


Question: Given 2 Reeb graphs, how to compare them?

Goal: At least an extended pseudo metric:

$$1) d(X, X) = \infty$$

$$2) d(X, Y) = d(Y, X)$$

$$3) d(X, Z) \leq d(X, Y) + d(Y, Z)$$

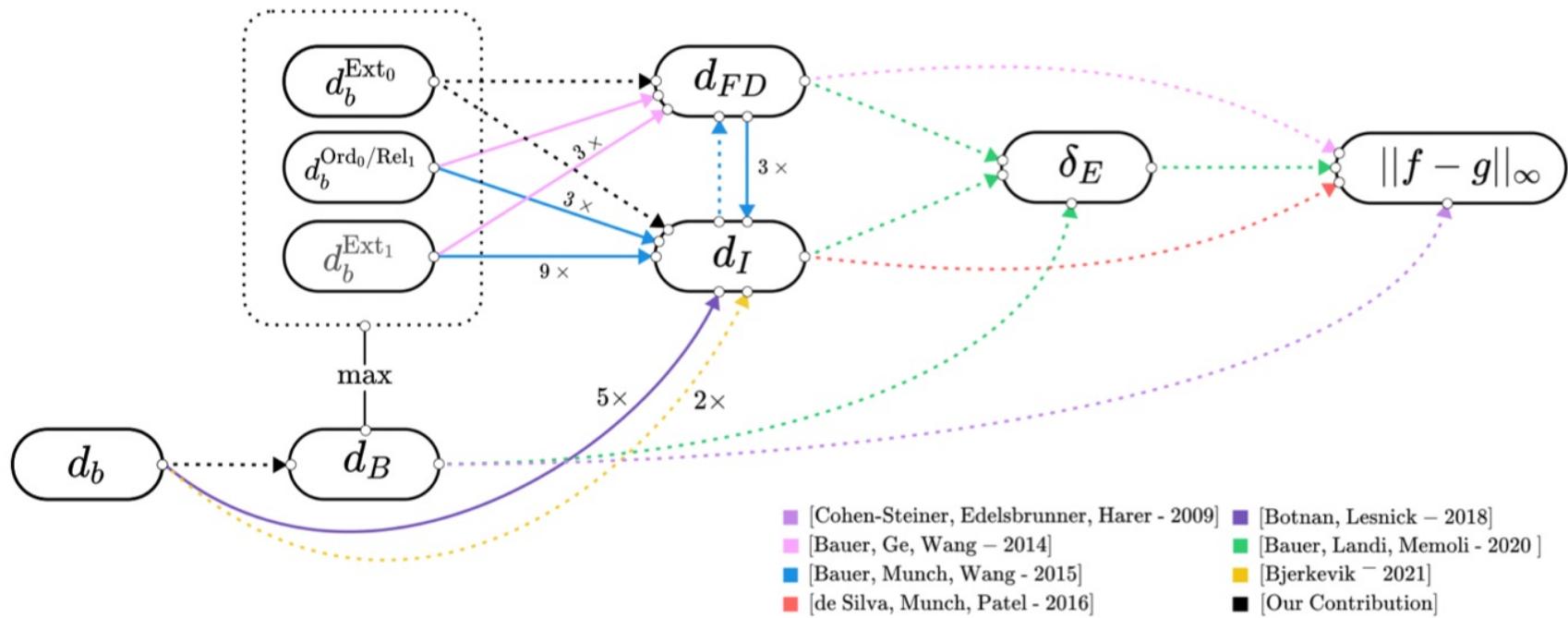
Weaker than metric:

-extended: could be ∞

-pseudo: $d(x, y) = 0 \not\Rightarrow x = y$

Lots of well-studied metrics!

Bollen-G-Leinweber-Munch 2023



d_B : bottleneck (cost tree)

d_I : interleaving (cont.)

d_{FD} : functional distortion

d_E : Edit distance

Interleaving instance: Some Category theory

A category C consists of objects (X, Y, Z , etc) with morphisms (\rightarrow) between them, satisfying some "niceness" properties (associativity, identity).

Examples	Objects	Morphisms
"Set"	Sets	functions
"Vect"	Vector Spaces	Linear Transformations
"Top"	Topological Spaces	Continuous Maps
"Int"	Intervals $(a, b) \subset \mathbb{R}$	\subseteq

Interleaving distance

Takes a categorical point of view.
Reeb graph is a set valued cosheaf,

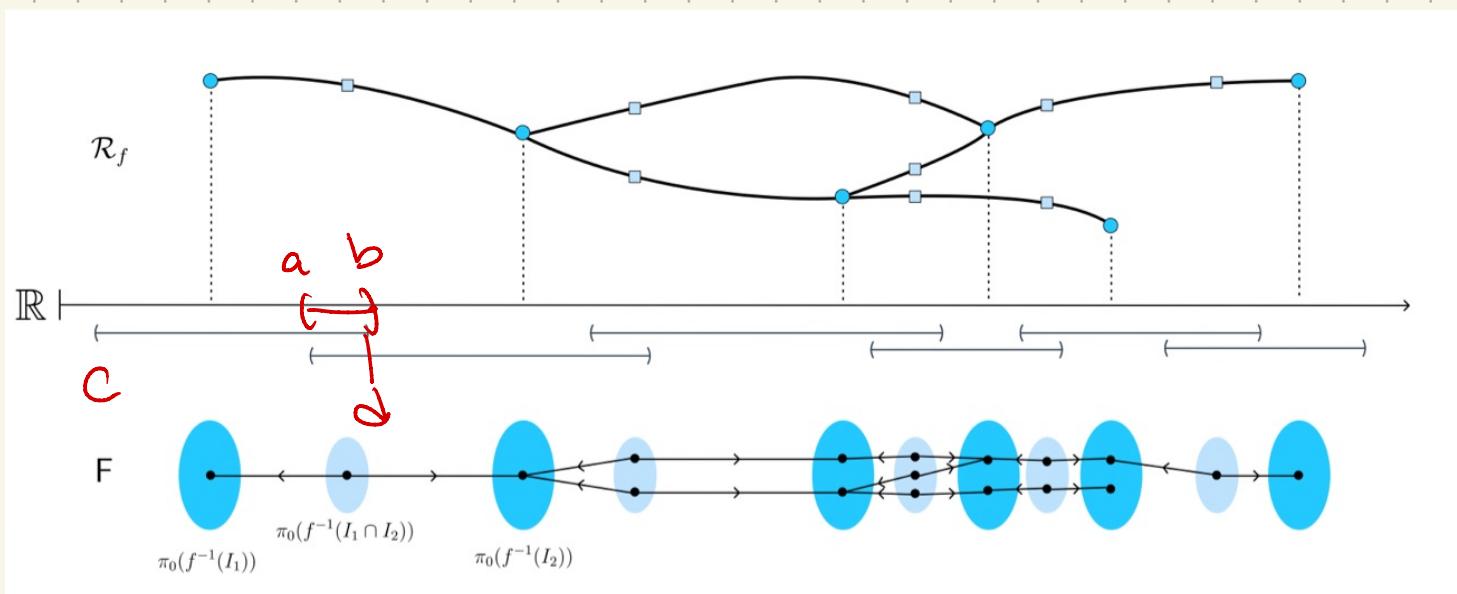
$$F: \text{Int} \rightarrow \text{Set}$$

$$I = (a, b) \longrightarrow \pi_0(f^{-1}((a, b)))$$



$$\downarrow F[I \subseteq J]$$

$$J = (c, d) \longrightarrow \pi_0(f^{-1}((c, d)))$$



Definition: "Reeb" is the category with Reeb graphs as objects, & morphisms as function preserving maps.

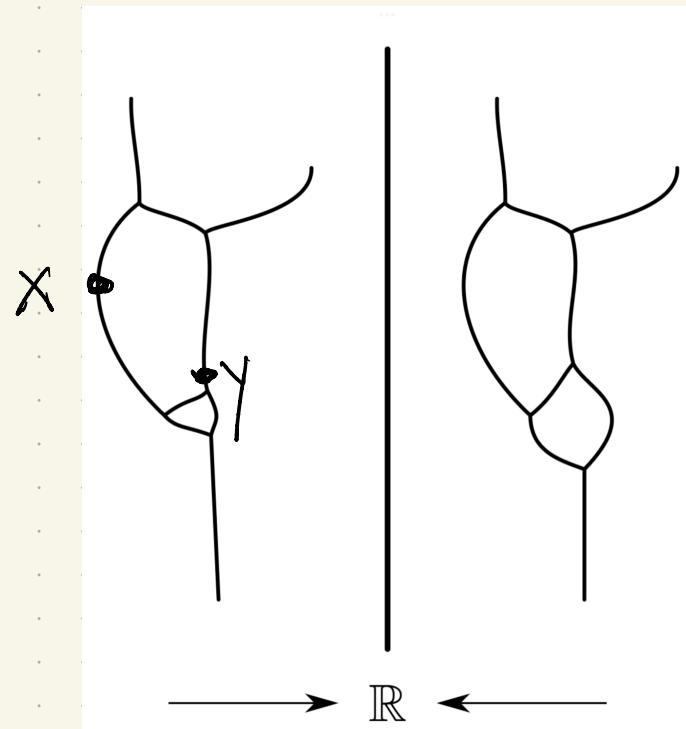
function preserving map:

$$\ell: (X_f) \rightarrow (Y_g)$$

is a continuous map s.t.

$$\begin{array}{ccc} X & \xrightarrow{\ell} & Y \\ f \searrow & & \swarrow g \\ & R & \end{array}$$

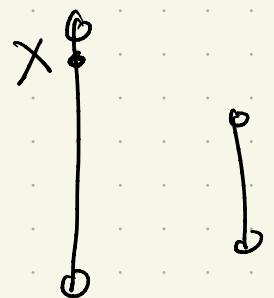
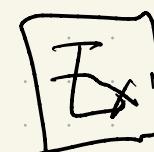
commutes.



$\phi(x)?$

Note:

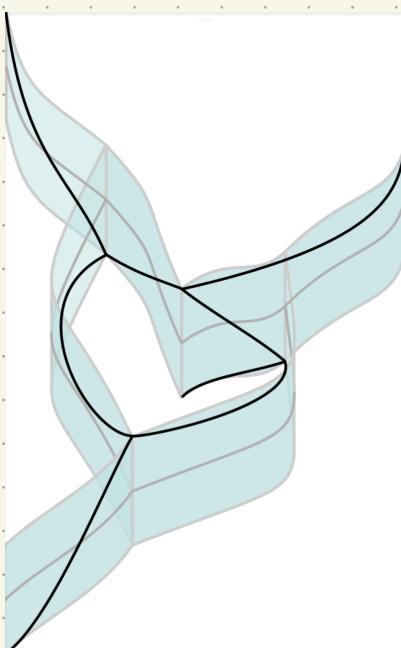
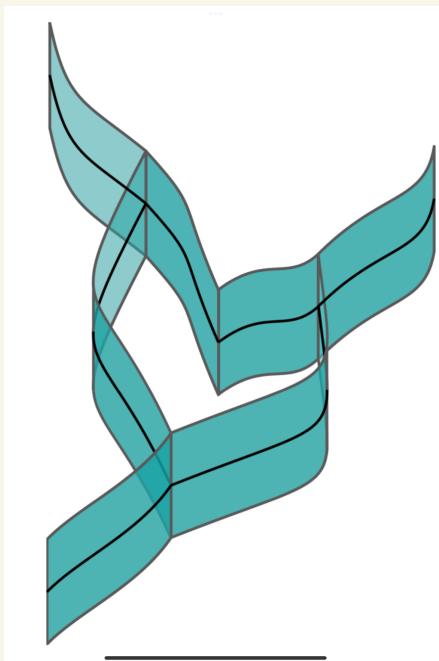
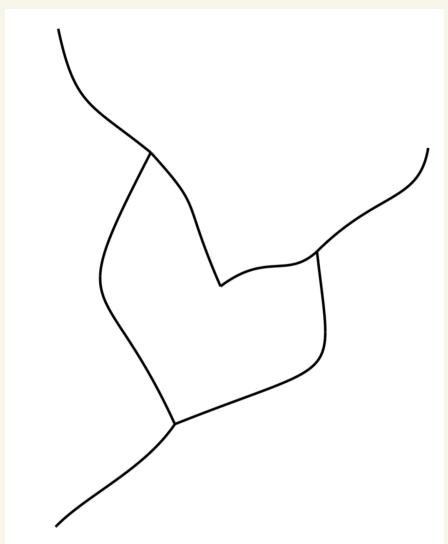
Don't always exist!



Smoothing a Reeb co-sheet: (Definition by example)

$$G \times [-\varepsilon, \varepsilon]$$

G



- Given (G, f)

- Thicken:

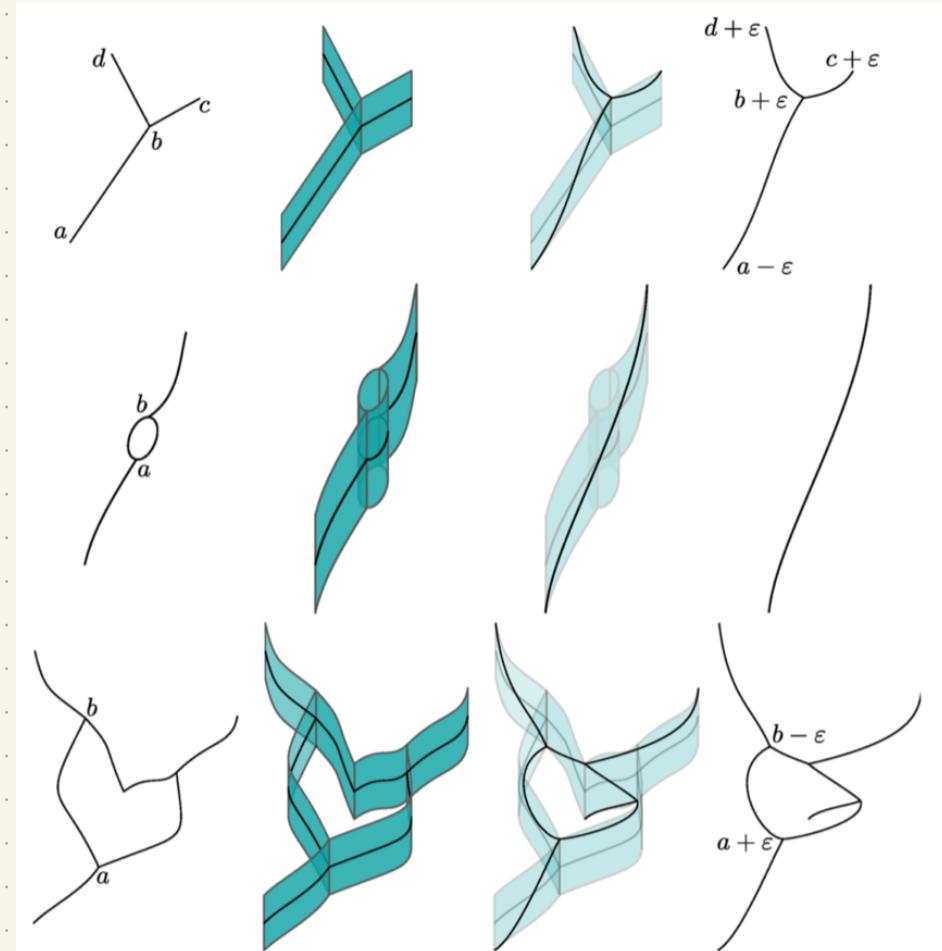
$$\begin{aligned}f_\varepsilon: G \times [-\varepsilon, \varepsilon] &\rightarrow \mathbb{R} \\(x, t) &\mapsto f(x) + t\end{aligned}$$

- Take Reeb graph of $(G \times [-\varepsilon, \varepsilon], f_\varepsilon)$

Result of smoothing: $S_\varepsilon(G, f)$:

- loops get 2ε smaller
- Max + mins "stretch"

Munch, de Silva
& Patel 2016



Interleaving distance: Why do we care??

Given 2 Reeb cosheaves

$$F, G : \text{Int} \rightarrow \text{Set}$$

an ε -interleaving is φ and ψ such that the diagram commutes:

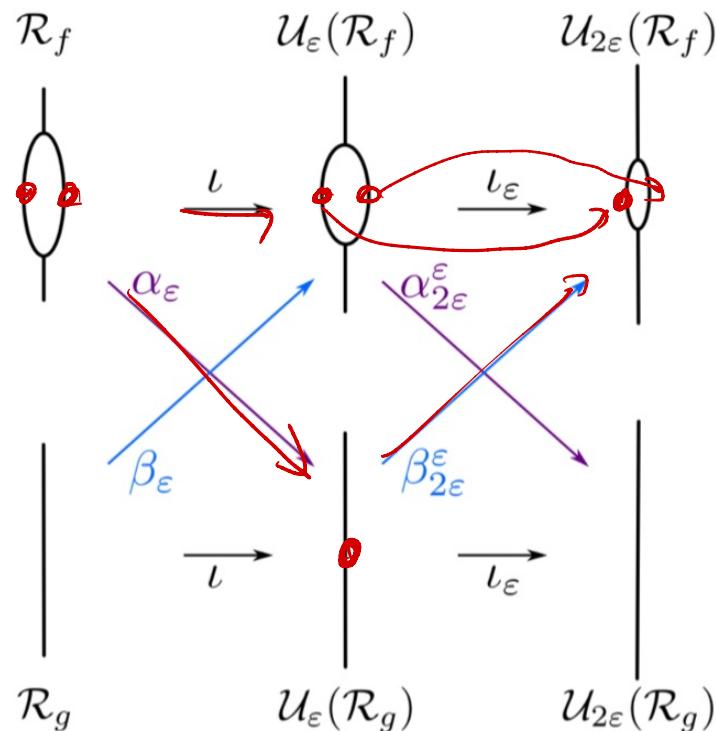
$$\begin{array}{ccccc} F & \longrightarrow & S_\varepsilon F & \longrightarrow & S_{2\varepsilon} F \\ \swarrow & & \downarrow & & \searrow \\ G & \longrightarrow & S_\varepsilon G & \longrightarrow & S_{2\varepsilon} G \end{array}$$

Reeb Interleaving distance is

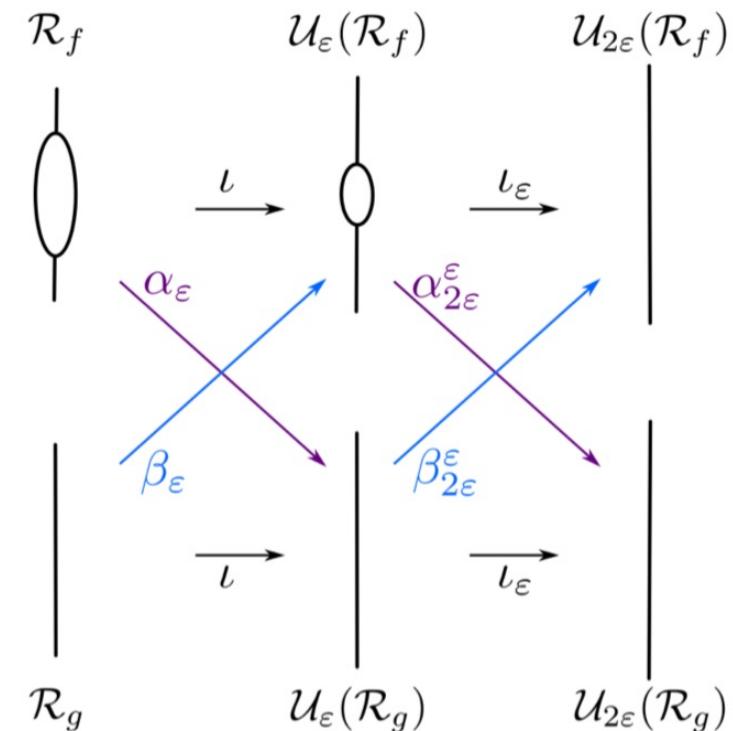
$$d_I(F, G) = \inf \{ \varepsilon \geq 0 \mid \exists \varepsilon\text{-interleaving} \}$$

Example

Bad ε :



Good ε :



Interleaving distance pros & cons:

It is:

- discriminative
- stable

- path component sensitive

- Isomorphism invariant

[Bauer et al 2015]

[de Silva et al, 2016]

Unfortunately, it's Graph-Isomorphism
hard to compute.

↳ Essentially, can use O-interleaved
to check for graph isomorphism.

Functional distortion

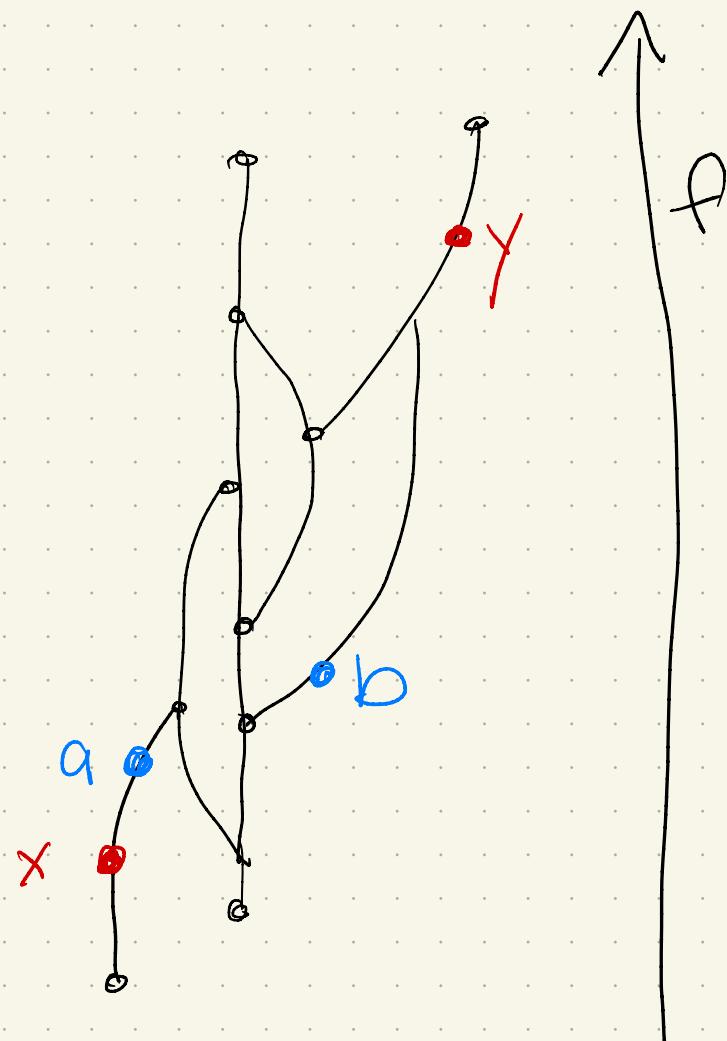
Based on Gromov-Hausdorff idea:

Height of a path?

given path Π ,

$$\max_{x \in \Pi} f(x) - \min_{x \in \Pi} f(x)$$

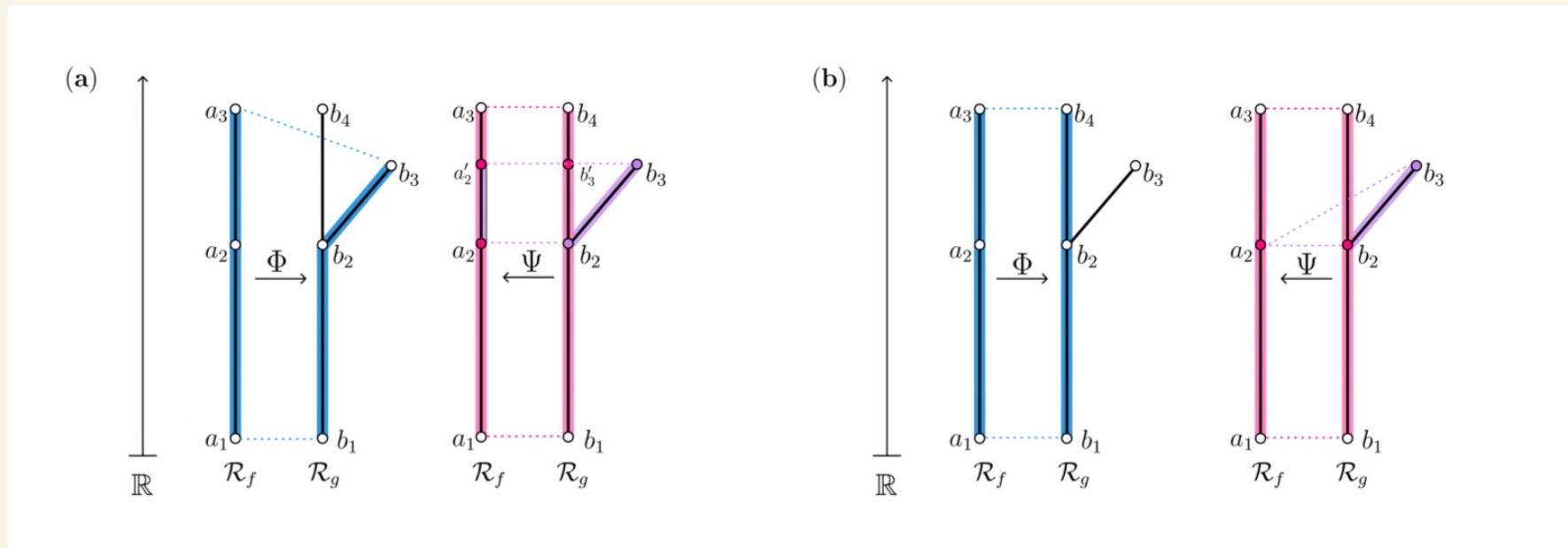
$d(x, y) = \text{height of}$
 $\text{smallest such } \Pi$



Point distortion:

Fix maps $\Psi: \mathcal{R}_f \rightarrow \mathcal{R}_g$
 $\Phi: \mathcal{R}_g \rightarrow \mathcal{R}_f$

(continuous, but not function preserving)



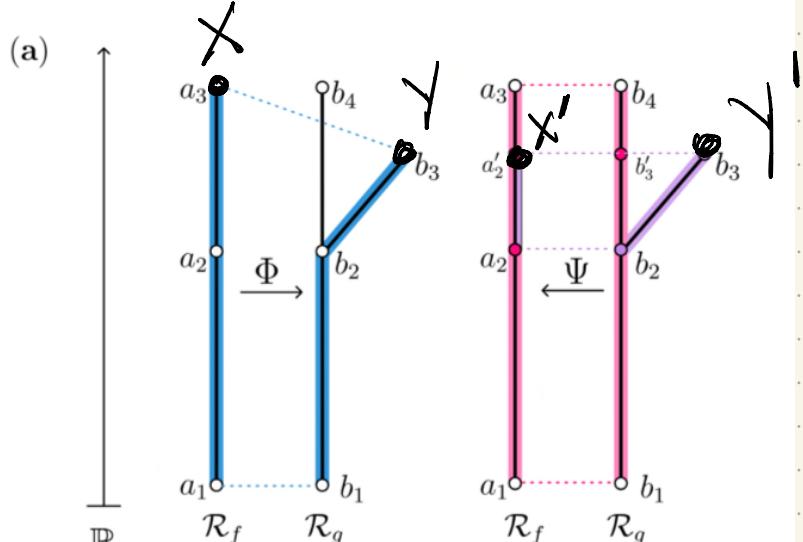
Note: Many choices here!

$$G(\Phi, \Psi) = \{ (x, \Phi(x)) \mid x \in R_f \} \\ \cup \{ (\Psi(y), y) \mid y \in R_g \}$$

Point distortion

For $(x, y), (x', y') \in G(\Phi, \Psi)$,

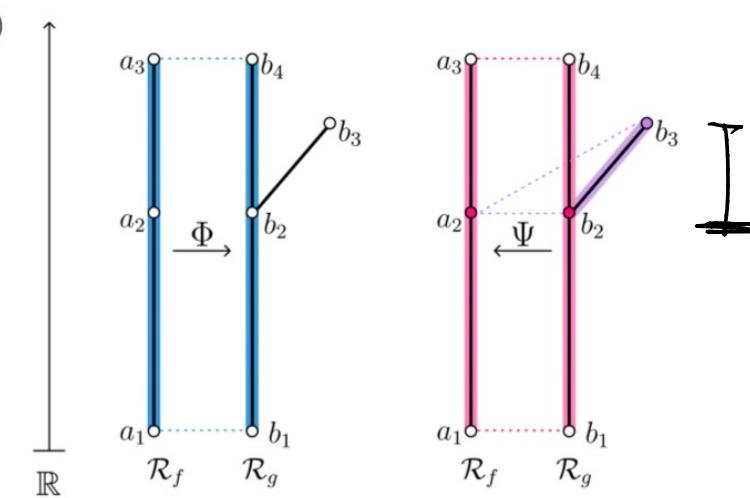
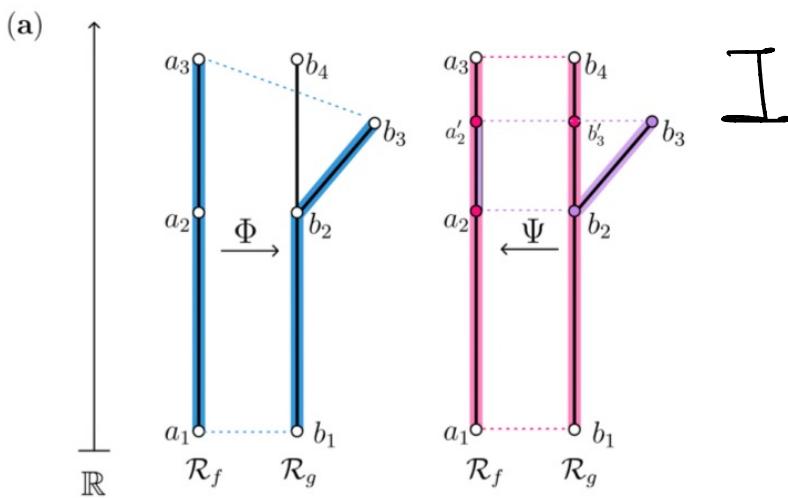
$$\Delta((x, y), (x', y')) = \frac{1}{2} | d_f(x, x') - d_g(y, y') |$$



Map distortion

Supremum over all pairs of points
of the point distortion

Note: all depends upon maps Ψ & Φ !



Functional distortion

$$d_{FD}(R_f, R_g) =$$

$$\inf_{\Phi, \Psi} \max \{ D(\Phi, \Psi), \| f - g \circ \Phi \|_\infty, \| f \circ \Psi - g \|_\infty \}$$

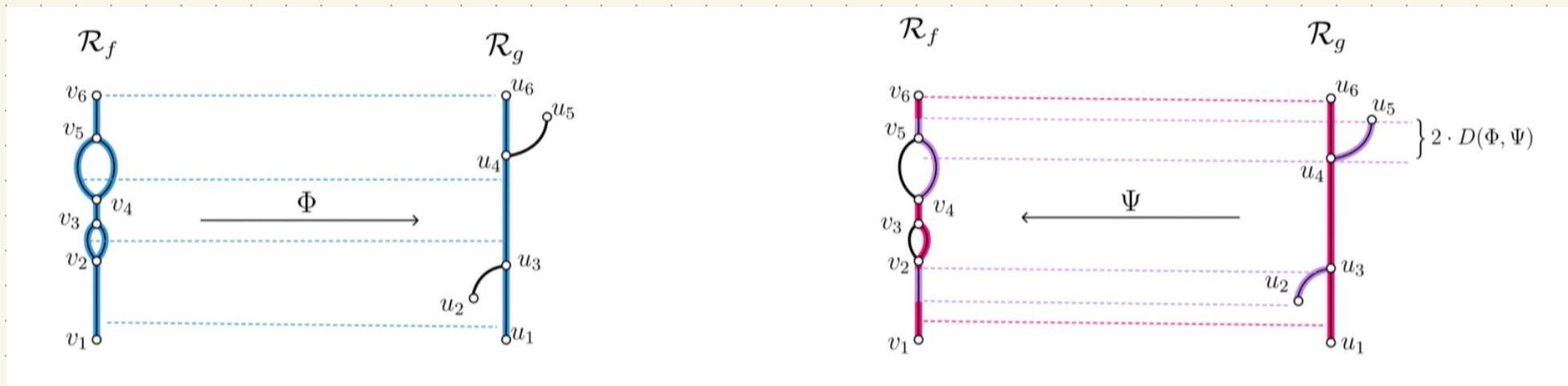
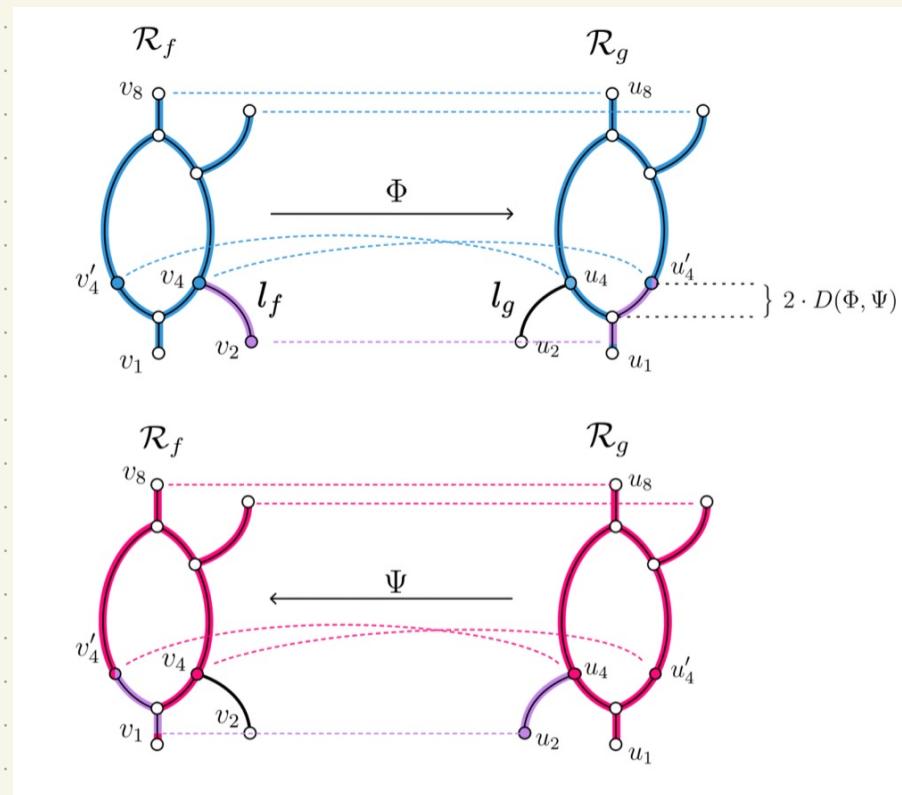
Map

Distortion

(intrinsic
to graph)

global
rescaling

Examples



Functional distortion pros + cons

It is:

- stable

- discriminative

- isomorphism invariant

- path component sensitive

[Bauer et al 2014]

[Bauer et al 2016]

No idea if it's computable in any efficient way.

(or even at all, except on trees!)

Edit distances:

Edit distances are well studied for strings and abstract graphs.

↳ See [Bille 2005] for the many variants on graphs.

In a graph, usually have:

- vertex insertion / deletion
- edge insertion / deletion

and some cost associated with each operation.

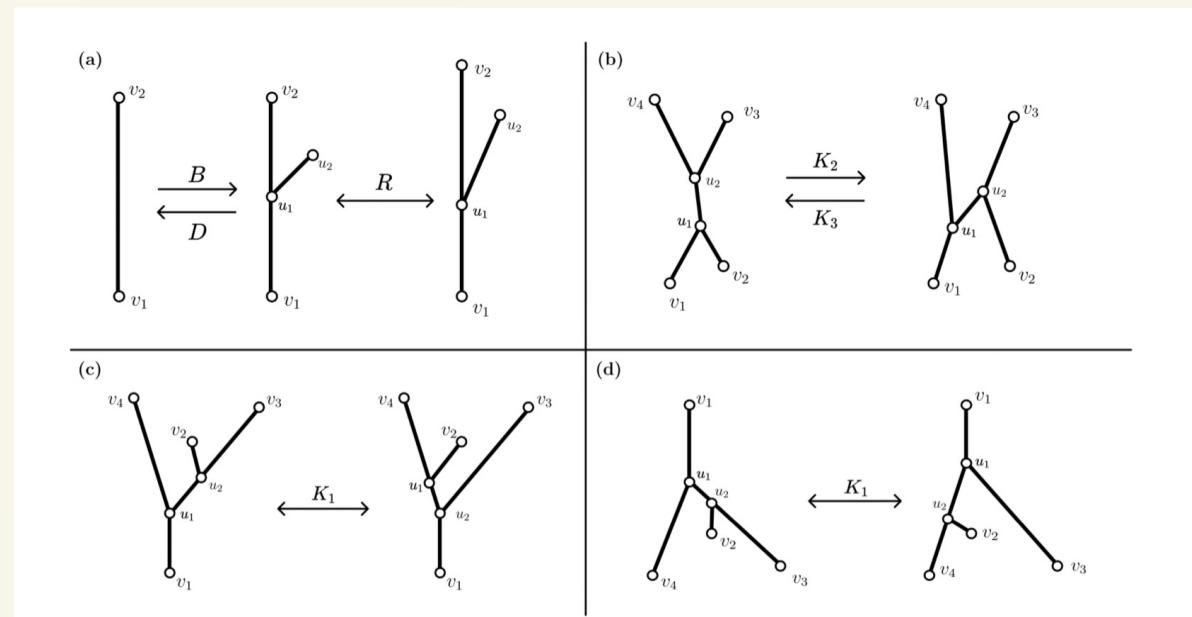
Total Edit distance = $\min \left\{ \sum (\text{edit costs}) \right\}$

Edits:

In Reeb graphs, only certain edits will correspond to topological deformations on original space.

[Di Fabio & Landi, 2012 and 2016]

- Insert
- delete
- relabel
- K-type

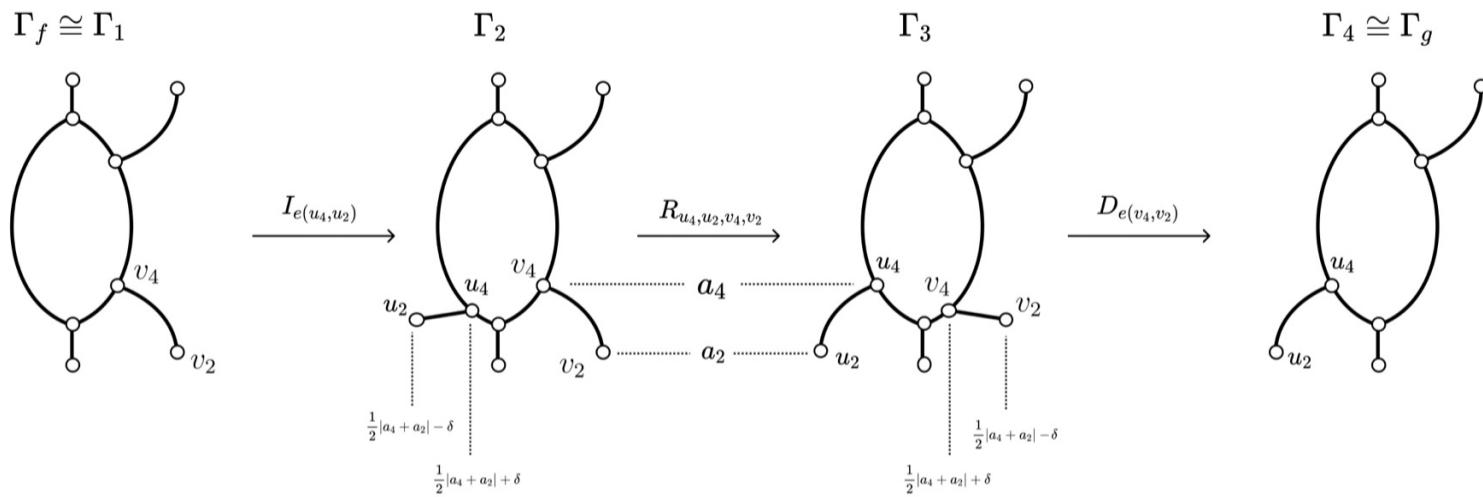


(Reason: Morse theory)

Downside

Could only compare homeomorphic spaces, originally.

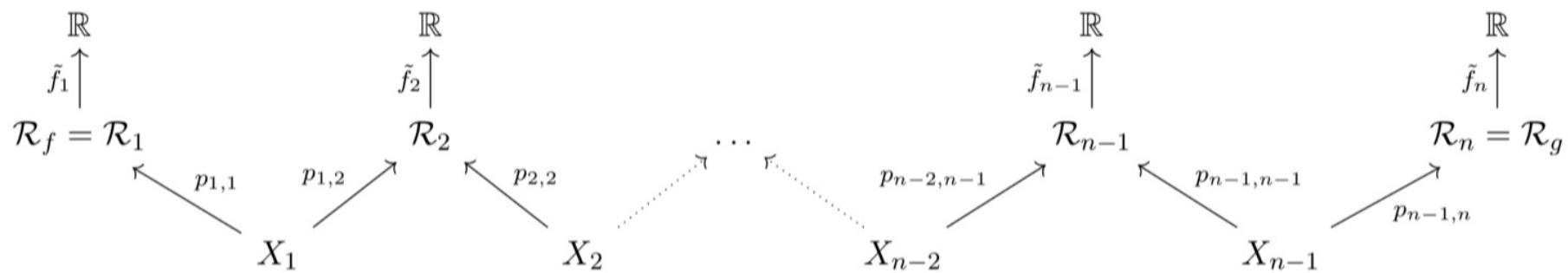
Also a more subtle flaw: can pay less by infinitesimally shifting vertices, since relabels cost max of operations.



Categorical edit distance

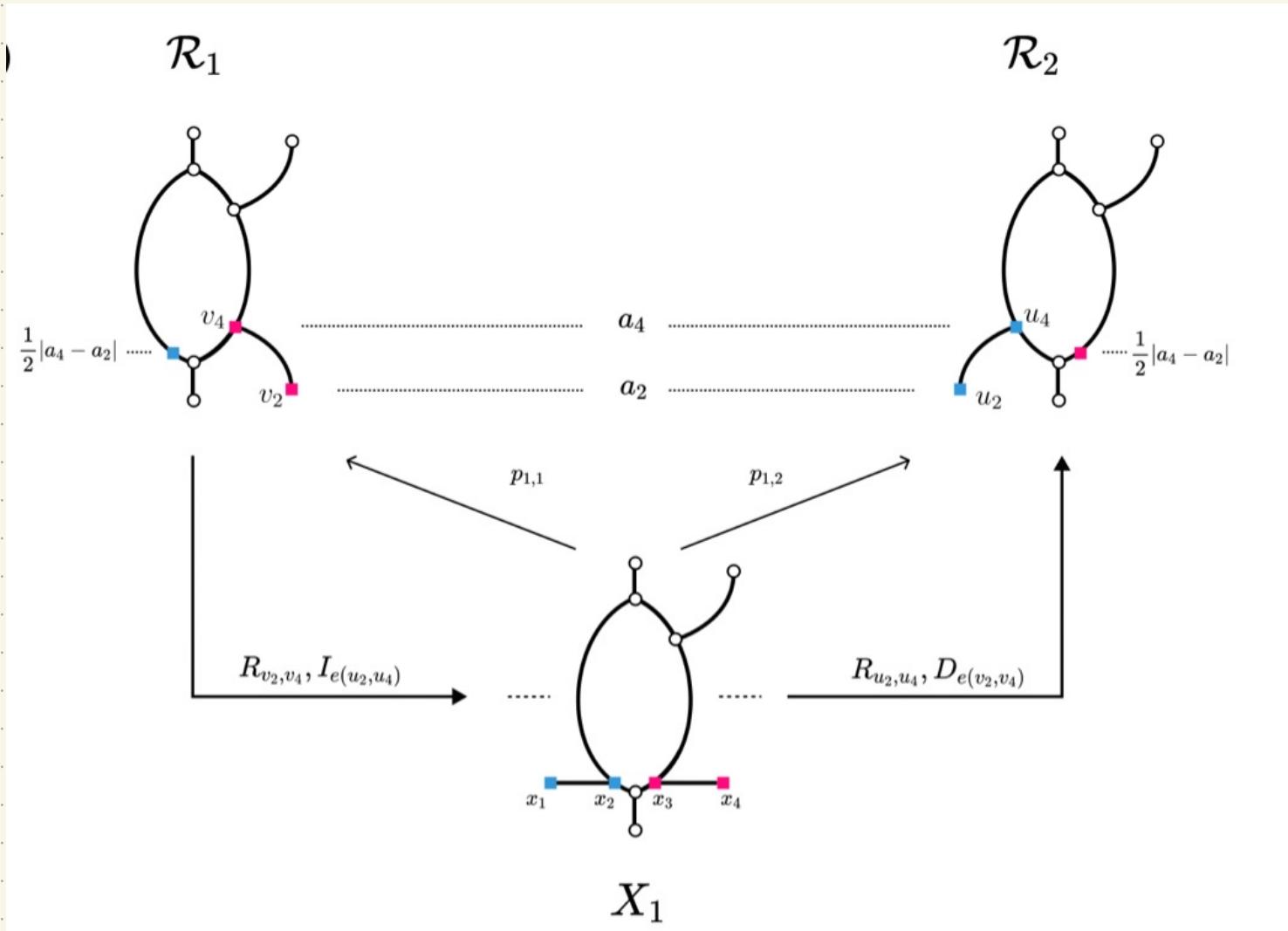
(Bauer et al 2021)

Instead of "direct" edits to Reeb graphs, goal it to find a sequence of topological spaces making this diagram commute.



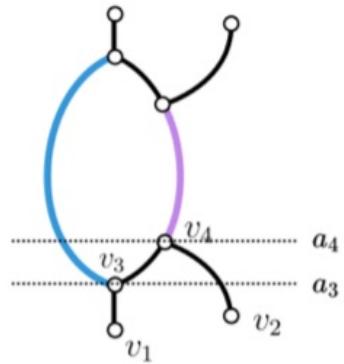
then, pullback (or cone) of this diagram must exist, & can measure how much things move across the maps.

Example

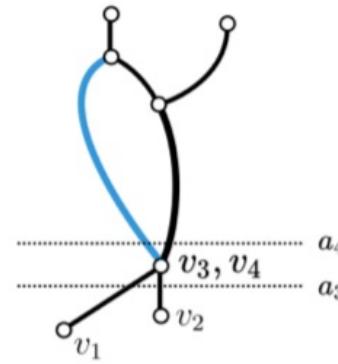


Alternative

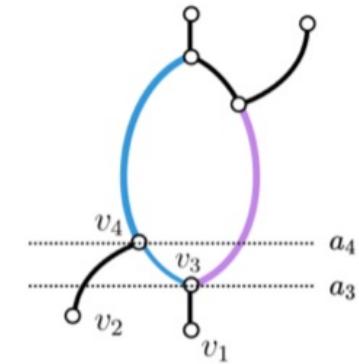
\mathcal{R}_1



\mathcal{R}_2



\mathcal{R}_3



$p_{1,1}$

R_{v_3,v_4}

$p_{1,2}$

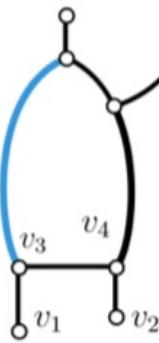
$S_{(v_6,v_3) \rightarrow (v_6,v_4)}$

$p_{2,2}$

$S_{(v_5,v_4) \rightarrow (v_5,v_3)}$

$p_{2,3}$

R_{v_3,v_4}



X_1

X_2

Categorical edit distance pros + cons:

It is:

- stable
- discriminative
- Iso morphism invariant
- path component sensitive

and universal:

for any stable distance S ,

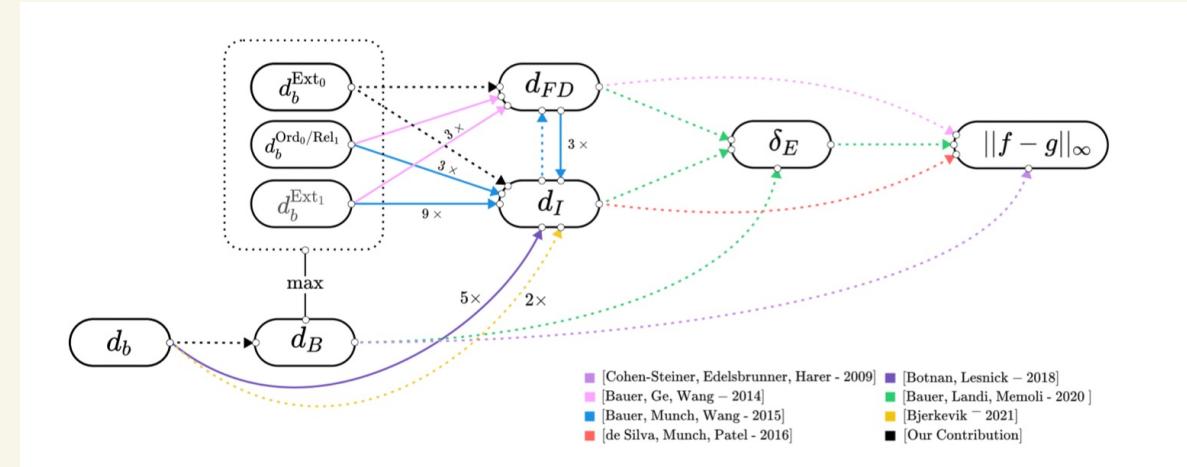
$$d_S(R_f, R_g) \leq d_E(R_f, R_g)$$

[Bauer et al 2020]

Unfortunately, no idea how to
compute it!

Take-aways:

① These are very connected!



In fact, we conclude our survey with a conjecture:

Conjecture 9.3. *The functional distortion distance, interleaving distance, and universal edit distance are equivalent on the space of PL Reeb graphs where the domain \mathbb{X} is simply connected. That is*

$$d_B \leq d_I = d_{FD} = \delta_E.$$

(up to constant factors)

② They do capture different types of features, even though there are connections!

	d_B	d_I	d_I^m	d_{FD}	d_E	δ_E
Stable	[27]	[62]	-	[4]	[35]	[5]
Discriminative	-	[6], 7.24	[22], C.3	[4], 7.20	[35]	[5]
Isomorphism Invariant	-	[62]	[22]	[6], 7.10	[35]	[5]
Path Component Sensitive	-	[62]	[22]	[6]	-	[5], 7.39
Universal	-	-	-	-	-	[5]

Table 1: Table of distance properties. Entry corresponds to a citation where the distance was proved. We supply additional references to statements we contributed in this work which solidify these properties. We denote disproven properties or properties that are not applicable to the given distance with “-”.

We consider
4 simple graph
classes, to find
differences!

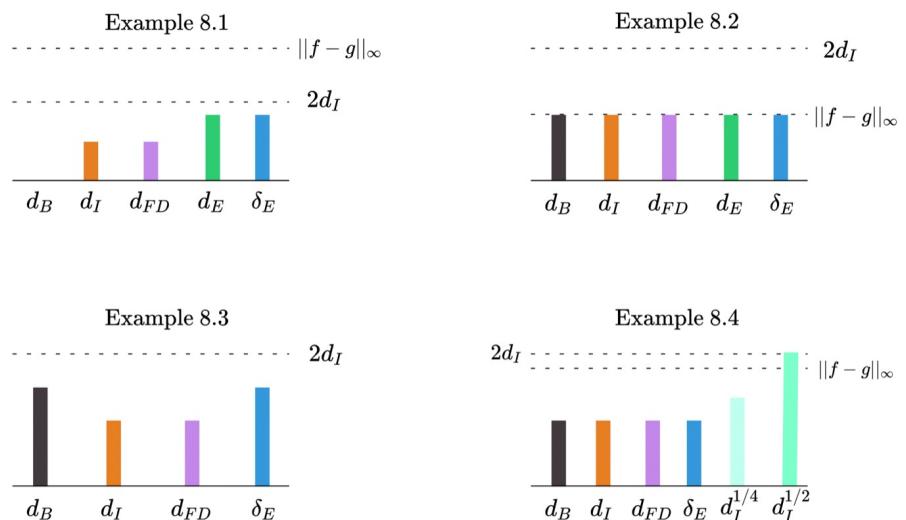
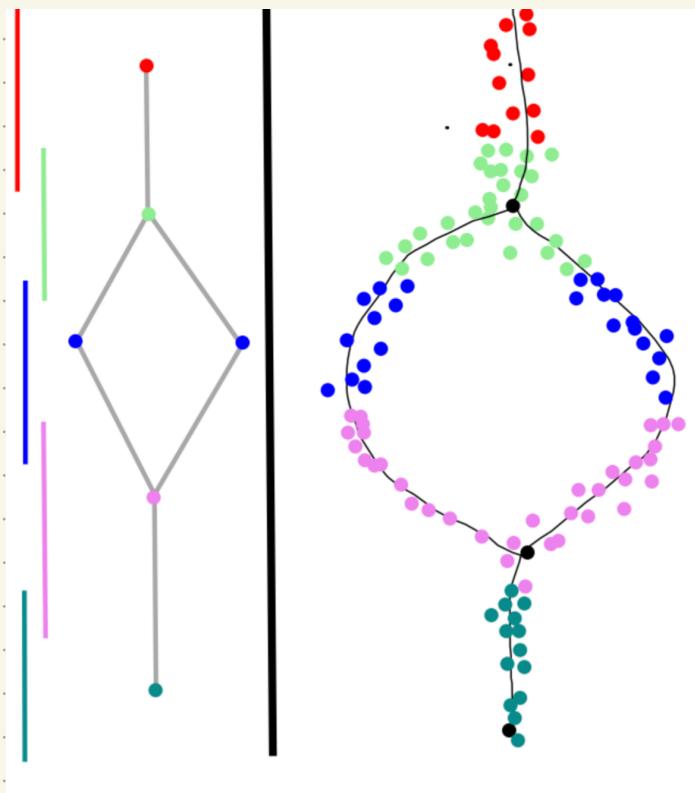


Figure 16: Visual summary of the distance values attained for each example. For Example 4, we show two different choices of m for the truncated interleaving distance to illustrate the affect that m plays on the distance values.

Mapper graphs

Idea: Approximate Reeb graphs

Won't always have a PL-space!
What about point clouds?



Idea:

- Give R-values to dots
- Use a cover of \mathbb{R}
- Cluster into components
+ build a graph

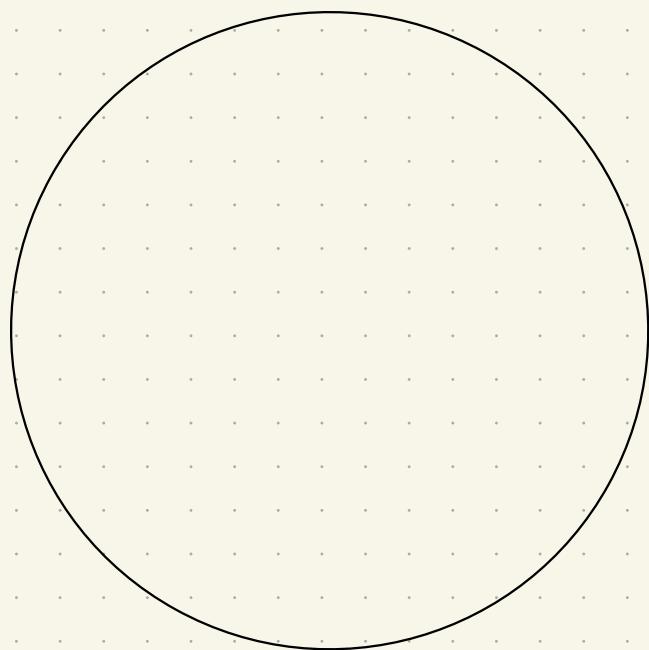
More details:

A cover of a set X is a collection of sets $M = \{U_1, \dots, U_n\}$ st. $X \subseteq \bigcup_i U_i$

Open cover \rightarrow each U_i open

Ex: S^1

Cover:



Let's start on a simplicial complex!