective Reeb cosheaves as $\widetilde{F}, \widetilde{G}$: $\mathbf{Int} \to \mathbf{Set}$, and the respective mapper cosheaves as $F, G : \mathbf{Open}(K) \to \mathbf{Set}$. Then*

$$d_R(\widetilde{F}, \widetilde{G}) \leq (d_I(F, G) + 1)\,\delta.$$

Proof. Let φ, ψ be an n-interleaving for $F, G : \mathbf{Open}(K) \to \mathbf{Set}$. We will construct an $\varepsilon = \delta(n+1)$-interleaving $\widetilde{\varphi}$, $\widetilde{\psi}$ for $\widetilde{F}, \widetilde{G} : \mathbf{Int} \to \mathbf{Set}$.

We start by defining $\widetilde{\varphi} : \widetilde{F} \Rightarrow \widetilde{G}^{\varepsilon}$ as $\widetilde{\psi}$ is analogous. Given an arbitrary interval $I = (a, b)$, let $J = (j\delta, k\delta)$ be the smallest grid-aligned interval containing I; i.e. $j\delta \leq a < (j+1)\delta$ and $(k - 1\delta) < b \leq k\delta$.

Note that $I \subseteq J \subseteq J^{\delta n} \subseteq I^{(n+1)\delta} = I^{\varepsilon}$. Let $S = \{\tau_i \mid j \leq i \leq k-1\} \cup \{\sigma_i \mid j < i < k\}$. A quick check shows that $S \in \mathbf{Open}(K)$, that $J = |S|$, and that $J^{\delta n} = |S^n|$. Chasing definitions, this means that $\widetilde{F}(J) = \pi_0(f^{-1}(J))$ and $F(S) = \pi_0(f^{-1}(|S|))$ are equal; similarly $\widetilde{F}(J^{\delta n}) = F(S^n)$. Then define $\widetilde{\varphi}_I$ to be the map defined by the composition

$$\begin{tikzcd}
\widetilde{F}(I) \arrow[rr, dashed, "\widetilde{\varphi}_I"] \arrow[d, "\widetilde{F}[\subseteq]"'] & & \widetilde{G}(I^{(n+1)\delta}) \\
\widetilde{F}(J) \arrow[r, dashed, "\widetilde{\varphi}_J"] \arrow[d, "="'] & \widetilde{G}(J^{\delta n}) \arrow[ur, "\widetilde{G}[\subseteq]"'] & \\
F(S) \arrow[r, "\varphi_S"'] & G(S^n). \arrow[u, "="'] &
\end{tikzcd}$$

Notice that setting I to be an axis aligned interval J gives the map $\widetilde{\varphi}_J$ marked.

Now that we have built $\widetilde{\varphi}$ and $\widetilde{\psi}$, we need to check (i) that each is a natural transformation and (ii) that they satisfy the triangle diagrams of Defn. 5.2. For (i) we check only $\widetilde{\varphi}$ as, again, $\widetilde{\psi}$ is symmetric. To this end, assume we have $I \subseteq I'$ with minimal grid-aligned intervals J and J', and let $S, S' \in \mathbf{Open}(K)$ be such that $|S| = J$ and $|S'| = J'$. Then consider the diagram

$$\begin{tikzcd}
\widetilde{F}(I) \arrow[rr, "\widetilde{\varphi}_I"] \arrow[d, "\widetilde{F}[\subseteq]"'] \arrow[dddr, "\widetilde{F}[\subseteq]"] & & \widetilde{G}(I^{(n+1)\delta}) \arrow[dddr, "\widetilde{G}[\subseteq]"] & \\
\widetilde{F}(J) \arrow[d, "="'] & & \widetilde{G}(J^{\delta n}) \arrow[u, "\widetilde{G}[\subseteq]"] & \\
F(S) \arrow[rr, "\varphi_S"'] \arrow[dddr, "F[\subseteq]"'] & & G(S^n) \arrow[u, "="'] \arrow[dddr, "G[\subseteq]"'] & \\
& \widetilde{F}(I') \arrow[rr, "\widetilde{\varphi}_{I'}"] \arrow[d, "\widetilde{F}[\subseteq]"'] & & \widetilde{G}((I')^{(n+1)\delta}) \\
& \widetilde{F}(J') \arrow[d, "="] & & \widetilde{G}((J')^{\delta n}) \arrow[u, "\widetilde{G}[\subseteq]"'] \\
& F(S') \arrow[rr, "\varphi_{S'}"'] & & G((S')^n). \arrow[u, "="']
\end{tikzcd}$$

Note that the front and back panels of the cube are the diagrams that were used to define $\widetilde{\varphi}_I$ and $\widetilde{\varphi}_{I'}$, so they commute. The bottom panel commutes because φ is a natural transformation. The left and right panels commute because F and $\widetilde{F}$ arise from computing connected components on the same underling input data. Thus, the top square commutes, and this is exactly what is needed to say that $\widetilde{\varphi}$ is a natural transformation.

To check (ii), fix an interval I with grid aligned $J \subseteq I$ and $S \in \mathbf{Open}(K)$ with $|S| = J$. Then consider the diagram

$$\begin{tikzcd}
\widetilde{F}(I) \arrow[rr, "\widetilde{\varphi}_I"] \arrow[d, "\widetilde{F}[\subseteq]"'] & & \widetilde{G}(I^{(n+1)\delta}) \arrow[rr, "\widetilde{\psi}_{I^\varepsilon}"] \arrow[d] & & \widetilde{F}(I^{2(n+1)\delta}) \\
\widetilde{F}(J) \arrow[d, "="'] & \widetilde{G}(J^{\delta n}) \arrow[ur, "\widetilde{G}[\subseteq]"] \arrow[r, "\widetilde{G}[\subseteq]"'] & \widetilde{G}(J^{(n+1)\delta})) \arrow[d, "="'] & \widetilde{F}(J^{\delta(2n+1)}) \arrow[ur, "\widetilde{F}[\subseteq]"'] & \\
F(S) \arrow[r, "\varphi_S"] \arrow[drr, "{F[\subseteq]}"'] & G(S^n) \arrow[u, "="'] \arrow[r, "{G[\subseteq]}"] \arrow[dr, "\psi_{S^n}"] & G(S^{n+1}) \arrow[r, "\psi_{S^{n+1}}"] & F(S^{2n+1}) \arrow[u, "="'] & \\
& & F(S^{2n}). \arrow[ur, "{F[\subseteq]}"'] & &
\end{tikzcd}$$

The left and right hexa-laterals commute by definition of $\widetilde{\varphi}$ and $\widetilde{\psi}$ respectively. The middle top triangle commutes because $\widetilde{G}$ is a functor, and the middle square commutes because $\widetilde{G}$ and G are defined as connected components of the same input data. The bottom left triangle commutes because φ and ψ are an n-interleaving. The right quadrilateral commutes because ψ is a natural transformation. All this shows that the outside boundary of the diagram commutes. Swapping out the interior, we have

$$\begin{tikzcd}
& \widetilde{G}(I^{(n+1)\delta}) \arrow[dr, "\widetilde{\psi}_{I^\varepsilon}"] & \\
\widetilde{F}(I) \arrow[ur, "\widetilde{\varphi}_I"] \arrow[rr, "\widetilde{F}[\subseteq]"] \arrow[d, "\widetilde{F}[\subseteq]"'] & & \widetilde{F}(I^{2(n+1)\delta}) \\
\widetilde{F}(J) \arrow[rr, "\widetilde{F}[\subseteq]"] \arrow[d, "="'] & & \widetilde{F}(J^{\delta(2n+1)}) \arrow[u, "\widetilde{F}[\subseteq]"'] \\
F(S) \arrow[rr, "{F[\subseteq]}"] \arrow[dr, "{F[\subseteq]}"'] & & F(S^{2n+1}) \arrow[u, "="'] \\
& F(S^{2n}). \arrow[ur, "{F[\subseteq]}"'] &
\end{tikzcd}$$

The bottom triangle commutes because F is a functor, the next square up commutes by definition of F and $\widetilde{F}$, and the top square commutes because $\widetilde{F}$ is a functor. Combining this with the outside ring commuting means that the top triangle commutes, which is the final ingredient needed for the definition of an interleaving. □

7 Discussion

In this paper, we define a loss function that quantifies how far a diagram is from being commutative, and use such a loss function to bound the interleaving distance, both for mapper and Reeb graph settings. This work provides a way to evaluate a particular set of maps, which immediately suggests the question of utilizing this quantification to iteratively improve our comparison. Here, the quality of the bound is dependent on the quality of the input n-assignment, but we assume no control over that input in this paper and so we cannot evaluate the tightness of the bound. In the long term, we envision this bound to be used in the context of a gradient descent style framework, where an input n-assignment can be improved incrementally thus finding a better bound on the distance. Of course, we know that deciding if two Reeb graphs are ϵ-interleaved (for $\epsilon \geq 1$) is NP-hard [2], so our gradient decent has no guarantee of reaching the global optimal solution. However, the potential for not only getting better approximations but also returning the actual interleaving maps used in the bound is an exciting step toward computing interleaving distances for graph-based signatures available in practice. Furthermore, the current approach focuses on 0-dimensional interleavings involving connected components, it is possible to extend our framework in the future to study 1-dimensional interleavings by studying homologous cycles.

We also believe that our loss function based framework is applicable in a broader context where data are modeled as sheaves or cosheaves in the category of sets, as sheaf theory is emerging as a tool in data science to study, e.g., distributed systems [25, 26], sensor networks [31], model fit [23], and uncertainty quantification [20]. In particular, one interesting next step is to study how to extend our framework to work with persistence modules as cosheaves in the category of vector spaces (e.g., [7]). As the interleaving distance for multiparameter persistence modules is similarly NP-hard [3], this would be an exciting step toward computational efforts in this broad class of topological signatures.

References

[1] Jonathan A. Barmak. *Algebraic Topology of Finite Topological Spaces and Applications.* Berlin, Heidelberg: Springer, 2011. DOI: 10.1007/978-3-642-22003-6.

[2] Håvard Bakke Bjerkevik and Magnus Bakke Botnan. "Computational Complexity of the Interleaving Distance". In: *34th International Symposium on Computational Geometry (SoCG 2018).* Ed. by Bettina Speckmann and Csaba D. Tóth. Vol. 99. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018, 13:1–13:15. ISBN: 978-3-95977-066-8. DOI: 10.4230/LIPIcs.SoCG.2018.13.

[3] Håvard Bakke Bjerkevik, Magnus Bakke Botnan, and Michael Kerber. "Computing the Interleaving Distance is NP-Hard". In: *Foundations of Computational Mathematics* 20.5 (Nov. 2019), pp. 1237–1271. DOI: 10.1007/s10208-019-09442-y.

[4] Brian Bollen, Erin Wolf Chambers, Joshua Levine, and Elizabeth Munch. "Reeb Graph Metrics from the Ground Up". In: (2021). arXiv: 2110.05631 [cs.CG].

[5] Magnus Bakke Botnan, Justin Curry, and Elizabeth Munch. "A Relative Theory of Interleavings". In: (2020). arXiv: 2004.14286 [math.CT].

[6] Adam Brown, Omer Bobrowski, Elizabeth Munch, and Bei Wang. "Probabilistic convergence and stability of random mapper graphs". In: *Journal of Applied and Computational Topology* (Dec. 2020). DOI: 10.1007/s41468-020-00063-x.

[7] Peter Bubenik and Nikola Milićević. "Homological Algebra for Persistence Modules". In: *Foundations of Computational Mathematics* 21 (2021), pp. 1233–1278. DOI: 10.1007/s10208-020-09482-9.

[8] Peter Bubenik, Vin de Silva, and Jonathan Scott. "Metrics for Generalized Persistence Modules". In: *Foundations of Computational Mathematics* 15.6 (Oct. 2014), pp. 1501–1531. DOI: 10.1007/s10208-014-9229-5.

[9] Maike Buchin, Erin Chambers, Pan Fang, Brittany Terese Fasy, Ellen Gasparovic, Elizabeth Munch, and Carola Wenk. "Distances Between Immersed Graphs: Metric Properties". In: *La Matematica* 2.1 (Jan. 2023), pp. 197–222. DOI: 10.1007/s44007-022-00037-8.

[10] Mathieu Carrière, Bertrand Michel, and Steve Oudot. "Statistical Analysis and Parameter Selection for Mapper". In: *Journal of Machine Learning Research* 19.12 (2018), pp. 1–39.

[11] Mathieu Carrière and Steve Oudot. "Structure and Stability of the One-Dimensional Mapper". In: *Foundations of Computational Mathematics* 18.6 (Oct. 2017), pp. 1333–1396. DOI: 10.1007/s10208-017-9370-z.

[12] Frédéric Chazal, David Cohen-Steiner, Marc Glisse, Leonidas J. Guibas, and Steve Y. Oudot. "Proximity of persistence modules and their diagrams". In: *Proceedings of the 25th annual symposium on Computational geometry.* SCG '09. Aarhus, Denmark: ACM, 2009, pp. 237–246. ISBN: 978-1-60558-501-7. DOI: 10.1145/1542362.1542407.

[13] Aura Conci and Carlos Kubrusly. "Distances between sets—a survey". In: *Advances in Mathematical Sciences and Applications* 26.1 (2017), pp. 1–18. ISSN: 1343-4373.

[14] Justin Curry. "Sheaves, Cosheaves and Applications". PhD thesis. University of Pennsylvania, 2014.

[15] Vin de Silva, Elizabeth Munch, and Anastasios Stefanou. "Theory of interleavings on categories with a flow". In: *Theory and Applications of Categories* 33.21 (2018), pp. 583–607.

[16] Michel Marie Deza and Elena Deza. *Encyclopedia of Distances*. second. Heidelberg New York Dordrecht London: Springer Berlin, Heidelberg, 2013. DOI: 10.1007/978-3-642-30958-8.

[17] Claire Donnat and Susan Holmes. "Tracking network dynamics: A survey using graph distances". In: *The Annals of Applied Statistics* 12.2 (June 2018). DOI: 10.1214/18-aoas1176.

[18] Elena Farahbakhsh Touli and Yusu Wang. "FPT-Algorithms for Computing Gromov-Hausdorff and Interleaving Distances Between Trees". In: *27th Annual European Symposium on Algorithms (ESA 2019)*. Ed. by Michael A. Bender, Ola Svensson, and Grzegorz Herman. Vol. 144. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019, 83:1–83:14. ISBN: 978-3-95977-124-5. DOI: 10.4230/LIPIcs.ESA.2019.83.

[19] Ellen Gasparovic, Elizabeth Munch, Steve Oudot, Katharine Turner, Bei Wang, and Yusu Wang. "Intrinsic Interleaving Distance for Merge Trees". In: (July 2019). arXiv: 1908.00063 [cs.CG].

[20] Cliff A. Joslyn, Lauren Charles, Chris DePerno, Nicholas Gould, Kathleen Nowak, Brenda Praggastis, Emilie Purvine, Michael Robinson, Jennifer Strules, and Paul Whitney. "A Sheaf Theoretical Approach to Uncertainty Quantification of Heterogeneous Geolocation Information". In: *Sensors* 20.12 (2020). ISSN: 1424-8220. DOI: 10.3390/s20123418.

[21] Tomasz Kaczynski, Konstantin Mischaikow, and Marian Mrozek. *Computational Homology*. New York, NY, USA: Springer, Jan. 9, 2004. 500 pp. ISBN: 0387408533. DOI: 10.1007/b97315.

[22] Woojin Kim, Facundo Mémoli, and Anastasios Stefanou. "Interleaving by Parts: Join Decompositions of Interleavings and Join-Assemblage of Geodesics". In: *Order* (Sept. 2023). DOI: 10.1007/s11083-023-09643-9.

[23] Henry Kvinge, Brett Jefferson, Cliff Joslyn, and Emilie Purvine. "Sheaves as a Framework for Understanding and Interpreting Model Fit". In: *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)* (2021). DOI: 10.1109/ICCVW54120.2021.00469.

[24] Michael Lesnick. "The Theory of the Interleaving Distance on Multidimensional Persistence Modules". English. In: *Foundations of Computational Mathematics* 15.3 (2015), pp. 613–650. ISSN: 1615-3375. DOI: 10.1007/s10208-015-9255-y.

[25] Grant Malcolm. "Sheaves, Objects, and Distributed Systems". In: *Electronic Notes in Theoretical Computer Science* 225.2 (2009), pp. 3–19.

[26] Seyed Mansourbeigi. "Sheaf Theory Approach to Distributed Applications: Analysing Heterogeneous Data in Air Traffic Monitoring". In: *International Journal of Data Science and Analysis* 3.5 (2017), pp. 34–39. DOI: 10.11648/j.ijdsa.20170305.11.

[27] Elizabeth Munch and Anastasios Stefanou. "The ℓ^∞-Cophenetic Metric for Phylogenetic Trees as an Interleaving Distance". In: (Feb. 2018). arXiv: 1803.07609v1 [cs.CG].

[28] Elizabeth Munch and Bei Wang. "Convergence between Categorical Representations of Reeb Space and Mapper". In: *32nd International Symposium on Computational Geometry (SoCG 2016)*. Ed. by Sándor Fekete and Anna Lubiw. Vol. 51. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2016, 53:1–53:16. ISBN: 978-3-95977-009-5. DOI: 10.4230/LIPIcs.SoCG.2016.53.

[29] nLab authors. *unnatural transformation*. https://ncatlab.org/nlab/show/unnatural+transformation. Revision 1. Nov. 2023.

[30] Emily Riehl. *Category theory in context*. New York, NY, USA: Courier Dover Publications, 2017.

[31] Michael Robinson. "Sheaves are the canonical data structure for sensor integration". In: *Information Fusion* 36 (2017), pp. 208–224. DOI: 10.1016/j.inffus.2016.12.002.

[32] Michael Robinson. "Assignments to sheaves of pseudometric spaces". In: *Compositionality* 2 (2 June 2020). ISSN: 2631-4444. DOI: 10.32408/compositionality-2-2.

[33] Vin de Silva, Elizabeth Munch, and Amit Patel. "Categorified Reeb Graphs". In: *Discrete & Computational Geometry* 55.4 (Apr. 2016), pp. 854–906. ISSN: 1432-0444. DOI: 10.1007/s00454-016-9763-9.

[34] Gurjeet Singh, Facundo Mémoli, and Gunnar Carlsson. "Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition". In: *Eurographics Symposium on Point-Based Graphics*. 2007. DOI: 10.2312/SPBG/SPBG07/091-100.

[35] Peter Wills and François G. Meyer. "Metrics for graph comparison: A practitioner's guide". In: *PLOS ONE* 15.2 (Feb. 2020). Ed. by Pin-Yu Chen, e0228728. DOI: 10.1371/journal.pone.0228728.

[36] Lin Yan, Talha Bin Masood, Raghavendra Sridharamurthy, Farhan Rasheed, Vijay Natarajan, Ingrid Hotz, and Bei Wang. "Scalar Field Comparison with Topological Descriptors: Properties and Applications for Scientific Visualization". In: *Computer Graphics Forum (CGF)* 40.3 (2021), pp. 599–633. DOI: 10.1111/cgf.14331.