A Family of Metrics from the Truncated Smoothing of Reeb Graphs

Erin Wolf Chambers
Department of Computer Science, St. Louis University, MO, USA

Elizabeth Munch
Department of Computational Mathematics,
Science and Engineering and Department of Mathematics,
Michigan State University, East Lansing, MI, USA

Tim Ophelders
Department of Mathematics and Computer Science, TU Eindhoven, The Netherlands

Abstract

In this paper, we introduce an extension of smoothing on Reeb graphs, which we call truncated smoothing; this in turn allows us to define a new family of metrics which generalize the interleaving distance for Reeb graphs. Intuitively, we "chop off" parts near local minima and maxima during the course of smoothing, where the amount cut is controlled by a parameter τ. After formalizing truncation as a functor, we show that when applied after the smoothing functor, this prevents extensive expansion of the range of the function, and yields particularly nice properties (such as maintaining connectivity) when combined with smoothing for $0 \leq \tau \leq 2\varepsilon$, where ε is the smoothing parameter. Then, for the restriction of $\tau \in [0, \varepsilon]$, we have additional structure which we can take advantage of to construct a categorical flow for any choice of slope $m \in [0, 1]$. Using the infrastructure built for a category with a flow, this then gives an interleaving distance for every $m \in [0, 1]$, which is a generalization of the original interleaving distance, which is the case $m = 0$. While the resulting metrics are not stable, we show that any pair of these for $m, m' \in [0, 1)$ are strongly equivalent metrics, which in turn gives stability of each metric up to a multiplicative constant. We conclude by discussing implications of this metric within the broader family of metrics for Reeb graphs.

2012 ACM Subject Classification Mathematics of computing → Algebraic topology; Theory of computation → Computational geometry

Keywords and phrases Reeb graphs, interleaving distance, graphical signatures

Digital Object Identifier 10.4230/LIPIcs.SoCG.2021.22

Related Version *Full Version*: https://arxiv.org/abs/2007.07795v3

Funding *Erin Wolf Chambers*: This work was funded in part by the National Science Foundation through grants CCF-1907612 and DBI-1759807.
Elizabeth Munch: This work was funded in part by the National Science Foundation through grants CCF-1907591 and DEB-1904267.
Tim Ophelders: This work was funded in part by the National Science Foundation through grant CCF-1907591.

Acknowledgements The authors wish to thank the anonymous reviewers for their helpful feedback and insights that tightened the bounds of Theorem 5.3.

1 Introduction

The Reeb graph, originally defined in the context of Morse theory [44], represents a portion of the underlying structure of a topological space $\mathbb{X}$ through the lens of a real valued function $h : \mathbb{X} \to \mathbb{R}$; the pair of data $(\mathbb{X}, h)$ is known as an $\mathbb{R}$-space. Specifically, points in the Reeb graph correspond to connected components in the levelsets of the function; as such, the Reeb graph inherits a real valued function from the original input data. For nice enough

37th International Symposium on Computational Geometry (SoCG 2021).
Editors: Kevin Buchin and Éric Colin de Verdière; Article No. 22; pp. 22:1–22:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

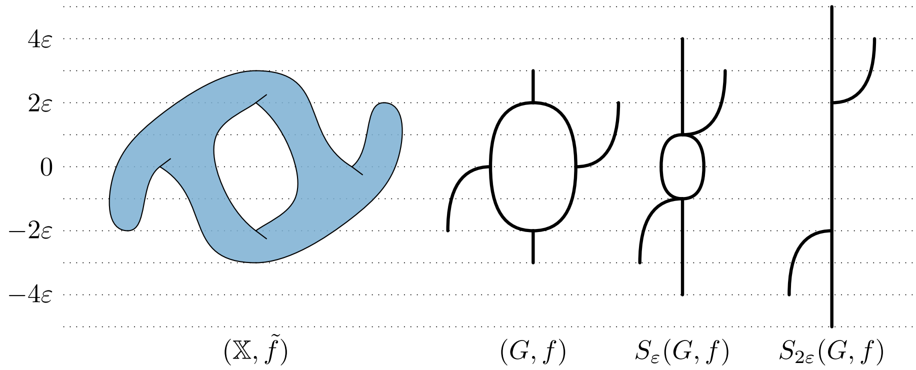


Figure 1 From left to right: an $\mathbb{R}$-space $(\mathbb{X}, \tilde{f})$, its Reeb graph (G, f), smoothings are shown for two parameters, ε and 2ε. Function values are shown by height.

inputs, the resulting object is a finite graph. So, at its core, we focus our study on objects of the form (G, f) where G is a graph and $f : G \to \mathbb{R}$ is a function given on vertices and interpolated linearly on the edges. See Figure 1 for an example.

Reeb graphs have become increasingly useful in a wide range of applications, including settings such as shape comparison [35, 29], denoising [52], shape understanding [25, 34], reconstructing non-linear 1-dimensional structure in data [42, 32, 19, 50], summarizing collections of trajectory data [15], and allowing for informed exploration of otherwise hard-to-visualize high-dimensional data [51, 33]; see [5] for a survey of these and more topics. As a result, there is interest in defining metrics on these objects, to evaluate their quality in the face of noisy input data as well as to allow for more accurate shape comparison and analysis. In this setting, we are focused on metrics that incorporate both the graph and function information: so $d((G, f_1), (G, f_2))$ should be non-zero if $f_1 \neq f_2$ even though they are defined on the same underlying graphs.

Several metrics have arisen recently to do this, taking inspiration from different mathematical backgrounds [23, 4, 3, 16, 27, 28, 1, 47, 2]. In this paper, we focus on the Reeb graph interleaving distance [23]. The basic idea is to work with a notion of smoothing, which returns a parameterized family of Reeb graphs, $S_\varepsilon(G, f)$ for every $\varepsilon \geq 0$, starting with $\varepsilon = 0$ which leaves the input unchanged. This procedure simplifies the loop structures and stretches tails [54]; see Figure 1 for an example. Then the goal is to find an ε-interleaving, which is a pair of families of maps making a particular diagram commute. If $\varepsilon = 0$, this diagram simplifies down to finding an isomorphism between the two Reeb graphs; increasing ε provides more flexibility to find such pairs of maps. Then we have a metric by defining $d_I((G, f), (H_h))$ to be the infimum over the set of ε for which such a diagram exists.

This metric takes root in the interleaving distance defined for persistence modules [18], and is largely inspired by the subsequent category theoretic treatment [14, 12]. This viewpoint comes from encoding the data of a Reeb graph in a constructible set-valued cosheaf [21, 22]. It was later shown that these metrics are special cases of a more general theory of interleaving distances given on a *category with a flow* [24, 48, 20]. This framework encompases common metrics including ℓ_∞ distance on points or functions, regular Hausdorff distance, and the Gromov-Hausdorff distance [48, 13]. Using this framework, interleaving metrics have been studied in the context of $\mathbb{R}$-spaces [8], multiparameter persistence modules [38], merge trees [39], and formigrams [36, 37], and on more general category theoretic constructions [10, 45]. There are also interesting restrictions to labeled merge trees, where one can pass to a matrix representation and show that the interleaving distance is equivalent to the point-wise ℓ_∞ distance [40, 31, 53, 49].

On the negative side, it has been shown that Reeb graph interleaving is graph isomorphism complete [23, 6], and that many other variants are also NP-hard [6, 7]. All of this means that these metrics, while mathematically interesting, may not lead to feasible algorithms for comparison and analysis. However, a glimmer of hope arises with work investigating fixed parameter tractable algorithms [30, 49]. Despite the issues of computational complexity, notions of similarity for graphs in general, and Reeb graphs in particular, are of pressing interest due to their extensive use in data analysis; in many such settings, we are concerned with questions of quality in the face of noise, and understanding convergence of approximations to a true underlying structure. For example, the interleaving distance has been used in evaluating the quality of the mapper graph [46], which can be proven to be a approximation of the Reeb graph using this metric [41, 11]. Furthermore, there is considerable interest in unifying the interleaving distance with the emerging collection of other Reeb graph metrics.

In this paper, we introduce a truncation operation, which intuitively cuts off portions of the Reeb graph near local extrema with respect to f; this operation is easy to compute for any Reeb graph and tends to result in a simplified Reeb graph. We show that truncation is a functor, and when combined with the smoothing functor, defines a flow on the the category of Reeb graphs. We investigate and prove particularly desirable geometric and topological properties of *truncated smoothing* for certain ranges of the two parameters controlling the functors. We then introduce a new family of metrics for Reeb graphs, called truncated interleaving distances. They are parameterized by $m \in [0, 1]$, and generalize the interleaving distance, with the setting $m = 0$ being the original interleaving distance. We show that the metrics arising from $m \in [0, 1)$ are strongly equivalent. Although the metrics are not stable in the sense of [3], strong equivalence implies that they are at least stable up to a constant.

When combined with preliminary work on geometric implications of smoothing [54], truncated smoothing is interesting in its own right, as it provides a collection of paths for Reeb graph space to be studied in terms of the resulting persistence diagrams. It also is useful when considering algorithms to test planarity for Reeb graphs, or find planar representations of them. The new family of metrics also provide the possibility for new approaches for approximation algorithms for the interleaving distance, as well as new avenues for further unification of the broader family of Reeb graph metrics.

Outline. We give the basic background on Reeb graphs, smoothing, and the Reeb graph interleaving distance in Section 2. Next, we introduce our definition of truncated smoothing in Section 3. In Section 4 we check properties of the truncated smoothing operation. We then take a categorical view of truncated smoothing to develop a family of metrics and investigate their properties in Sections 5 and 6. Finally, the results are discussed in Section 7. Note that many proofs and technical details, as well as full background, several equivalent alternative formulations, and more examples are included in the full version of the paper [17].

2 Background: Reeb graphs, smoothing, and interleaving

Given a topological space $\mathbb{X}$ along with a continuous $\mathbb{R}$-valued function $f\colon \mathbb{X} \to \mathbb{R}$, we call the pair $(\mathbb{X}, f)$ an $\mathbb{R}$-*space*. For two $\mathbb{R}$-spaces $(\mathbb{X}, f)$ and $(\mathbb{X}', g)$, we call a continuous map $\varphi\colon \mathbb{X} \to \mathbb{X}'$ *function-preserving* if $f = g \circ \varphi$, and write $\varphi\colon (\mathbb{X}, f) \to (\mathbb{X}', g)$ in that case.

For an $\mathbb{R}$-space $(\mathbb{X}, f)$, we define an equivalence relation $\sim_f$ on the points of $\mathbb{X}$, such that $x \sim_f x'$ if and only if x and x' lie in the same path-connected component of $f^{-1}(y)$ for some $y \in \mathbb{R}$. For sufficiently nice functions[1], the quotient space $\mathbb{X}/\sim_f$ is a graph,

[1] e.g. a Morse function on a manifold, or a constructible space and function [23], or a space with a levelset-tame function [26].

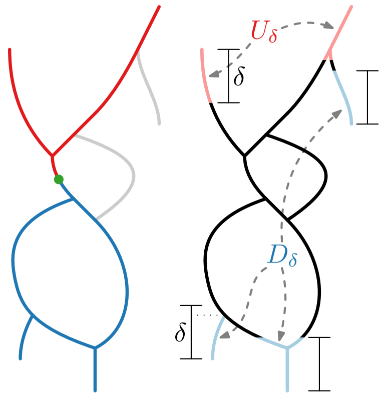


Figure 2 Left: the up-set (red) and down-set (blue) of a point. Although the up-set is a tree, it is not an up-tree as it contains down-forks of the ambient graph. Right: the sets U_δ and D_δ of points with no length δ up-path or down-path, respectively. The leftmost component of D_δ does not contain the down-fork.

called a *Reeb graph*, and we denote the quotient map by $q_f\colon (\mathbb{X}, f) \to (\mathbb{X}/{\sim_f}, g)$. Since $f(x) = f(x')$ whenever $x \sim_f x'$, we can treat the Reeb graph as an $\mathbb{R}$-space $(X/{\sim_f}, g)$ by defining $g(q_f(x)) = f(x)$, so that q_f is function-preserving. Most but not all functions in this paper are function preserving. Figure 1 illustrates the construction of a Reeb graph of an $\mathbb{R}$-space.

For the purposes of this work, we will largely divorce the idea of the Reeb graph from the need for a starting space that was used to construct it. Thus for our purposes, a *Reeb graph* is a pair (G, f) where $G = (V_G, E_G)$ is a finite multigraph and $f\colon G \to \mathbb{R}$, referred to as the *height function*, is a continuous map that is linearly interpolated along edges of G, and for which no two neighboring vertices have the same function value. We write $\mathrm{Im}(G, f) := f(G) \subset \mathbb{R}$ for the image of the graph in $\mathbb{R}$. The function can equivalently be stored by defining $f\colon V_G \to \mathbb{R}$ as a function on the vertices, and extending it to the edges implicitly. We treat G as a topological space, so that a point $x \in G$ lies either on a vertex of G, or interior to an edge of G. For succinctness, we also write $x \in (G, f)$ to mean $x \in G$. Since no two adjacent vertices have the same function value, a level set $f^{-1}(y)$ for $y \in \mathbb{R}$ is a finite set of points in G which could be vertices and/or points in the interior edges.

Together, the collection of Reeb graphs (treated as $\mathbb{R}$-spaces) with function-preserving maps as morphisms forms a category, **Reeb**. For the reader without a background in category theory, the basic idea is that that this collection of objects and morphisms satisfy some basic axiomatic structures that make their analysis easier to view as a collection. It also makes available the viewpoint of *functors* between categories, which are essentially structure preserving maps. For now, we will largely hand-wave past the categorical constructions, and defer the technicalities to the full version of the paper.

Define a *path* from x to x' in (G, f) to be a continuous map $\pi\colon [0, 1] \to G$ such that $\pi(0) = x$ and $\pi(1) = x'$. A path is called an *up-path* if it is monotone-increasing with respect to the function, i.e. $f(\pi(t)) \leq f(\pi(t'))$ for $t \leq t'$. Symmetrically, a path is a *down-path* if it is monotone-decreasing. In the case of an up- or down-path π, we call $|f(\pi(0)) - f(\pi(1))|$ the *height* of the path.

In a Reeb graph (G, f), let the *up-paths* of a point x be the set of f-monotone paths that have x as minimum. The *up-set* of a point x is the set of points reachable from x by an up-path, including x itself. Define an *up-fork* to be a vertex x whose up-set contains at least

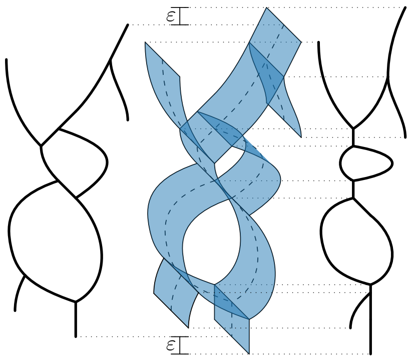


Figure 3 From left to right: a Reeb graph (G, f), its ε-thickening $(G \times [-\varepsilon, \varepsilon], f + \mathrm{Id})$, and the Reeb graph $S_\varepsilon(G, f)$ of the ε-thickening. The product of an edge with an interval is drawn to reflect the function value at a given height.

two edges adjacent to x. We define *down-paths*, *down-sets*, and *down-forks* symmetrically. Call the up-set of a point x an *up-tree* if it contains no down-forks of (G, f), and say that x *roots* an up-tree in such case. The concept of rooting a *down-tree* is defined symmetrically. See Figure 2.

▶ **Definition 2.1.** *Fix a Reeb graph (G, f) and $\varepsilon \geq 0$. Define the* ε-thickening *of G to be the space $G \times [-\varepsilon, \varepsilon]$ with the product topology, and define $(f + \mathrm{Id})\colon G \times [-\varepsilon, \varepsilon] \to \mathbb{R}$ by $(f + \mathrm{Id})(x, t) = f(x) + t$. We define the* ε-smoothing *$S_\varepsilon(G, f)$ to be the Reeb graph of $(f + \mathrm{Id})$, and denote the corresponding quotient map by $q\colon G \times [-\varepsilon, \varepsilon] \to S_\varepsilon(G, f)$. The composition of q with the the inclusion $G \hookrightarrow G \times [-\varepsilon, \varepsilon]; x \mapsto (x, 0)$ is denoted $\eta = q \circ (\mathrm{Id}, 0)$.*

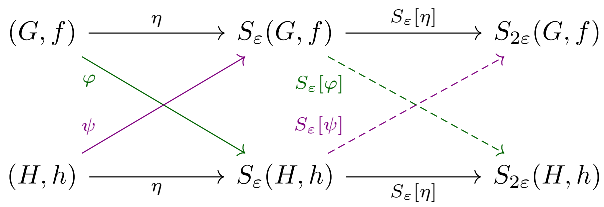
See Figure 3 for an example. In essence, smoothing eliminates small cycles whose height is $\leq 2\varepsilon$, and shrinks all other cycles; it also moves every up-fork and local maximum up and every down-fork and local minimum down. Under the lens of studying the topology of the graph (and in turn the original space), this serves as a functor that can be used to remove noise and simplify topology in a parameterized fashion.

The smoothing construction, S_ε, holds quite a bit more useful structure as not only is it a functor, it is an example of a flow [24]. While we do not provide the full definition here, the specifics are given in the full version of the paper. In particular, this comes from using the additional structure afforded by the function preserving map $\eta\colon (G, f) \to S_\varepsilon(G, f)$. We will reserve the full investigation of η until the full version of the paper, but will use the following property of categories with a flow.

▶ **Theorem 2.2** ([24, Thm. 2.7])**.** *A category with a flow gives rise to an interleaving distance on the objects of the category; specifically, this construction is an extended pseudometric.*

This construction is quite useful since simply by finding some relatively easy to check structure on a category, we immediately get a distance measure on the objects. Depending on the category and flow, this construction encompasses many standard metrics such as the Hausdorff distance; and with a choice of other categories and flows we can construct new metrics. We are particularly interested in the special case of the interleaving distance for Reeb graphs as studied in [23].

▶ **Definition 2.3.** *An* ε-interleaving *with respect to* S_ε *is a pair of maps,* $\varphi\colon (G,f) \to S_\varepsilon(H,h)$ *and* $\psi\colon (H,h) \to S_\varepsilon(G,f)$ *such that the diagram*



commutes. The interleaving distance *is defined to be*

$$d_I((G,f),(H,h)) = \inf_\varepsilon\{\textit{there exists an } \varepsilon\textit{-interleaving of } (G,f) \textit{ and } (H,h)\}.$$

In the construction on this category, d_I is an extended metric since the interleaving distance between Reeb graphs with different numbers of connected components is ∞ as there is no interleaving available for any ε [23]. One particularly useful property we will make use of is understanding how the image of the smoothed Reeb graph, $\mathrm{Im}(S_\varepsilon(G,f)) := f(G) \subseteq \mathbb{R}$, changes under smoothing. Note that if G is connected, $\mathrm{Im}(G,f)$ is connected so it is an interval.

▶ **Proposition 2.4.** *For a connected Reeb graph* (G,f) *with* $\mathrm{Im}(G,f) = [a,b]$*,*

$$\mathrm{Im}(S_\varepsilon(G,f)) = [a-\varepsilon, b+\varepsilon].$$

Proof. For any $c \in \mathrm{Im}(S_\varepsilon(G,f))$, we show that $c \in [a-\varepsilon, b+\varepsilon]$. There is some $x \in S_\varepsilon(G,f)$ with $f_\varepsilon(x) = c$, where f_ε is the induced function on $S_\varepsilon(G,f)$. Then there is a $(y,t) \in G\times[-\varepsilon,\varepsilon]$ with $f(y)+t = c$. Combining $a \le f(y) \le b$ and $-\varepsilon \le t \le \varepsilon$ gives that $a-\varepsilon \le c \le b+\varepsilon$.

For the other direction, let $c \in [a-\varepsilon, b+\varepsilon]$. There exists some $d \in [a,b]$ with $c-d \in [-\varepsilon,\varepsilon]$. Because $\mathrm{Im}(G,f) = [a,b]$, there exists some $x \in f^{-1}(d)$ and $(x, c-d) \in G\times[-\varepsilon,\varepsilon]$ quotients to some $y \in S_\varepsilon(G,f)$ with $f_\varepsilon(y) = c$, so $\mathrm{Im}(S_\varepsilon(G,f)) = [a-\varepsilon, b+\varepsilon]$. ◀

3 Truncated smoothing

We can now introduce our new, modified smoothing of Reeb graphs. Notice from Proposition 2.4 that as the Reeb graph is smoothed, the image becomes larger. The basic idea of truncated smoothing is to cut off some of those expanding tails in a well-defined way.

Let $U_\tau(G,f)$ be the set of points of G that do not have a length τ up-path, and define $D_\tau(G,f)$ symmetrically for down-paths. Note that for any point $x \in U_\tau(G,f)$, all up-paths from x also lie in $U_\tau(G,f)$; the symmetric property is true for $D_\tau(G,f)$. Both $U_\tau(G,f)$ and $D_\tau(G,f)$ are open subsets of (G,f). See Figure 2 for an example. With this, we can define truncation as follows.

▶ **Definition 3.1.** *The* τ-truncation *of* (G,f)*, is the subgraph of* (G,f) *consisting of the points that have both an up-path and a down-path of height* τ*; specifically*

$$T^\tau(G,f) := (G,f) \setminus (U_\tau(G,f) \cup D_\tau(G,f)).$$

This operation can be seen in the second and third graphs of Figure 4. Notice that $T^0(G,f) = (G,f)$, and that for large enough τ, it is entirely possible to disconnect the graph, or even to be left with an empty graph. Utilizing the truncation operation in conjunction with the Reeb graph smoothing operation is what we call truncated smoothing.

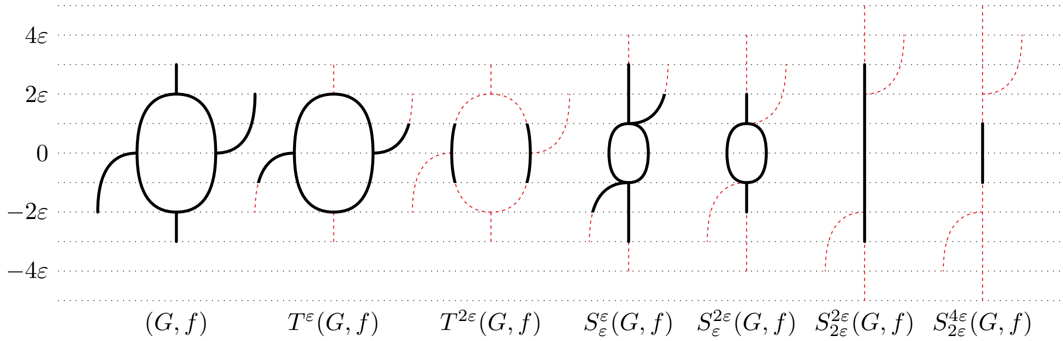


Figure 4 Example of smoothing and truncating for a range of values, on the graph from Figure 1.

▶ **Definition 3.2.** *Let* (G, f)*,* $\varepsilon \geq 0$ *and* $\tau \geq 0$ *be given. Then the* truncated smoothing *of* (G, f) *is defined by* $S^\tau_\varepsilon(G, f) = T^\tau S_\varepsilon(G, f)$*.*

If $\tau = 0$, $S^0_\varepsilon(G, f) = T^0(S_\varepsilon(G, f)) = S_\varepsilon(G, f)$. So S^0_ε is the same as S_ε, and thus the truncated smoothing can be thought of as a generalization of the smoothing definition.

Consider Figure 4, which shows why we smooth before truncating and more generally, why we will soon want to place restrictions on the relationship between τ and ε. Namely, for this example, we have drawn $T^\varepsilon(G, f)$ and $T^{2\varepsilon}(G, f)$. In the second case in particular, it is clear that truncation has massive detrimental effects on the topology as evidenced by the fact that $T^{2\varepsilon}(G, f)$ has two connected components. However, we can avoid these issues when we smooth first. In the last four examples, smoothing serves to move cycles away from the extrema, so that for a limited amount of truncation, no cycles are broken. We will quantify this "safe" amount of truncation in Section 4. So, while the smoothing parameter still gets rid of the center circle, the truncation only gets rid of expanding tails.

Algorithm. The τ-truncation of a Reeb graph (G, f) can be computed by first storing the length of the longest up-path and down-path of each vertex. This can be done in linear time using a topological sort of the graph based on directing all edges upward. We can for each local maximum store that it has a 0-length up-path, and for the remaining vertices, processes in the order given by the topological sort, storing the length of their up-path based on the stored length of all previously processed neighbors. We store the length of the longest down-path for each vertex symmetrically. Now, we can compute for each edge how much of it remains in the truncation, and subdivide the edges if necessary. Finally, remove all vertices and edges that do not have a sufficiently long up-path or down-path. This procedure takes $O(n + m)$ time on a graph with n vertices at m edges. The truncated smoothing can be computed by first computing the smoothing [23] in $O(m \log(m + n))$ time, giving a total running time of $O(m \log(m + n))$.

4 Properties of truncated smoothing

We can visualize the relationship between τ and ε as drawn in Figure 5. For this figure, we assume we start with a connected Reeb graph (G, f) and study properties of $S^\tau_\varepsilon(G, f)$ which is represented by the point (ε, τ) in the plane. In the remainder of this section, we state the properties of S^τ_ε in different regions of the ε-τ-plane, culminating in the parameter space labeling of Figure 7. We will focus in this section on the case where G is a connected graph, although some results can be modified to incorporate disconnected inputs. These results on disconnected graphs, as well as many of the more technical proofs, are presented in the full version of the paper.

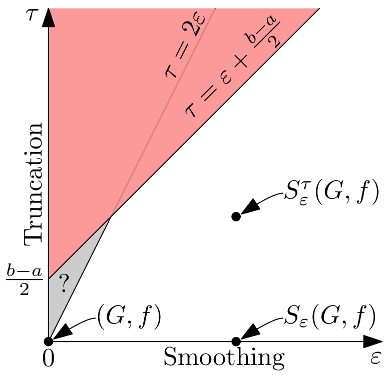


Figure 5 Visualization of Proposition 4.1. Given a connected G where $\mathrm{Im}(G,f) = [a,b] \subset \mathbb{R}$, $S_\varepsilon^\tau(G,f)$ is empty if it is in the red region and non-empty if it is in the white region. Parameters in the grey region can be either empty or not.

4.1 When is $S_\varepsilon^\tau(G,f)$ empty?

We first study the values of ε and τ for which the truncated smoothing is empty. For the purposes of notation, define $\mathrm{Im}(G,f) = f(G) \subset \mathbb{R}$. Consider the following simple example: Let $L_{[a,b]}$ be a Reeb graph consisting of a single edge with image $[a,b] \subseteq \mathbb{R}$, and for an interval $I \subseteq [a,b]$, let $L_I \subseteq L_{[a,b]}$ be the unique subgraph with image I. Then $T^\tau(L_{[a,b]}) = L_{[a+\tau,b-\tau]}$ if $2\tau \leq b-a$, and is the empty Reeb graph for $2\tau > b-a$. On the other hand, $S_\varepsilon(L_{[a,b]})$ is isomorphic to $L_{[a-\varepsilon,b+\varepsilon]}$.

In particular, T^τ and S_ε transform any monotone path with image $[a,b]$ into a monotone path with image $[a+\tau, b-\tau]$, $[a-\varepsilon, b+\varepsilon]$, respectively. In addition, smoothing or truncating the empty Reeb graph again yields the empty Reeb graph. We can build this intuition into the following proposition; details are in the full version of the paper. Note that in the case of a connected graph G, $\mathrm{Im}(G,f)$ is connected and thus is an interval.

▶ **Proposition 4.1.** *Let (G,f) be connected with $\mathrm{Im}(G,f) = [a,b]$.*
- *If $b-a < 2(\tau-\varepsilon)$, then $\mathrm{Im}(S_\varepsilon^\tau(G,f)) = \emptyset$.*
- *If $b-a \geq 2(\tau-\varepsilon)$ and $\tau \leq 2\varepsilon$, then $\mathrm{Im}(S_\varepsilon^\tau(G,f)) = [a-(\varepsilon-\tau), b+(\varepsilon-\tau)]$.*

Sketch proof. We first show that $b-a < 2(\tau-\varepsilon)$ implies the image is empty. We show in the full version of the paper that for a connected graph (H,h) with image $[a',b']$ and $b'-a' < 2\tau$, $T^\tau(H,h)$ is empty. By Proposition 2.4, $\mathrm{Im}(S_\varepsilon(G,f)) = [a-\varepsilon, b+\varepsilon]$. Then setting $S_\varepsilon(G,f) = (H,h)$, we have for $b-a < 2(\tau-\varepsilon)$, that $(b+\varepsilon)-(a-\varepsilon) \leq 2\tau$, so $\mathrm{Im}(S_\varepsilon^\tau(G,f)) = \mathrm{Im}(T^\tau(S_\varepsilon(G,f))) = \emptyset$.

Now, we can assume $b-a \geq 2(\tau-\varepsilon)$. One direction of containment is easy since by Proposition 2.4, $\mathrm{Im}(S_\varepsilon^\tau(G,f)) = \mathrm{Im}(T^\tau(S_\varepsilon(G,f))) \subseteq [a-(\varepsilon-\tau), b+(\varepsilon-\tau)]$. Thus, it remains to show that $[a-(\varepsilon-\tau), b+(\varepsilon-\tau)] \subseteq \mathrm{Im}(S_\varepsilon^\tau(G,f))$. The basic idea is to take two points $s,t \in S_\varepsilon(G,f)$ with $f(s) = a-\varepsilon$ and $f(t) = b+\varepsilon$, and show that they are connected by a path π in $S_\varepsilon(G,f)$ for which the only portions that get truncated are the endpoints. This is simple if π is itself a monotone path; otherwise we use the fact that G has already been smoothed and that we do not truncate too much ($\tau \leq 2\varepsilon$) to show that the parts of the path which are not monotone still have long enough up- and down-paths to not be removed. ◀

This proposition gives us that $S_\varepsilon^\tau(G,f)$ is an empty graph if (ε,τ) is interior to the red region of Figure 5, and is empty in the white region. We cannot expand this proposition to the grey region of Figure 5 as there are examples for which $S_\varepsilon^\tau(G,f)$ can be either empty or

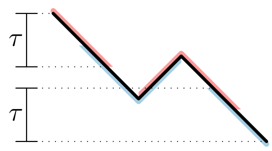


Figure 6 A Reeb graph (G, f) for which $T^\tau(G, f) = S_0^\tau(G, f)$ is empty. This choice of τ is such that $\text{Im}(G, f)$ has diameter greater than 2τ, thus $S_0^\tau(G, f)$ is in the grey region of Figure 5.

not. For instance, in the example of Figure 6, $|\text{Im}(G, f)| \geq 2(\tau - \varepsilon)$, but each position in the graph is either missing a long enough up- or down-path, and hence the truncated graph is empty. On the other hand, for the graph with a single edge $L_{[a,b]}$, any truncation $\tau < \frac{b-a}{2}$ is non empty.

4.2 When does $S_\varepsilon^\tau(G, f)$ maintain connectivity?

Our next goal is to understand when truncation preserves the connectivity of the input. As seen in Figure 4, clearly just truncating the graph can disconnect an originally connected graph. However, what is interesting is that smoothing first and not truncating too much relative to the smoothing will maintain the connectivity; this will be made precise in Proposition 4.6. For this, we introduce two properties, *t-tailed* and *s-safe*, and study how they are affected by smoothing and truncation.

▶ **Definition 4.2.** *A Reeb graph is* t-tailed *if it has a height t up-path at every down-fork and a length t down-path at every up-fork. A Reeb graph is* weakly s-safe *if each component has a point with both an up-path and a down-path of height at least s. A Reeb graph is* s-safe *if it is both s-tailed and weakly s-safe.*

Note that every non-empty Reeb graph is 0-safe. For example, the graph drawn in Figure 2 is not δ-tailed because the bottommost up-fork has no down-path of height δ; in addition, the topmost down-fork has no up-path of height δ.

We next have two results, proved in the full version of the paper, which show how the •-tailed and •-safe properties are maintained under smoothing and truncating, albeit with modified parameters.

▶ **Proposition 4.3.** *If (G, f) is t-tailed, then $S_\varepsilon(G, f)$ is $(t + 2\varepsilon)$-tailed. If (G, f) is s-safe, then $S_\varepsilon(G, f)$ is $(s + \varepsilon)$-safe. In particular, $S_\varepsilon(G, f)$ is always 2ε-tailed and ε-safe.*

▶ **Lemma 4.4.** *Fix $0 \leq \tau \leq \varepsilon$. If (G, f) is ε-tailed or safe, then $T^\tau(G, f)$ is $(\varepsilon - \tau)$-tailed or safe, respectively.*

Combining Proposition 4.3 and Lemma 4.4, we can see that outside the pink and grey regions of Figure 7, we know that $S_\varepsilon^\tau(G, f)$ is $(t + 2\varepsilon - \tau)$-tailed and $(s + \varepsilon - \tau)$-safe.

▶ **Proposition 4.5.** *Fix $0 \leq \varepsilon$ and $0 \leq \tau$, and assume (G, f) is t-tailed and s-safe. If $\tau \leq t + 2\varepsilon$ and $\tau \leq \varepsilon + \|\text{Im}(G, f)\|/2$, then $S_\varepsilon^\tau(G, f)$ is $(t + 2\varepsilon - \tau)$-tailed and $(s + \varepsilon - \tau)$-safe.*

Proof. Because (G, f) is t-tailed, $S_\varepsilon(G, f)$ is $(t + 2\varepsilon)$-tailed by the first statement of Proposition 4.3. Since $\tau \leq t + 2\varepsilon$, $S_\varepsilon^\tau(G, f) = T^\tau S_\varepsilon(G, f)$ is $(t + 2\varepsilon - \tau)$-tailed by Lemma 4.4. Similarly, since (G, f) is s-safe, $S_\varepsilon(G, f)$ is $(t + \varepsilon)$-safe by the second statement of Proposition 4.3. Then since $\tau \leq t + 2\varepsilon$, $S_\varepsilon^\tau(G, f) = T^\tau S_\varepsilon(G, f)$ is $(s + \varepsilon - \tau)$-safe by Lemma 4.4. ◀

This brings us to our conclusion of parameters for which the connectivity is maintained, with full details provided in the full version of the paper.

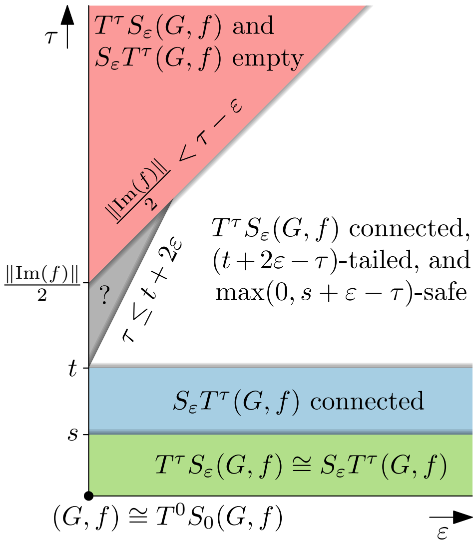


Figure 7 For connected, t-tailed, and s-safe (G, f), properties of $S_\varepsilon^\tau(G, f) = T^\tau S_\varepsilon(G, f)$ and $S_\varepsilon T^\tau(G, f)$, parameterized by τ and ε.

▶ **Proposition 4.6.** *If (G, f) is connected and $\tau \in [0, 2\varepsilon]$, then $S_\varepsilon^\tau(G, f)$ is also connected.*

Sketch proof. We show in the full version of the paper that for a connected, t-tailed graph, $T^t(G, f)$ is connected by ensuring disjointness of the portion of the graph G removed because it is lacking an up-path, and that which is removed because it is lacking a down-path. The result is then a corollary of Proposition 4.3. ◀

4.3 When do S^ε and T^τ commute?

We finally investigate the commutativity of smoothing and truncating. The example of Figure 4 shows why we must be careful with order of operations since $T^\tau S_\varepsilon(G, f)$ is not necessarily the same as $S_\varepsilon T^\tau(G, f)$. Specifically, $S_{2\varepsilon}^{2\varepsilon}(G, f) = T^{2\varepsilon} S_{2\varepsilon}(G, f)$ has one connected component, but any smoothing of $T^{2\varepsilon}(G, f)$ has two connected components. However, the next two results imply that this issue does not arise if we smooth sufficiently before truncating.

▶ **Proposition 4.7.** *If (G, f) is τ-safe, then $S_\varepsilon T^\tau(G, f) \cong T^\tau S_\varepsilon(G, f)$.*

The proof is provided in the full version of the paper. Combining the proposition with Lemma 4.4 and Proposition 4.3 gives the surprising result that the functors T and S do commute in the green region of Figure 7. We can next use this result to show that for certain choices of ε and τ, we can additively combine the parameters for truncated smoothing.

▶ **Theorem 4.8.** *If (1) (G, f) is empty or (2) $\tau_1 \leq 2\varepsilon_1$ and (G, f) is weakly $(\tau_1 - \varepsilon_1)$-safe, then $S_{\varepsilon_2}^{\tau_2} S_{\varepsilon_1}^{\tau_1}(G, f) \cong S_{\varepsilon_1+\varepsilon_2}^{\tau_1+\tau_2}(G, f)$.*

Proof. Both smoothing and truncating the empty Reeb graph yields the empty Reeb graph. So we are done if (G, f) is the empty Reeb graph, and we obtain not only an isomorphism but an equality. Now suppose that (G, f) is not empty. Then $S_{\varepsilon_1}(G, f)$ is $2\varepsilon_1$-tailed and weakly $(\tau_1 - \varepsilon_1 + \varepsilon_1)$-safe, and by definition $\min(2\varepsilon_1, \tau_1) \geq \tau_1$-safe. Therefore $S_{\varepsilon_2} T^{\tau_1} S_{\varepsilon_1}(G, f) \cong T^{\tau_1} S_{\varepsilon_2} S_{\varepsilon_1}(G, f)$, and hence using Proposition 4.7,

$$S_{\varepsilon_2}^{\tau_2} S_{\varepsilon_1}^{\tau_1}(G, f) = T^{\tau_2} S_{\varepsilon_2} T^{\tau_1} S_{\varepsilon_1}(G, f) \cong T^{\tau_2} T^{\tau_1} S_{\varepsilon_2} S_{\varepsilon_1}(G, f) \cong S_{\varepsilon_1+\varepsilon_2}^{\tau_1+\tau_2}(G, f).$$ ◀

In particular, the assumptions of the theorem are satisfied if $\tau_1 \leq \varepsilon_1$ since every non-empty graph is 0-safe.

5 Truncated interleaving distance

In this section, we survey the results related to defining the family of truncated interleaving distances, proving that certain linear subspaces of our two parameter functor space (shown in Figure 7) form a categorical flow. Since any category with a flow gives an interleaving distance, we then use truncated smoothing to build a new family of metrics for Reeb graphs.

The whole idea behind building a category with a flow is that the flow itself must be functorial, which means we must have knowledge of how it acts both on objects and morphisms. So far, the results discussed in Section 4 only correspond to the object information. In the full version of the paper, we will describe how to explicitly build the morphisms $S^\tau_\varepsilon(G, f) \to S^{\tau'}_{\varepsilon'}(G, f)$ (i.e., function preserving maps). However, these morphisms are only available for certain choices of parameters. Restricting our view only to (ε, τ) pairs for which these morphisms exist gives us that for any choice of $m \in [0, 1]$ we can set $\tau = m\varepsilon$ to get a flow.

▶ **Theorem 5.1.** *For any $m \in [0, 1]$, the map $S^m : ([0, \infty), \leq) \to \mathbf{End}(\mathbf{Reeb}); \varepsilon \mapsto S^{m\varepsilon}_\varepsilon$ is a functor and defines a categorical flow on* **Reeb**.

Essentially, this m can be thought of as defining the slope of a line based at the origin in the parameter space of Figure 7, and thus using Theorem 2.2, we have an interleaving distance for any line with slope less than 1.

▶ **Corollary 5.2.** *For any $m \in [0, 1]$, S^m gives rise to an interleaving-type distance*

$$d^m_I((G, f), (H, h)) := \inf\{\varepsilon \geq 0 \mid \textit{there exists a } \varepsilon\textit{-interleaving with respect to } S^m\}.$$

Specifically, d^m_I is an extended pseudo-metric.

In the next theorem, we show that with the exception of $m = 1$, all the metrics created are closely related in the following sense. Two metrics d_A and d_B are said to be *strongly equivalent* if there are positive constants α_1 and α_2 such that $\alpha_1 d_A \leq d_B \leq \alpha_2 d_A$. In the following theorem, we show that d^m_I and $d^{m'}_I$ are strongly equivalent if (m, m') is contained in the white region of Figure 8.

▶ **Theorem 5.3.** *For any pair $m, m' \in [0, 1)$ with $0 \leq m' - m < 1 - m'$ the metrics d^m_I and $d^{m'}_I$ are strongly equivalent. Specifically, given Reeb graphs (G, f) and (H, h),*

$$d^m_I((G, f), (H, h)) \leq d^{m'}_I((G, f), (H, h)) \leq \frac{1 - m}{1 - m'} d^m_I((G, f), (H, h))$$

The proof of this theorem is contained in the full version of the paper. Of course, as long as we are willing to loosen the bounds, this result extends to any pair of $m, m' \in [0, 1)$.

▶ **Corollary 5.4.** *For all pairs $0 \leq M \leq M' < 1$, there exist positive constants C_1 and C_2 dependent on M and M' such that*

$$C_1 d^M_I((G, f), (H, h)) \leq d^{M'}_I((G, f), (H, h)) \leq C_2 d^M_I((G, f), (H, h)),$$

and thus d^M_I and $d^{M'}_I$ are strongly equivalent metrics.

Proof. Consider M, M' given with $M \leq M'$. If $M' \leq \frac{1+M}{2}$, then Theorem 5.3 applies directly. Otherwise, we assume that $M' \geq \frac{1+M}{2}$. Then d^M_I is equivalent to d^α_I for any α in the interval $(M, \frac{1+M}{2})$ and $d^{M'}_I$ is equivalent to d^β_I for any β in the interval $(2M' - 1, M')$. Then there is a zigzag like the example in Figure 8 between α and β which remains in the white region and for which each adjacent pair are strongly equivalent metrics. Equivalence of metrics is transitive, so this implies d^M_I and $d^{M'}_I$ are equivalent. ◀

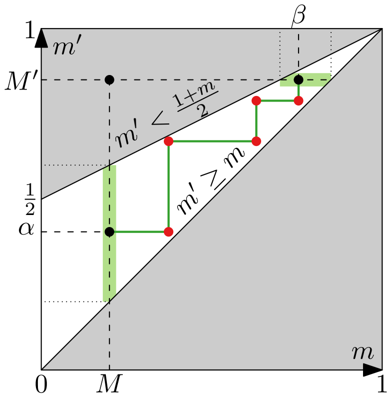


Figure 8 Parameter space for comparing metrics d_I^m and $d_I^{m'}$. The white region is allowable pairs for Theorem 5.3. The vertices of the zigzag (shown as red points) give pairs of strongly equivalent metrics which, when combined, show that M and M' are strongly equivalent in Corollary 5.4.

In particular, this corollary gives that the original Reeb graph interleaving distance (where $m = 0$) is strongly equivalent to d_I^m for all $m \in [0, 1)$. We note that there are many possible zigzag paths which can be used to obtain this bound, but further exploration is needed to determine which, if any, provide optimal constants.

6 Properties of the metrics

As noted, d_I^m is an extended pseudometric, which means it is possible for $d_I^m((G, f), (H, h))$ to be infinite. However, it turns out this is not the case for broad classes of graphs. In fact, in order to take infinite value, there must be no ε-interleaving with respect to S^m between the two Reeb graphs. That being said, there are very specific instances where this metric takes on infinite value.

The easiest case to handle is when $m \in [0, 1)$, since we can use the characterization given in [23] in conjunction with the equivalence of metrics Corollary 5.4.

▶ **Proposition 6.1.** *Let $m \in [0, 1)$. Then $d_I^m((G, f), (H, h)) < \infty$ iff G and H have the same number of path-connected components.*

Proof. Note that $d_I^0 = d_I$. By [23, Prop. 4.5], $d_I((G, f), (H, h))$ is finite if and only if G and H have the same number of path connected components. This combined with Corollary 5.4 gives the proposition. ◀

The characterization of when d_I^m is infinite for $m = 1$ is more complicated. Consider a connected graph (G, f) with $\mathrm{Im}(G, f) = [a, b]$. When $m = 1$, we are interested in understanding the behavior of $S_\varepsilon^\varepsilon(G, f)$. By Proposition 4.1, we see that $b - a \geq 2(\tau - \varepsilon) = 0$, so $S_\varepsilon^\varepsilon(G, f) = [a, b]$. That is to say that the image of (G, f) is unchanged by $S_\varepsilon^\varepsilon$. Now, if we wanted to determine the interleaving distance d_I^1 for a given (G, f) and (H, h), one requirement is always that we must smooth the given graphs enough for there to be a morphism $(G, f) \to S_\varepsilon^\varepsilon(H, h)$. However, because $S_\varepsilon^\varepsilon$ does not change the image, the function preserving requirement of morphisms mean that if the graphs did not start with the same image no choice of ε will make this possible. With this example in mind, we can characterize when d_I^m takes on infinite values for $m = 1$.

▶ **Proposition 6.2.** *Let $m = 1$ and assume G and H are connected. Then*

$$d_I^m((G,f),(H,h)) < \infty \text{ if and only if } \mathrm{Im}(G,f) = \mathrm{Im}(H,h).$$

Further, if $\mathrm{Im}(G,f) = \mathrm{Im}(H,h)$*, then* $d_I^m((G,f),(H,h)) \leq |\mathrm{Im}(G,f)|$.

Proof. Note that by Proposition 4.1, for any connected G' with $\mathrm{Im}(G',f') = [a,b]$, we have $\mathrm{Im}S_\varepsilon^\varepsilon(G',f') = [a,b]$. So the truncated smoothing maintains the image for every connected component, and thus for the union of the connected components. Thus, we have $\mathrm{Im}(G,f) = \mathrm{Im}(S_\varepsilon^\varepsilon(G,f))$ and $\mathrm{Im}(H,h) = \mathrm{Im}(S_\varepsilon^\varepsilon(H,h))$ for any choice of ε.

Assume we have an $S_\varepsilon^\varepsilon$ interleaving $\varphi\colon (G,f) \to S_\varepsilon^\varepsilon(H,h)$ and $\psi\colon (H,h) \to S_\varepsilon^\varepsilon(G,f)$. Because φ and ψ are function preserving, $\varphi(G) = \mathrm{Im}(G,f) \subseteq \mathrm{Im}(S_\varepsilon^\varepsilon(H,h))$ and $\psi(H) = \mathrm{Im}(H,h) \subseteq \mathrm{Im}(S_\varepsilon^\varepsilon(G,f))$. But since $S_\varepsilon^\varepsilon$ leaves the images unchanged, this implies that $\mathrm{Im}(G,f) = \mathrm{Im}(H,h)$.

Now assume $\mathrm{Im}(G,f) = \mathrm{Im}(H,h)$. Let $\varepsilon = |\mathrm{Im}(G,f)|$ and consider the thickening $G \times [-\varepsilon,\varepsilon]$ and a value $a \in \mathrm{Im}(G,f)$. We claim that $(f+\mathrm{Id})^{-1}(a) \subseteq G \times [-\varepsilon,\varepsilon]$ is exactly $A = \{(x, a - f(x)) \mid x \in G\}$ and in particular, that it is homeomorphic to G. Indeed, for any $x \in G$, $a - f(x) \in [-\varepsilon,\varepsilon]$ and the point $y = (x, a-f(x))$ has image $(f+\mathrm{Id})(y) = a$ so $A \subseteq (f+\mathrm{Id})^{-1}(a)$. Moreover, for any $(x,t) \in (f+\mathrm{Id})^{-1}(a)$, $f(x)+t = a$ so $t = a - f(x)$, thus $(f+\mathrm{Id})^{-1}(a) \subseteq A$.

So, since G is connected and f is continuous, $(f+\mathrm{Id})^{-1}(a) \cong G$ is a single connected component for any $a \in \mathrm{Im}(G,f)$, and the same is true for (H,h). Because the $S_\varepsilon^\varepsilon$ smoothing maintains the image, this implies $S_\varepsilon^\varepsilon(G,f) = S_\varepsilon^\varepsilon(H,f)$ is a single line segment with the same image. We obtain an interleaving by simply sending every point in (G,f) to the unique point at the same height in $S_\varepsilon(H,h)$ and vice versa, so the d_I^m distance is finite. ◀

We next investigate stability, for this collection of metrics.

▶ **Definition 6.3.** *Let $(\mathbb{X}, f)$ and $(\mathbb{X}, g)$ be $\mathbb{R}$-spaces with the same total space $\mathbb{X}$, and let $R(\mathbb{X}, f)$ and $R(\mathbb{Y}, g)$ be the respective Reeb graphs. A metric d is said to be stable if*

$$d(R(\mathbb{X},f), R(\mathbb{X},g)) \leq \|f-g\|_\infty.$$

The original Reeb interleaving distance, $m = 0$, is stable [23, Thm 4.4]. Unfortunately, d_I^m is not stable in the strictest sense; to see why, consider the following simple example. Consider two simple line segments for graphs, for example, (L, f_1) and (L, f_2) where $\mathrm{Im}(L,f_1) = [-a,a]$ and $\mathrm{Im}(L,f_2) = [-b,b]$ for $a < b$. Then $\|f_1 - f_2\|_\infty = b - a$. However, the interleaving distance requires that we smooth at least until $[-b,b] = \mathrm{Im}(L,f_2) \subseteq \mathrm{Im}(S_\varepsilon(L,f_1))$. But by Proposition 4.1, $\mathrm{Im}(S_\varepsilon(L,f_1)) = [a-(\varepsilon - m\varepsilon), a+(\varepsilon-m\varepsilon)]$. Thus $d_I^m(f_1,f_2) \geq \frac{b-a}{1-m} \geq b-a$, and is strictly greater if $m \neq 0$. This means that $b - a = \|f_1 - f_2\|_\infty < d_I^m((L,f_1),(L,f_2))$, and thus d_I^m is not stable.

We can regain at least partial control of the distance, however, as d_I^m is still Lipschitz when given a fixed choice of m.

▶ **Proposition 6.4.** *Let $m \in [0,1)$. Assume $(\mathbb{X}, g_1)$ and $(\mathbb{X}, g_2)$ are given for a connected space $\mathbb{X}$ and denote the associated Reeb graphs by (G,f) and (H,h) respectively. Then there is a positive constant C dependent on m for which*

$$d_I^m((G,f),(H,h)) \leq C\|g_1 - g_2\|_\infty.$$

Proof. By Corollary 5.4, d_I^0 and d_I^m are strongly equivalent metrics, so there is a positive constant C for which $d_I^m \leq C d_I^0$. Then because the Reeb graph interleaving distance d_I^0 is stable, we have

$$d_I^m((G,f),(H,h)) \leq C d_I^0((G,f),(H,h)) \leq C\|g_1 - g_2\|.$$ ◀

Because of the dependence on Corollary 5.4 where the optimal choice of zigzag to find the constant C is unclear, we do not give an explicit formulation here. We conclude by connecting our extended pseudometric to two other metrics for Reeb graphs, the functional distortion distance [2] and the bottleneck distance [43]. The proof is a straightforward implication of inequalities, so due to space constraints we simply state these results without formally defining either. The interested reader can find further details on the metrics in [2] and [43].

▶ **Proposition 6.5.** *The truncated interleaving distance is strongly equivalent to the functional distortion distance. Further, defining d_B as the bottleneck distance of the level set persistent homology, we have the inequality $d_B \leq 5d_I^m$.*

Proof. The interleaving distance, d_I^0, is strongly equivalent to the functional distortion distance by [4, Thm 16]. So by Corollary 5.4 and transitivity of strong equivalence, they are each strongly equivalent to d_I^m for any $m \in [0, 1)$. To obtain the inequality, we use the bound on the bottleneck distance of level set persistent homology by the Reeb graph interleaving distance in [9, Thm. 4.13]. ◀

7 Conclusion and discussion

Our primary aim has been to introduce the concept of truncated smoothing and establish properties and connections of this operation. We have several reasons for considering this as a similarity measure on Reeb graphs. First, it has potential for providing bounds for the stable interleaving distance via the equivalence of metrics. Second, we came to this definition while investigating drawings of Reeb graphs and when planarity is achievable (that, is whether a Reeb graph has a planar drawing which respects the function in the y-coordinate). In a subsequent paper, we will show that while traditional smoothing does not maintain planarity, the truncated smoothing does for $\varepsilon \leq \tau \leq 2\varepsilon$.

We suspect additional potential applications of truncated smoothing in comparing geometric or planar graphs, since it simplifies the graph's topology (via smoothing) without suffering from extensive expansion of the co-domain or destruction of desirable combinatorial properties like level planarity. Truncated smoothing also allows for interesting manipulation of the extended persistence diagram of the Reeb graph and computation of morphs between Reeb graphs; again, we defer details to future work, as a full classification of that manipulation is necessary.

We suspect that the loss of stability discussed in Section 6 is not as dire as it seems. If nothing else, Proposition 6.4 gives a Lipschitz constant in advance dependent only on m, so it is possible to upper bound the difference using these new interleaving distances.

While we are able to connect our collection of metrics to several Reeb metrics (Proposition 6.5), we have not investigated further connections to other metrics as of yet. One particularly interesting future direction is to determine whether this collection of metrics provides results related to strong equivalence between the interleaving distance and the universal distance of [3]. Perhaps this broader collection of metrics will help to provide stronger bounds between the various metrics on Reeb graphs, since strong equivalence with one is strong equivalence with all.

References

1 Ulrich Bauer, Barbara Di Fabio, and Claudia Landi. An edit distance for Reeb graphs. In A. Ferreira, A. Giachetti, and D. Giorgi, editors, *Eurographics Workshop on 3D Object Retrieval*. The Eurographics Association, 2016. doi:10.2312/3dor.20161084.

2 Ulrich Bauer, Xiaoyin Ge, and Yusu Wang. Measuring distance between Reeb graphs. In *Proceedings of the Thirtieth Annual Symposium on Computational Geometry - SoCG '14, Kyoto, Japan*, 2014.

3 Ulrich Bauer, Claudia Landi, and Facundo Mémoli. The Reeb Graph Edit Distance Is Universal. In Sergio Cabello and Danny Z. Chen, editors, *36th International Symposium on Computational Geometry (SoCG 2020)*, volume 164 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 15:1–15:16, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.SoCG.2020.15.

4 Ulrich Bauer, Elizabeth Munch, and Yusu Wang. Strong equivalence of the interleaving and functional distortion metrics for Reeb graphs. In Lars Arge and János Pach, editors, *31st International Symposium on Computational Geometry (SoCG 2015)*, volume 34 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 461–475, Dagstuhl, Germany, 2015. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.SOCG.2015.461.

5 S. Biasotti, D. Giorgi, M. Spagnuolo, and B. Falcidieno. Reeb graphs for shape analysis and applications. *Theoretical Computer Science: Computational Algebraic Geometry and Applications*, 392(13):5–22, 2008. doi:10.1016/j.tcs.2007.10.018.

6 Håvard Bakke Bjerkevik and Magnus Bakke Botnan. Computational Complexity of the Interleaving Distance. In Bettina Speckmann and Csaba D. Tóth, editors, *34th International Symposium on Computational Geometry (SoCG 2018)*, volume 99 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 13:1–13:15, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.SoCG.2018.13.

7 Håvard Bakke Bjerkevik, Magnus Bakke Botnan, and Michael Kerber. Computing the interleaving distance is NP-hard. *Foundations of Computational Mathematics*, November 2019. doi:10.1007/s10208-019-09442-y.

8 Andrew J. Blumberg and Michael Lesnick. Universality of the homotopy interleaving distance. *CoRR*, 2017. arXiv:1705.01690v1.

9 Magnus Botnan and Michael Lesnick. Algebraic stability of zigzag persistence modules. *Algebraic & Geometric Topology*, 18(6):3133–3204, October 2018. doi:10.2140/agt.2018.18.3133.

10 Magnus Bakke Botnan, Justin Curry, and Elizabeth Munch. A relative theory of interleavings. *CoRR*, 2020. arXiv:2004.14286v1.

11 Adam Brown, Omer Bobrowski, Elizabeth Munch, and Bei Wang. Probabilistic convergence and stability of random mapper graphs. *Journal of Applied and Computational Topology*, December 2020. doi:10.1007/s41468-020-00063-x.

12 Peter Bubenik, Vin de Silva, and Jonathan Scott. Metrics for generalized persistence modules. *Foundations of Computational Mathematics*, 2014.

13 Peter Bubenik, Vin de Silva, and Jonathan Scott. Interleaving and Gromov-Hausdorff distance. *CoRR*, 2017. arXiv:1707.06288v3.

14 Peter Bubenik and Jonathan A. Scott. Categorification of persistent homology. *Discrete & Computational Geometry*, 51(3):600–627, 2014. doi:10.1007/s00454-014-9573-x.

15 Kevin Buchin, Maike Buchin, Marc van Kreveld, Bettina Speckmann, and Frank Staals. Trajectory grouping structure. In Frank Dehne, Roberto Solis-Oba, and Jörg-Rüdiger Sack, editors, *Algorithms and Data Structures*, volume 8037 of *Lecture Notes in Computer Science*, pages 219–230. Springer Berlin Heidelberg, 2013. doi:10.1007/978-3-642-40104-6_19.

16 Mathieu Carrière and Steve Oudot. Local equivalence and intrinsic metrics between Reeb graphs. In Boris Aronov and Matthew J. Katz, editors, *33rd International Symposium on Computational Geometry (SoCG 2017)*, volume 77 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 25:1–25:15, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.SoCG.2017.25.

17 Erin Wolf Chambers, Elizabeth Munch, and Tim Ophelders. A family of metrics from the truncated smoothing of Reeb graphs. *CoRR*, 2020. arXiv:2007.07795v3.

18 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, pages 237–246, New York, NY, USA, 2009. ACM. doi:10.1145/1542362.1542407.

19 Frédéric Chazal and Jian Sun. Gromov-Hausdorff approximation of filament structure using Reeb-type graph. In *Proceedings of the Thirtieth Annual Symposium on Computational*

Geometry, SOCG'14, pages 491:491–491:500, New York, NY, USA, 2014. ACM. doi:10.1145/2582112.2582129.

20 Joshua Cruz. Metric limits in categories with a flow. *CoRR*, 2019. arXiv:1901.04828v1.

21 Justin Curry. *Sheaves, Cosheaves and Applications*. PhD thesis, University of Pennsylvania, 2014. arXiv:1303.3255.

22 Justin Curry and Amit Patel. Classification of constructible cosheaves. *CoRR*, 2016. arXiv:1603.01587v5.

23 Vin de Silva, Elizabeth Munch, and Amit Patel. Categorified Reeb graphs. *Discrete & Computational Geometry*, pages 1–53, 2016. doi:10.1007/s00454-016-9763-9.

24 Vin de Silva, Elizabeth Munch, and Anastasios Stefanou. Theory of interleavings on categories with a flow. *Theory and Applications of Categories*, 33(21):583–607, 2018. URL: http://www.tac.mta.ca/tac/volumes/33/21/33-21.pdf.

25 Tamal K. Dey, Fengtao Fan, and Yusu Wang. An efficient computation of handle and tunnel loops via Reeb graphs. *ACM Trans. Graph.*, 32(4):32:1–32:10, 2013. doi:10.1145/2461912.2462017.

26 Tamal K. Dey and Yusu Wang. Reeb graphs: Approximation and persistence. *Discrete & Computational Geometry*, 49(1):46–73, 2013. doi:10.1007/s00454-012-9463-z.

27 B. Di Fabio and C. Landi. Reeb graphs of curves are stable under function perturbations. *Mathematical Methods in the Applied Sciences*, 35(12):1456–1471, 2012. doi:10.1002/mma.2533.

28 Barbara Di Fabio and Claudia Landi. The edit distance for Reeb graphs of surfaces. *Discrete & Computational Geometry*, 55(2):423–461, January 2016. doi:10.1007/s00454-016-9758-6.

29 Francisco Escolano, Edwin R. Hancock, and Silvia Biasotti. Complexity fusion for indexing Reeb digraphs. In Richard Wilson, Edwin Hancock, Adrian Bors, and William Smith, editors, *Computer Analysis of Images and Patterns*, volume 8047 of *Lecture Notes in Computer Science*, pages 120–127. Springer Berlin Heidelberg, 2013. doi:10.1007/978-3-642-40261-6_14.

30 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, September 9-11, 2019, Munich/Garching, Germany*, volume 144 of *LIPIcs*, pages 83:1–83:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPICS.ESA.2019.83.

31 Ellen Gasparovic, Elizabeth Munch, Steve Oudot, Katharine Turner, Bei Wang, and Yusu Wang. Intrinsic interleaving distance for merge trees. *CoRR*, 2019. arXiv:1908.00063.

32 Xiaoyin Ge, Issam I. Safa, Mikhail Belkin, and Yusu Wang. Data skeletonization via Reeb graphs. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 837–845, 2011.

33 William Harvey and Yusu Wang. Topological landscape ensembles for visualization of scalar-valued functions. In *Proceedings of the 12th Eurographics / IEEE - VGTC Conference on Visualization*, EuroVis'10, pages 993–1002, Aire-la-Ville, Switzerland, Switzerland, 2010. Eurographics Association. doi:10.1111/j.1467-8659.2009.01706.x.

34 Franck Hétroy and Dominique Attali. Topological quadrangulations of closed triangulated surfaces using the Reeb graph. *Graphical Models*, 65(1-3):131–148, 2003. doi:10.1016/s1524-0703(03)00005-5.

35 Masaki Hilaga, Yoshihisa Shinagawa, Taku Kohmura, and Tosiyasu L. Kunii. Topology matching for fully automatic similarity estimation of 3D shapes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '01, pages 203–212, New York, NY, USA, 2001. ACM. doi:10.1145/383259.383282.

36 Woojin Kim and Facundo Memoli. Stable signatures for dynamic metric spaces via zigzag persistent homology. *CoRR*, 2017. arXiv:1712.04064v1.

37 Woojin Kim, Facundo Mémoli, and Anastasios Stefanou. The metric structure of the formigram interleaving distance. *CoRR*, 2019. arXiv:1912.04366v1.

38 Michael Lesnick. The theory of the interleaving distance on multidimensional persistence modules. *Foundations of Computational Mathematics*, 15(3):613–650, 2015. `doi:10.1007/s10208-015-9255-y`.

39 Dmitriy Morozov, Kenes Beketayev, and Gunther Weber. Interleaving distance between merge trees. In *Proceedings of TopoInVis*, 2013.

40 Elizabeth Munch and Anastasios Stefanou. The $\ell\infty$-cophenetic metric for phylogenetic trees as an interleaving distance. In *Association for Women in Mathematics Series*, pages 109–127. Springer International Publishing, 2019. `doi:10.1007/978-3-030-11566-1_5`.

41 Elizabeth Munch and Bei Wang. Convergence between categorical representations of Reeb space and Mapper. In Sándor Fekete and Anna Lubiw, editors, *32nd International Symposium on Computational Geometry (SoCG 2016)*, volume 51 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 53:1–53:16, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. `doi:10.4230/LIPIcs.SoCG.2016.53`.

42 Mattia Natali, Silvia Biasotti, Giuseppe Patanè, and Bianca Falcidieno. Graph-based representations of point clouds. *Graphical Models*, 73(5):151–164, 2011. `doi:10.1016/j.gmod.2011.03.002`.

43 Steve Y. Oudot. *Persistence Theory: From Quiver Representations to Data Analysis (Mathematical Surveys and Monographs)*. American Mathematical Society, 2017.

44 Georges Reeb. Sur les points singuliers d'une forme de pfaff complèment intégrable ou d'une fonction numérique. *Comptes Rendus de L'Académie ses Séances*, 222:847–849, 1946.

45 Luis Scoccola. *Locally persistent categories andmetric properties of interleaving distances*. PhD thesis, Western University, 2020.

46 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.

47 Raghavendra Sridharamurthy, Talha Bin Masood, Adhitya Kamakshidasan, and Vijay Natarajan. Edit distance between merge trees. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1, 2018. `doi:10.1109/tvcg.2018.2873612`.

48 Anastasios Stefanou. *Dynamics on Categories and Applications*. PhD thesis, University at Albany, State University of New York, 2018.

49 Anastasios Stefanou. Tree decomposition of reeb graphs, parametrized complexity, and applications to phylogenetics. *Journal of Applied and Computational Topology*, February 2020. `doi:10.1007/s41468-020-00051-1`.

50 Andrzej Szymczak. A categorical approach to contour, split and join trees with application to airway segmentation. In V. Pascucci, H. Tricoche, H. Hagen, and J. Tierny, editors, *Topological Methods in Data Analysis and Visualization*, pages 205–216. Springer, 2011.

51 G.H. Weber, P.-T. Bremer, and V. Pascucci. Topological landscapes: A terrain metaphor for scientific data. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1416–1423, November 2007. `doi:10.1109/TVCG.2007.70601`.

52 Zoë Wood, Hugues Hoppe, Mathieu Desbrun, and Peter Schröder. Removing excess topology from isosurfaces. *ACM Trans. Graph.*, 23(2):190–208, 2004. `doi:10.1145/990002.990007`.

53 Lin Yan, Yusu Wang, Elizabeth Munch, Ellen Gasparovic, and Bei Wang. A structural average of labeled merge trees for uncertainty visualization. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1, 2019. `doi:10.1109/tvcg.2019.2934242`.

54 Hiroki Yuda. Topological smoothing of Reeb graphs. Master's thesis, St. Louis University, 2019.