Sorbonne Université

Laboratoire d'Informatique Paris 6

Doctoral Thesis

Discipline : Computer Science

presented by

Keanu Sisouk

to obtain the degree of

Ph.D. of Sorbonne Université

Wasserstein Barycenters of Persistence Diagrams and Applications

---

under the supervision of Julie Delon and Julien Tierny

Defense scheduled on November the 14th, 2025, before the committee :

Vijay Natarajan (Indian Institute of Science Bangalore)	Reviewer
Nicolas Papadakis (CNRS, Bordeaux University)	Reviewer
Carola Doerr (CNRS, Sorbonne University)	Examiner
Bertrand Michel (Centrale Nantes)	Examiner
Julie Delon (University Paris-Cité & ENS Paris)	Advisor
Julien Tierny (CNRS, Sorbonne University)	Advisor

Sorbonne université
LIP6 - Laboratoire d'Informatique de Paris 6
UMR 7606 Sorbonne Université – CNRS
4 Place Jussieu – 75252 Paris Cedex 05 – France

Remerciements

En hommage à A., P. et E.

Je tenais tout d'abord à remercier du fond du coeur mes directeurs de thèse Julie Delon et Julien Tierny. Leur soutient et gentillesse tout au long de ma thèse ont été pour moi un moteur important pour mener à bien mon travail. Leur côté humain pendant toutes les réunions passées ensemble, Julie et Julien me racontant leur vie de tous les jours et les anecdotes passées. Un exemple serait une réunion où Julien nous racontait qu'il était privé de roller après avoir eu un accident l'empêchant de porter son premier enfant dans ses bras. Ou une autre réunion où Julie me demanda des conseils sur des jeux vidéos pour son enfant, notamment sur quel jeu Pokémon était bien... Merci encore pour tout.

I am switching to English for this part, but I also want to thank Vijay and Nicolas for reading my thesis and your feedback, and again sorry to Vijay for the mishap. I also want to thank the both Carola and Bertrand for accepting to be in my committee, you followed the evolution of my thesis as members of my CSI and gave me advices along the way.

Retour en Français, mes remerciements vont ensuite à des personnes avec qui j'ai eu tout particulièrement eu un attachement durant cette thèse. Je remercie Ariane, qui a commencé son stage en même temps que moi, avec qui j'ai pu commérer fréquemment au 8ème étage au MAP5. Tu étais un rayon de soleil durant cette thèse. Je remercie Francesco dans mon bureau au LIP6 qui m'a fait découvrir la cuisine Italienne à mainte reprise, et qui m'a appris diverses phrases en Italiens. Marco avec qui j'ai partagé des moments de solitutes dans le bureau à plusieurs reprises. Mat(t)hieu (oui même sur la CNI on ne sait pas) qui m'a appris plusieurs techniques pour mener à bien cette thèse. Sylvain pour ces visionnages de championnats de League of Legends. Eloi et Ivan avec qui, vers la fin de cette thèse, avions passés des jours fériés à rédiger nos thèses respectives. Je remercie Loïc, qui m'a accueilli plusieurs fois dans ces terres en Bourgogne et pour toutes ces discussions sur le vin.

Avec beaucoup de tristesse je quitte le LIP6 et MAP5 qui ont été presque des "maisons" pour moi. Je remercie tous les doctorants, doctorantes, post-docs et ingénieurs que j'ai pu rencontrer au sein de mon bureau au LIP6, Sébastien, Mohamed, Matéo (même si tu es à Lyon), Eve (même si tu es à Lille), Pierre (mon dieu du code), Danaël, Amaury, Milla, Teodors, Paul, Jules, Raphaël, Mathieu, Martin, Ada, Thomas, Guillaume, Ghiles, Alice, David, Clément, Matthieu D. pour ces moments dans ce bureau qui me manquera. De même pour le MAP5, tout d'abord les membres du bureau au 8ème étage, Clémence, Lena, les deux Rémy (les diablotins), Mariem, Bianca, Beatriz, Lucia, Angie, Léonard, Thibaut, Yassine, Tom pour ces moments partagés isolés au 8ème. Je remercie aussi Alexander,

Mehdi, Antoine M., Herb, Adélie, Lucie, Antoine S., Florian, Marie, Ousmane, Anton, Pierre-Louis, Zoé, Chabane, Carlos, Safa, Yen pour les divers échanges que l'on a eu et vos conseils. J'espère n'avoir oublié personne ce n'est pas très évident

Merci infiniment à amis qui m'ont soutenu tout le long. Cyprien pour tous ces "dates" aux restaurants pendant 4 ans, on en a fait beaucoup mais il en reste encore plein à faire ! Je remercie Fedya sans qui, cette dernière année aurait pu être plus compliquée.

Je n'oublie pas mes proches que je considère comme ma famille, Benji qui m'a soutenu depuis la Corée, Sébastien et Laura ainsi que vos parents qui continuent de m'inviter pour manger chez vous. Léa et ta famille avec qui je continue d'être considéré comme de la famille après tout ce temps. Cécile, ma mentor, ma conseillère et grande soeur, qui m'a permit de trouver cette thèse.

Et pour finir je remercierai jamais assez mon père et ma mère pour tout, d'avoir fait de moi qui je suis aujourd'hui, votre sacrifice, votre amour sans faille et d'être fiers de moi même si mon parcours n'est peut-être pas ce que vous espéreriez. Mon petit frère Kevin qui me soutient même dans nos différents.

Contents

Notations

Notation	Meaning
$\mathbb{R}$	Set of real numbers
$\mathbb{R}^d$	Euclidean space of dimension d
$\mathbb{S}^{d-1}$	Hypersphere of dimension $d-1$
$\mathbb{Z}$	Set of integers
$\mathbb{Z}/2\mathbb{Z}$	Cyclic group of order 2
$\mathcal{R}$	Equivalence relation
$[x]$	Equivalence class
$O(d)$	Orthogonal group of $\mathbb{R}^d$
$\mathcal{M}$	Manifold or submanifold
$\|\cdot\|$	Euclidean norm
$\langle\cdot,\cdot\rangle$	Euclidean scalar product
$x^T,\ A^T$	Transpose of a vector and a matrix
λ	Lebesgue measure on $\mathbb{R}$
$\lambda^{\otimes d}$	Lebesgue measure on $\mathbb{R}^d$
$L^2(\mathbb{S}^{d-1})$	Quotient space of squared integrable functions
$\mathcal{C}^k$	Set of k-differentiable functions
s_{d-1}	Uniform measure on $\mathbb{S}^{d-1}$
$H^\alpha(\mathbb{S}^{d-1})$	Sobolev space on $\mathbb{S}^{d-1}$
$f : A \to B$	Application from A to B
∇f	Gradient of a function f in $\mathbb{R}^d$
$\nabla_{(d-1)} f$	Gradient of a function f on the hypersphere $\mathbb{S}^{d-1}$
Δf	Amplitude of a scalar function f
$\partial^{\|j\|} f$	Mixed partial derivative of a function f
C_f	Lipschitz constant of f
d_X	Metric on a space X
Δ	Diagonal in $\mathbb{R}^2$
π_Δ	Orthogonal projection application on Δ
$\mathcal{O}(a_n)$	Asymptotically bounded by a_n
Σ_m	Set of vectors of size m with positive entries summing to one
Π_A	Orthogonal projection on a convex set A
θ	Point in $\mathbb{S}^{d-1}$
$\sigma_\theta,\ \tau_\theta$	Permutations on the line of direction θ
π_θ	Orthogonal projection application on a line of direction θ
$\mathbb{P}(A)$	Probability of event A

$\mathbb{E}[X]$	Expectation of a random variable X
$\mathbb{V}[X]$	Variance of a random variable X
$\mathcal{U}(A)$	Uniform distribution on A
$\mathcal{N}(0,1)$	Standard normal distribution
$X \sim \mu$	X is a random variable with distribution μ
$T\#\mu$	Push forward of μ by the application T
$\Pi(\mu,\nu)$	Set of measures having μ and ν as marginals
δ_x	Dirac at location $x \in \mathbb{R}^d$
W_q	q-Wasserstein distance
SW_2	Sliced Wasserstein distance
SSW	Spherical Sliced Wasserstein distance
$\mathcal{O}_{\mathbb{P}}(a_n)$	Stochastically bounded by a_n
$\mathcal{M}_a$	Sublevel set of a function f at a
$\mathcal{K}$	Simplicial complex
C_k	Group of k-chains
$\partial_k \sigma$	Boundary of a k-chain σ
$Z_k(\mathcal{K})$	Group of k-cycles of $\mathcal{K}$
$B_k(\mathcal{K})$	Group of k-boundaries of $\mathcal{K}$
$H_k(\mathcal{K})$	k-th homology group of $\mathcal{K}$
β_k	k-th Betti number of $\mathcal{K}$
$St(v)$, $\overline{St(v)}$	Star and Closed star of a vertex v
$Lk(v)$, $LLk(v)$, $ULk(v)$	Link, lower link and upper link of a vertex v
K_i	i-th step of a filtration
$f_k^{i,j}$	Homomorphism between $H_k(K_i)$ and $H_k(K_j)$
$H_k^{i,j}$	k-th persistent homology groups given $H_k(K_i)$ and $H_k(K_j)$
$\boldsymbol{\lambda}$	vectors of positive entries summing to one
E_D	Wasserstein dictionary energy
E_W	Weight energy
E_A	Global atoms energy
e_a	Pointwise atom energy
$c_q(x,y)$	Ground cost between x and y
$\mathfrak{b}_q$	Formalized ground barycenter for the cost c_q
E_F	Fréchet energy

Chapter 1

Introduction

Data, in its simplest form, is a collection of values (a collection of datum) that can become *information* about a specific phenomenon through interpretation by an external medium (observer). Data form the foundation of modern scientific inquiry. They represent measured observations, experimental results, and simulations collected across disciplines—from climate and medical science to particle physics and artificial intelligence. With advances in sensors, computing, and digital communication, the scale and complexity of data have increased, leading to what is often called the big data era.

This thesis' goal is developing tools that ease the analysis and visualization of data through the lens of Topological Data Analysis (TDA), powered by the theory of Optimal Transport (OT). TDA offers a robust mathematical framework for capturing the underlying shape of data in a way that is easy to interpret. By leveraging tools from algebraic topology, such as persistent homology, TDA can aid in data analysis by revealing global patterns and features that are resistant to noise and invariant under continuous transformations. As such TDA provides mathematical objects called *topological abstractions*, which extract those patterns and features of interest in a concise manner, hence facilitating their analysis. Moreover, modern results from OT can be applied in combination to topological abstraction to further enhance their descriptive power.

1.1 Context & Motivations

1.1.1 Data acquisition, Analysis and Visualization

As stated before, the scale and complexity of data have increased significantly, leading to what is often referred to as the data boom. This explosion in data generation has transformed scientific research, enabling more accurate models, deeper insights, and interdisciplinary discoveries. However, it also presents significant challenges. The analysis bottleneck arises as the volume, dimensionality, and geometry complexity increase, making it difficult to extract meaningful patterns efficiently.

Despite these challenges, this evolution in data generation has changed the way we make informed decisions based on information interpreted through large-scale data analysis. In that context, analysis methods from machine learning and data science in general have been either developed or further improved. Indeed, methods such as k-means or support vector machines, which classify large populations into groups and reveal trends

in data, are widely used in modern applications across various fields, such as healthcare.

One way of analyzing data is through simple and interpretable visualization of the data being studied. Indeed, visualization is a first and natural interaction with data, helping to uncover explicit patterns and salient features. Thus, in response to the growing complexity of data, developing more advanced visualization tools has become a necessity. Creating more dynamic, interactive, interpretable, and—importantly—simple visualizations leads to a deeper understanding of complex data during analysis. Visualization is not only a tool for analyzing data; it is also a tool for conveying information interpreted from data.

1.1.2 Topological Data Representations

The geometrical complexity of modern data makes interactive exploration and analysis difficult, which challenges the interpretation of the data by the users. This motivates the creation of expressive data abstractions, capable of encapsulating the main features of interest of the data into simple representations, visually conveying the main information to the user.

TDA [58] is a family of techniques which precisely addresses this issue. It provides concise topological descriptors of the main structural features hidden in a dataset. The relevance of TDA for analyzing scalar data, its efficiency and robustness have been documented in a number of visualization tasks [88]. Examples of successful applications include turbulent combustion [34, 78, 105], material sciences [63, 80, 81, 176], nuclear energy [117], fluid dynamics [96, 126], bioimaging [9, 24, 40], chemistry [20, 72, 131, 132] or astrophysics [171, 177].

Among the different topological descriptors studied in TDA (such as the merge and contour trees [2, 38, 39, 73, 115, 183], the Reeb graph [22, 57, 74, 139, 140, 185], or the Morse-Smale complex [33, 50, 59, 60, 71, 79, 161, 170]), the Persistence Diagram (Fig. 2.9) is a particularly prominent example. As described in Sec. 2.3, it is a concise topological descriptor which captures the main structural features in a dataset and assesses their individual importance.

1.1.3 Tools from Optimal Transport

Optimal allocation of resources is one of the most primal problem in our every day life. This problematic falls into the theory of *optimal transport*. Being initially presents in important field such as economics [168], it has gained a new wind for its application in different fields in mathematics. OT is especially acclaimed for its geometric relevance in comparing probability distributions. Having gathered a lot of theoretical work [164, 192], it has also proved to be relevant in numerous applied domains in the last fifteen years, such as image comparison [158], image registration [67], domain adaptation [47], generative modeling [12, 76, 163], inverse problems in imaging [90] or topological data analysis [58, 148], to name just a few.

1.1.4 The ERC-TORI Project

The big data era has brought datasets that are more complex and on a larger scale, making their analysis increasingly difficult and leading to a computational bottleneck. At

the same time, the demand for storage space is growing rapidly, requiring advanced data structures and raising concerns about sustainability, cost, and long-term accessibility. Addressing these issues is essential to ensure that data continues to advance scientific progress rather than hinder it. To address these challenges, the TORI [1] Project (In-Situ Topological Reduction of Scientific 3D Data) focuses on reducing the computational bottleneck by using topological abstractions of raw data. These topological abstractions have significantly smaller memory footprints compared to the original data. There are two principal axes of research: scaling topological methods to larger datasets, and developing analysis methods based on the data's topological abstraction.

This thesis focuses on the latter. Its goal is to adapt tools from OT theory to analyze a specific type of topological descriptor: the *persistence diagram*. More specifically, it aims to find a concise and meaningful topological representation of an ensemble of datasets that offers *good* computation time and robustness.

1.1.5 The Topology ToolKit

The Topology ToolKit (or TTK) [2] [184] is an open source C++ library for topological data analysis and visualization. Its goal is developing and providing efficient topological methods applied to data analysis. It is fully integrated into ParaView [6], a popular open-source software for data analysis and 3D visualization, developed by Kitware around the Visualization ToolKit (VTK) library. As a ParaView plugin, TTK is very accessible to end users working in topological data analysis, enabling them to use the methods developed in TTK without needing to dive into the C++ code −while having a visualization of the end result at the same time. Users can even use TTK's features in Python. All the research work that is presented in this thesis has been implemented in TTK. Thus it is accessible to the general public.

1.2 Problem formulation

Until now, we have given some general context about topological data analysis along with the project in which this thesis takes place. Recall that in most cases, end users need to analyze not just a single dataset, but many. As a result, a large amount of physical space must be allocated to store these ensembles of datasets. To address this challenge, the goal of the TORI project is to analyze such ensembles of datasets through the lens of their topological abstractions, persistence diagrams, in our case, as mentioned earlier. From there, many questions arise regarding how to find effective representations of ensembles of persistence diagrams and how to analyze them. We now further formulate the problematic this thesis and project addresses.

[1] `https://erc-tori.github.io/`
[2] `https://topology-tool-kit.github.io/`

1.2.1 Variability analysis and encoding of ensemble of persistence diagrams

The development of tools for describing an *average* of an ensemble of topological abstractions—enabling the adaptation of machine learning methods such as k-means—has led to the analysis of variability within such ensembles.

Initially, efforts to find alternative representations aimed to combine topological abstractions with standard machine learning methods. However, these approaches generally *vectorize* topological abstractions [4, 10, 37, 112, 160], allowing them to be used as input for classic machine learning frameworks. While effective to some extent, this strategy has several downsides: it can introduce approximation errors, and operations on the resulting vectors often lack meaningful interpretation with respect to the original topological descriptors.

Later, other works took a different path, avoiding the need to find such representations [150]. This thesis follows that approach, first focusing on providing a method for ensembles of persistence diagrams that is easy to understand and apply.

1.2.2 Accelerating and fortifying methods for topological data analysis

Some tools used in topological data analysis have computational complexities that can grow rapidly with the scale of the initial topological abstractions. A simple example is the so-called Wasserstein distance from OT theory, which has a cubic time complexity with respect to the size of its inputs. This rate can hinder the interactive aspect of data analysis. Additionally, several methods for analyzing ensembles of topological abstractions can be sensitive to outliers, potentially leading to deviations from expected results. This thesis also focuses on proposing alternatives from OT theory that can accelerate and strengthen existing methods for persistence diagrams

1.3 Contributions

This thesis addresses the previously mentioned challenges through the following contributions: developing a method for encoding an ensemble of persistence diagrams, and exploring ways to strengthen and improve this method.

1.3.1 Encoding of ensembles of persistence diagrams

Our initial focus was on developing a simple method for encoding ensembles of persistence diagrams. To achieve this, we drew inspiration from previous work on optimal transport [167], adapting a non-linear dictionary encoding based on the Wasserstein barycenter of histograms to the context of persistence diagrams. The output of this method is a set of representatives of the original ensemble, allowing users to inspect the main features of interest across the entire dataset. The goal is to ensure that the analysis of the representatives yields the same insights as the analysis of the full ensemble. This contribution is detailed in Chapt. 3.

1.3.2 Robust representative of persistence diagrams

A barycenter of point clouds is a generalization of the average—or more formally, the mean—of real numbers. As such, it naturally shares the same drawback: sensitivity to outliers. For example, if you compute the mean of a set of numbers and then add a number that is significantly larger than the rest, the new mean will shift. In light of this issue, work on generalized Wasserstein barycenters has emerged [181]. We demonstrate that this type of barycenter can be applied to ensembles of persistence diagrams. This contribution is detailed in Chapt. 4.

1.3.3 Accelerating tools for analyzing an ensemble of persistence diagrams

The main tool—the Wasserstein distance—for comparing persistence diagrams has a significant drawback: its computation time increases rapidly with the size of the diagrams [142]. This issue becomes more pronounced as the size of the persistence diagrams grows with the scale of the original data. One way to address this bottleneck is through Sliced Optimal Transport (Sliced OT) theory, particularly its main variant of the Wasserstein distance: the Sliced Wasserstein distance. In an effort to further accelerate the computation of the Sliced Wasserstein distance for persistence diagram comparison, we transformed this work into a practical user's guide for Sliced OT. This contribution is detailed in Chapt. 5.

1.4 Outline

This manuscript is structured the following way:

- We begin by introducing some general reminders for Optimal Transport and Topological Data Analysis in Chapt. 2.
- In Chapt. 3, we present our method for dictionary encoding of persistence diagrams based on the Wasserstein distance and barycenter.
- In Chapt. 4, we illustrate the utilization of a more robust Wasserstein barycenter for persistence diagrams.
- In Chapt. 5, we present a user's guide for sampling strategies for Sliced OT in the general case.
- Finally, Chapt. 6 gives the end of this thesis by presenting an overview of its different contributions and limitations, and discussing about further generalization and improvement of this Wasserstein dictionary method.

1.5 List of Publications and Submissions

The contributions that we described led to the following publications and submission:

- **Keanu Sisouk**, Julie Delon and Julien Tierny. Wasserstein Dictionaries of Persistence Diagrams. In *IEEE Transaction on Visualization and Computer Graphics (IEEE TVCG)*, volume 30, 2024. Presented at IEEE VIS 2024.
- **Keanu Sisouk**, Julie Delon and Julien Tierny. A User's Guide to Sampling Strategies for Sliced Optimal Transport. In *Transaction on Machine Learning Research (TMLR)*, 2025.
- **Keanu Sisouk**, Eloi Tanguy, Julie Delon and Julien Tierny. Robust Barycenters of Persistence Diagrams. Submitted to *IEEE TVCG* as a short paper.

Other publications during this thesis:

- Mohamed Kissi, **Keanu Sisouk**, Joshua Levine and Julien Tierny. Topology Aware Neural Interpolation of Scalar Fields. In *TopoInVIS*, 2025.

Chapter 2

Preliminaries

2.1 General prerequisites

This section introduces the elementary prerequisites to understand the different notions detailed in Sec. 2.2, Sec. 2.3 and in the rest of this thesis.

2.1.1 Quotient Groups, Topology & Geometry, Homotopy

2.1.1.1 Quotient Groups

Definition 2.1 : We call a *group* any pair $(G, \mathcal{L})$ with G a set and $\mathcal{L} : \begin{cases} G \times G \to G \\ (x, y) \mapsto x\mathcal{L}y \end{cases}$ an application called an intern law, such that:

- $\mathcal{L}$ is associative: $(x\mathcal{L}y)\mathcal{L}z = x\mathcal{L}(y\mathcal{L}z)$, $\forall x, y, z \in G$,
- $\mathcal{L}$ admits a neutral element (or identity element): $\exists e \in G$ s.t. $\forall x \in G,\ x\mathcal{L}e = e\mathcal{L}x = x$,
- every element admits an inverse: $\forall x \in G, \exists! y \in G$ s.t. $x\mathcal{L}y = y\mathcal{L}x = e$.

If G is finite we call its cardinal the order of G. For x in G we denote by x^{-1} its inverse and we denote by e_G its neutral element.

Definition 2.2 : A generating set of a group $(G, \mathcal{L})$ is a subset noted $\langle G \rangle$ such that for every $x \in G$ there exist $a_1, \ldots, a_n \in \langle G \rangle$ such that $x = a_1\mathcal{L} \ldots \mathcal{L}a_n$ with $n \in \mathbb{N}^*$.
We call the rank of G, noted $\mathrm{rk}(G)$, the cardinal of its smallest generating set.

Definition 2.3 : A group $(G, \mathcal{L})$ is abelian if for all $x, y \in G,\ x\mathcal{L}y = y\mathcal{L}x$.

Example 2.1 : A simple example of group is $(\mathbb{Z}, +)$. Its neutral element is 0, the inverse of an element is its opposite, it does not have an order as $|\mathbb{Z}| = +\infty$, $\mathrm{rk}(\mathbb{Z}) = 1$ and it is abelian.

Definition 2.4 : Let $(G, \mathcal{L})$ be a group and $H \subset G$. H is a subgroup of G if:

- $H \neq \emptyset$,

- $\forall x, y \in H,\ x\mathcal{L}y^{-1} \in H$.

Definition 2.5 : Let $(G, \mathcal{L}_G)$ and $(H, \mathcal{L}_H)$ be two groups. Let $f : (G, \mathcal{L}_G) \to (H, \mathcal{L}_H)$, we say that f is a homomorphism if and only if:

$$\forall x, y \in G,\ f(x\mathcal{L}_G y) = f(x)\mathcal{L}_H f(y).$$

We call an isomorphism any homomorphism that is a bijection, i.e. for any $y \in H$ there is a unique $x \in G$ such that $f(x) = y$. In this case we write $G \simeq H$.

Definition 2.6 : Let $(G, \mathcal{L}_G)$ and $(H, \mathcal{L}_H)$ be two groups, and $f : (G, \mathcal{L}_G) \to (H, \mathcal{L}_H)$ a homomorphism. We call kernel of f the subgroup of G defined as:

$$\mathrm{Ker}(f) = f^{-1}(\{e_H\}) = \{x \in G \mid f(x) = e_H\}.$$

We call image of f the subgroup of H defined as:

$$\mathrm{Im}(f) = f(G) = \{f(x) \mid x \in G\} = \{y \in H \mid \exists x \in G, f(x) = y\}.$$

Definition 2.7 : A subgroup H of G is called a normal subgroup if:

$$\forall x \in H, \forall y \in G,\ y\mathcal{L}_G x\mathcal{L}_G y^{-1} \in H.$$

Proposition 2.1 : If a group $(G, \mathcal{L}_G)$ is abelian, then every subgroup is normal.

Definition 2.8 : An equivalence relation on a set X, is a relation $\mathcal{R}$ between elements of X verifying those following conditions for every $x, y, z \in X$:

- reflexivity: $x\mathcal{R}x$,
- symmetry: if $x\mathcal{R}y$, then $y\mathcal{R}x$,
- transitivity: if $x\mathcal{R}y$ and $y\mathcal{R}z$, then $x\mathcal{R}z$.

We usually denote by $\sim$ an equivalence relation. For x in X, we call the set of elements equivalent to x an equivalence class, and x is called its *representative*.

Definition 2.9 : Let $(G, \mathcal{L})$ be a group and H a subgroup of G. For $x \in G$ we denote the (left) coset associated to H as: $xH = \{x\mathcal{L}y \mid y \in H\}$.

Proposition 2.2 : The cosets of G are equivalence classes for the relation equivalence $x \sim y \iff y \in xH \iff y = x\mathcal{L}z$ for $z \in H$.

Remark 2.1 : Intuitively, this relation $x \sim y$ can be seen as *"y is colinear to x"*.

Proposition 2.3 : For H a subgroup of G, the cosets of H form a partition of G, i.e. $\forall x \in G, xH \neq \emptyset$, $\bigcup_{z \in G} zH = G$, and two cosets xH and yH are either disjoint or equal.

Definition 2.10 : For X, Y two subsets of a group G we define the intern product as $XY = \{x\mathcal{L}y \mid x \in X, y \in Y\}$.

Definition 2.11 : For H a subgroup of G, we denote the set of cosets by $G/H = \{gH \mid g \in G\}$ and we call it *quotient set* of G by H.

Intuitively, a quotient set G/H is a set of equivalence classes where we identify each element of G to an equivalence class. Meaning that all elements in this equivalent class are identified by the same element, its representative, which they are "colinear" to.

Theorem 2.1 : Let G be a group and H a normal subgroup of G. Then the intern product defines an intern law between cosets of H such that G/H is a group. For an element $x \in G$, we denote its equivalent class (or coset) $[x]$ in G/H.

Remark 2.2 : Notice that we have $[e_G] = H$ as $y \sim e_G \iff y \in e_G H = \{e_G \mathcal{L} z | z \in H\} = H$.

Example 2.2 : A classic quotient group is $\mathbb{Z}/2\mathbb{Z}$, which has two cosets: the even integers (equal 0 modulo 2) and the odd integers (equal 1 modulo 2). Indeed for $n \in \mathbb{Z}$: if n is even then $n = 0 + 2 \times k$ for $k \in \mathbb{Z}$, else there is $k \in \mathbb{Z}$ such that $n = 1 + 2 \times k$. Thus $\mathbb{Z}/2\mathbb{Z} = \{[0], [1]\}$, or simply $\{0, 1\}$ (with an abuse of notation) with the addition modulo 2 noted $+_{\mathbb{Z}/2\mathbb{Z}}$.

2.1.1.2 Topology and geometry

Definitions 2.1 : Let X be a set, a topology on X is a collection of subsets of X, noted $\mathcal{T}(X)$ such that:

- $\emptyset \in \mathcal{T}(X)$ and $X \in \mathcal{T}(X)$,
- any union of elements of $\mathcal{T}(X)$ is still in $\mathcal{T}(X)$ (stable by union),
- any finite intersection of elements of $\mathcal{T}(X)$ is still in $\mathcal{T}(X)$ (stable by finite intersection).

We call X a topological space and an element of $\mathcal{T}(X)$ an open set.

Definition 2.12 : A base of open sets of a topological space X, is a collection $\mathcal{B}$ of open such that for any $\omega \in \mathcal{T}(X)$, there is a subset $B \subset \mathcal{B}$ such that $\omega = \cup_{b \in B} b$.

A simple example of a topological space is $\mathbb{R}^d$. An open set of $\mathbb{R}^d$ inherits its topology and thus is itself a topological space. If a topological space X is equipped with a metric (distance) d_X then it is called a metric space.

Definition 2.13 : Let (X, d_X) and (Y, d_Y) be two metric spaces, let $f : X \to Y$ be a function. f is continuous if for all $\epsilon > 0$, for all $x \in X$, there is $\eta > 0$ such that for all $y \in X$, $d_X(x, y) \leq \eta$ implies $d_Y(f(x), f(y)) \leq \varepsilon$.

Definitions 2.2 : Let X, Y be open sets of $\mathbb{R}^d$, we say that $f : X \to Y$ is differentiable on $x \in X$ if there is a linear application $df_x : \mathbb{R}^d \to \mathbb{R}^d$ called differential such that $f(x + h) = f(x) + df_x(h) + o(\|h\|)$ when $h \to 0$. If f is differentiable on any $x \in X$ then we say that f is differentiable on X. If df is continuous then we say that f is continuously differentiable. We denote by $\mathcal{C}^1(X, Y)$ the set of continuously differentiable functions.

We say that f is 2-differentiable if its differential is differentiable, etc.. We denote by $\mathcal{C}^k(X, Y)$ the set of continuously k-differentiable functions. We denote by $\mathcal{C}^\infty(X, Y)$ the set of smooth (infinitely differentiable) functions.

Definitions 2.3 : A function between two metric spaces is called a homeomorphism if it is continuous and a bijection and its inverse is continuous. Two topological spaces are said to be homeomorphic if there exists a homeomorphism between them.

A $\mathcal{C}^k$-diffeomorphism is a function that is differentiable k times, is a bijection such that its inverse is also differentiable k times.

Homeomorphisms and diffeomorphisms are important because two spaces that are homeomorphic or diffeomorphic have the same topology.

Example 2.3 : In the real line $\mathbb{R}$, $[0, 1]$ and $[0, 0.5[\cup]0.5, 1]$ are not homeomorphic. In $\mathbb{R}^3$, a sphere and a torus (donut) are not homemorphic. However a torus (donut) and a coffee mug with a handle are homeomorphic.

Definitions 2.4 : A topological space that has a countable base of open sets that is dense is called a separable space. A topological space that is separable and admits at least one metric d such that it is complete (i.e. any Cauchy sequence converges) is called a *Polish space*.

Definition 2.14 : Let X be a topological space and $x \in X$, a neighbourhood of x is a subset $V \subset X$ such that there exist an open O containing x with $O \subset V$. We denote by $\mathcal{V}(x)$ the set of neighbourhood of x.

Definition 2.15 : A topological space X is said to be separated if for $x, y \in X$, $x \neq y \implies \exists V, W \in \mathcal{V}(x) \times \mathcal{V}(y), V \cap W = \emptyset$.

Proposition 2.4 : A topological space endowed with a metric (distance) d is always separated.

Definition 2.16 : A *manifold* is a topological space $\mathcal{M}$ such that:

- $\mathcal{M}$ is separated and has a countable base of open sets,
- any point of $\mathcal{M}$ has an open neighbourhood that is homeomorphic to an open set of $\mathbb{R}^d$.

Definition 2.17 : A $(\mathcal{C}^k)$ r-submanifold $\mathcal{M}$ of $\mathbb{R}^d$ is a toplogical space such that for any $x \in \mathcal{M}$ there exist $V \in \mathcal{V}(x) \subset \mathbb{R}^d$ with $W \in \mathcal{V}(0) \subset \mathbb{R}^d$ and $f : V \to W$ a $\mathcal{C}^k$-diffeomorphism such that $f(V \cap \mathcal{M}) = W \cap (\mathbb{R}^r \times \{0\})$.

Intuitively, a $\mathcal{C}^k$ r-submanifold is a topological space that is locally very similar to a neighbourhood of 0 in $\mathbb{R}^r$.

Example 2.4 : Classic examples of 2-manifolds of $\mathbb{R}^3$ are the 2 surfaces and the affine spaces.

Definition 2.18 : Let (X, d_X) be a metric space, let $f : X \to \mathbb{R}$ be a function. We say that f is Lipschitz continuous if there is a constant $L > 0$ such that for all $x, y \in X$, $|f(x) - f(y)| \leq L d_X(x, y)$.

2.1.1.3 Homotopy

Definition 2.19 : Let X, Y be two topological spaces and $f, g \in \mathcal{C}^0(X, Y)$. A homotopy between f and g is a continuous application $h : X \times [0, 1] \to Y$ such that $h(x, 0) = f(x)$ and $h(x, 1) = g(x)$ for every $x \in X$.

Basically, a homotopy between two functions is roughly a continuous transformation of one to the other.

Definition 2.20 : Let X, Y be two topological spaces, we say that they are homotopy equivalent (or homotopic) if there exist $f \in \mathcal{C}^0(X, Y)$ and $g \in \mathcal{C}^0(Y, X)$ such that $f \circ g$ and $g \circ f$ are homotopic to the identity applications id_X and id_Y respectively.

2.1.2 Probability Measures

Definition 2.21 : Let E be a set, a σ-algebra $\mathcal{A}$ is a non empty set of subset of E that is stable by complement, countable intersections and countable unions of its elements. We say that $(E, \mathcal{A})$ is a measurable space.

Example 2.5 : Looking at the definition of a topology, one σ-algebra on $\mathbb{R}$ is the collection of subsets that are formed from its open sets (open intervals, closed intervals, union of intervals and intersection of intervals), this σ-algrebra is called *Borel set*.

Definition 2.22 : Let $(E, \mathcal{A})$ and $(F, \mathcal{B})$ be two measurable spaces. We say that a function $f : E \to F$ is measurable if for all $B \in \mathcal{B}$, $f^{-1}(B) = \{a \in E \mid f(a) \in B\} \in \mathcal{A}$.

Definition 2.23 : Let $(E, \mathcal{A})$ be a measurable space. A (finite) measure on E is an application $\mu : \mathcal{A} \to \mathbb{R}_+$ verifying two conditions:

- $\mu(\emptyset) = 0$,
- for every sequence $(A_n)_{n\in\mathbb{N}} \in \mathcal{A}^{\mathbb{N}}$ of pairwise disjoint sets, $\mu\left(\bigcup_{n=0}^{+\infty} A_n\right) = \sum_{n=0}^{+\infty} \mu(A_n)$.

μ is called a probability measure if $\mu(E) = 1$ and $(E, \mathcal{A}, \mu)$ is a probability space (more often only denoted (E, μ)). We say that a subset A of E is negligible if $\mu(A) = 0$. We denote by $\mathcal{P}(E)$ the set of probability measures on $(E, \mathcal{A})$.

Example 2.6 : A simple example of a measure is the counting measure μ on a countable set such as $\mathbb{N}$. μ is such that for A a subset of $\mathbb{N}$, $\mu(A) = |A|$.
Another example of a measure is the so-called Lebesgue measure on $\mathbb{R}$ denoted λ: for $a < b$ we have $\lambda([a, b]) = \lambda(]a, b[) = \lambda([a, b[) = \lambda(]a, b]) = b - a$. λ is also generalized on $\mathbb{R}^d$ and is often denoted $\lambda^{\otimes d}$, for U a d-dimensional open set of $\mathbb{R}^d$, $\lambda^{\otimes d}(U) = \text{Vol}(U)$.

Example 2.7 : In $\mathbb{R}^d$, any k-submanifold with $k < d$ is negligible w.r.t the Lebesgue measure.

Definition 2.24 : Let (Ω, μ) be a probability space. A real valued random variable X is a measurable function $X : E \to \mathbb{R}$. The probability that X has a value that belongs to a set $C \in \mathbb{R}$ is $\mu_X(C) = \mu(X \in C) = \mu\big(\{\omega \in \Omega \mid X(\omega) \in C\}\big)$. The notation $X \sim \mu_X$ means that X is a random variable with distribution μ_X. In practice, when it is not ambiguous we use the notation $\mathbb{P}$.

Definition 2.25 : Let X be a real-valued random variable defined on $(\Omega, \mathbb{P})$ such that X is integrable, the expected value of X is defined as $\mathbb{E}[X] = \int_\Omega X(x) d\mathbb{P}(x)$.

Proposition 2.5 : Let $(\Omega, \mathbb{P})$ be a probability space and X a real-valued random variable. Let $f : \mathbb{R} \to \mathbb{R}$ a measurable function such that $f \circ X$ is integrable, then :

$$\mathbb{E}[f(X)] = \int_{\mathbb{R}} f(x) d\mathbb{P}_X(x).$$

Definitions 2.5 : Let $X : \mathbb{R} \to \mathbb{R}$ be a random variable. The cumulative distribution function of X is defined as $F_X(x) = \mathbb{P}(X \leq x)$ for all $x \in \mathbb{R}$. The quantile function of X is defined as the "general inverse" of the cumulative distribution function, that is

$$\begin{aligned} Q :& [0,1] \to \mathbb{R} \\ & p \mapsto \inf\big(\{x \in \mathbb{R} \mid p \leq F_X(x)\}\big). \end{aligned}$$

Definition 2.26 : Let X be a random variable on $\mathbb{R}^d$, we say that X admits a density function with respect to the Lebesgue measure λ if there is a positive function f such that $d\mathbb{P}_X = f d\lambda$, that is $\mathbb{P}_X(X \in A) = \int_{X^{-1}(A)} d\mathbb{P}_X(x) = \int_A f(x) d\lambda(x)$.

Proposition 2.6 (Rademacher) **:** A Lipschitz continuous function f on a d-submanifold of $\mathbb{R}^d$ is differentiable almost everywhere, i.e. it is differentiable everywhere except on subsets that are negligible w.r.t. the Lebesgue measure. In particular, the Lipschitz constant is $\sup \|\nabla f(x)\|$.

2.2 Optimal Transport

This part lays technical foundations of Optimal Transport (OT) theory to help for a better understanding of the several tools used and studied in this thesis. Optimal Transport theory formalizes the problem of displacing materials from several starting locations to several destinations. For a simple illustration, consider N storage buildings of rice at location $x_1, \ldots, x_N$ containing each a certain amount $a_1, \ldots, a_N$ of rice, and L restaurants at location $y_1, \ldots, y_L$ demanding certain amounts $b_1, \ldots, b_L$. Consider the cost of transportation between two locations x_k and y_l. Optimal transport theory formally finds the best way to dispatch the rice from locations x_k to destinations y_l with minimal cost. Consequently, OT has become a modern method for formally comparing probability distributions − whether discrete, continuous or represented as histograms − and, importantly, it defines metrics controlling the topology of the space of probability measures. This part is based on the following general references: *"Optimal Transport for Applied Mathematicians"* by F.Santambrogio [164], *"Computational Optimal Transport: With applications to data science"* by G.Peyré & M.Cuturi [142] and *"Optimal Transport: Old and New"* by C.Villani [192].

2.2.1 Monge's problem

The first optimal transport problem was formulated by Monge [121] which writes as follows:

Problem 2.1 (Monge problem) **:** Let $\mathcal{X}, \mathcal{Y}$ be two Polish spaces, $\mu \in \mathcal{P}(\mathcal{X})$ and $\nu \in \mathcal{P}(\mathcal{Y})$ two probability measures, and $c : X \times Y \to [0, +\infty]$ a cost function, solve:

$$\inf_{T:\mathcal{X}\to\mathcal{Y},\ T\#\mu=\nu} \int_X c\big(x, T(x)\big) d\mu(x), \tag{2.1}$$

where $T\#\mu : A \mapsto \mu\big(T^{-1}(A)\big)$ is called the push forward of μ by T.

However, Monge's formulation leads to some difficulties. Indeed, Eq. 2.1 is a non convex optimization problem and its constraint is too rigid, and may result in no feasible solution [142, 164]. To illustrate that, let us take a look into the discrete case where $\mu = \sum_{k=1}^{N} a_k \delta_{x_k}$ and $\mu = \sum_{l=1}^{L} b_l \delta_{y_l}$. Monge's problem rewrites as:

Problem 2.2 : Find a map $T : \{x_0, \dots, x_N\} \to \{y_1, \dots, y_L\}$ such that

$$\forall l,\ b_l = \sum_{k:T(x_k)=y_l} a_k.$$

When $L > N$, this problem is not feasible as T cannot transport a mass at location x_k to several locations y_l.

2.2.2 Kantorovich's relaxation

Because of those difficulties, a modern and more relaxed formulation introduced by Kantorovich [95] is considered:

Problem 2.3 (Kantorovich problem) **:** For $\mu \in \mathcal{P}(\mathcal{X})$, $\nu \in \mathcal{P}(\mathcal{Y})$ and $c : \mathcal{X} \times \mathcal{Y} \to [0, +\infty]$, solve

$$\inf_{\gamma\in\Pi(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{Y}} c(x, y) d\gamma(x, y), \tag{2.2}$$

where $\Pi(\mu, \nu) = \{\gamma \in \mathcal{P}(\mathcal{X} \times \mathcal{Y}) \mid \pi_{\mathcal{X}}\#\gamma = \mu\ ,\ \pi_{\mathcal{Y}}\#\gamma = \nu\}$ with $\pi_{\mathcal{X}} : (x, y) \mapsto x$ and $\pi_{\mathcal{Y}} : (x, y) \mapsto y$.

Let us interpret the difference with a practical example of the displacement of particles. Intuitively, Monge's formulation Eq. 2.1 only allows a one-to-one transportation of a particle at location x to y, whereas Kantorovich's one Eq. 2.2 permits the displacement of many particles located at x to several destinations y.
Formally speaking, $\gamma(x, y)$ is the amount of mass transported from x to y. This means that the Kantorovich's formulation is more permissive as it allows mass from x to be displaced to several locations. Unlike Eq. 2.1, this optimization problem always admits a feasible solution as $\Pi(\mu, \nu)$ always contains the product measure $\mu \otimes \nu : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_+, \quad X \times Y \mapsto \mu(X)\nu(Y)$. The minimizers of Eq. 2.2 are called *optimal transport plans*. In the discrete case with $\mu = \sum_{k=1}^{N} a_k \delta_{x_k}$ and $\mu = \sum_{l=1}^{L} b_l \delta_{y_l}$, a coupling γ writes as

$$\gamma = \sum_{k=1}^{N} \sum_{l=1}^{L} \gamma_{k,l} \delta_{x_k, y_l}, \tag{2.3}$$

such that $\sum_{k=1}^{N} \gamma_{k,l} = b_l$ and $\sum_{l=1}^{L} \gamma_{k,l} = a_k$. We denote by $\Gamma = (\gamma_{k,l}) \in \mathcal{M}_{m,n}(\mathbb{R})$ the coupling matrix corresponding to a coupling γ and we denote $\Pi(a,b)$ the set of coupling matrices between histograms $a = (a_k)_{k=1,\dots,N}$ and $b = (b_l)_{l=1,\dots,L}$. Denoting $C = (C_{k,l})$ a matrix of cost between the x_k and the y_l, Kantorovich's problem rewrites as

$$\inf_{\Gamma \in \Pi(a,b)} \sum_{k=1}^{N} \sum_{l=1}^{L} C_{k,l}\gamma_{k,l}. \tag{2.4}$$

2.2.2.1 Solving Kantorovich's problem in practice

Recall that Problem 2.3 is a convex constrained optimization problem, thus it admits a dual problem that is a concave constrained optimization problem.

Problem 2.4 (Dual Problem) **:** Let $\mu \in \mathcal{P}(\mathcal{X})$, $\nu \in \mathcal{P}(\mathcal{Y})$ and a cost function $c : \mathcal{X} \times \mathcal{Y} \to [0, +\infty[$, we solve:

$$\sup_{\{\varphi \in \mathcal{C}_b(\mathcal{X}),\ \xi \in \mathcal{C}_b(\mathcal{Y}) \ \mid\ [\varphi(x)+\xi(y)] \leq c\}} \int_{\mathcal{X}} \varphi d\mu + \int_{\mathcal{Y}} \xi d\nu. \tag{2.5}$$

First, one can notice that

$$\sup_{\{\varphi \in \mathcal{C}_b(\mathcal{X}),\ \xi \in \mathcal{C}_b(\mathcal{Y}) \ \mid\ [\varphi(x)+\xi(y)] \leq c\}} \int_{\mathcal{X}} \varphi d\mu + \int_{\mathcal{Y}} \xi d\nu \leq \inf_{\gamma \in \Pi(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{Y}} c(x,y) d\gamma.$$

In fact those two problems are equivalents:

Proposition 2.7 ([164]) **:**

$$\sup_{\{\varphi \in \mathcal{C}_b(\mathcal{X}),\ \xi \in \mathcal{C}_b(\mathcal{Y}) \ \mid\ [\varphi(x)+\xi(y)] \leq c\}} \int_{\mathcal{X}} \varphi d\mu + \int_{\mathcal{Y}} \xi d\nu = \inf_{\gamma \in \Pi(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{Y}} c(x,y) d\gamma$$

But one can even go further. Indeed one can see that given the constraint $\{\varphi \in \mathcal{C}_b(\mathcal{X}),\ \xi \in \mathcal{C}_b(\mathcal{Y}) \mid \varphi(x) + \xi(y)] \leq c\}$, if we take a pair φ, ξ, with φ fixed, then taking ξ as the c-transform of φ improves the solution.

Definition 2.27 : For $\varphi \in \mathcal{C}_b(\mathcal{X})$, we call the c-transform of φ the function

$$\begin{aligned} \varphi^c(y) :& \mathcal{Y} \to \mathbb{R} \\ & y \mapsto \inf_{x \in \mathcal{X}} c(x,y) - \varphi(x). \end{aligned}$$

The same way, for a given ξ, the best φ is the $\bar{c}$-transform of ξ.

Definition 2.28 : For $\xi \in \mathcal{C}_b(\mathcal{Y})$, we call the $\bar{c}$-transform of ξ the function

$$\begin{aligned} \xi^{\bar{c}}(x) :& \mathcal{X} \to \mathbb{R} \\ & x \mapsto \inf_{y \in \mathcal{Y}} c(x,y) - \xi(y). \end{aligned}$$

We call φ a c-concave function on $\mathcal{X}$ if there is $\xi : \mathcal{Y} \to \mathbb{R}$ such that $\varphi = \xi^{\overline{c}}$, similarly we call ξ a $\overline{c}$-convave function if there is $\varphi : \mathcal{X} \to \mathbb{R}$ such that $\xi = \varphi^c$. We denote $c - concav(\mathcal{X})$ the space of c-concave function on $\mathcal{X}$ (same for $c - concav(\mathcal{Y})$). Consequently, one can replace a pair (φ, ξ) by (φ, φ^c), giving the following result.

Proposition 2.8 : If $\mathcal{X}$ and $\mathcal{Y}$ are compact and c is continuous. Then Problem 2.4 admits a solution $(\varphi, \psi) = (\varphi, \varphi^c)$ with φ $c - concav$. In other words

$$\sup_{\{\varphi \in \mathcal{C}_b(\mathcal{X}),\ \xi \in \mathcal{C}_b(\mathcal{Y}) \ \mid\ [\varphi(x)+\xi(y)] \leq c\}} \int_{\mathcal{X}} \varphi d\mu + \int_{\mathcal{Y}} \xi d\nu = \sup_{\varphi \in c-concav(\mathcal{X})} \int_{\mathcal{X}} \varphi d\mu + \int_{\mathcal{Y}} \varphi^c d\nu.$$

If we look into the case where $\mu = \sum_{k=1}^{N} a_k \delta_{x_k}$ and $\nu = \sum_{l=1}^{L} b_l \delta_{y_l}$ are both discrete, the dual problem simply writes as:

Problem 2.5 : Solve

$$\max_{\{\varphi \in \mathbb{R}^N,\ \xi \in \mathbb{R}^L \ \mid\ \forall (k,l), \varphi_k + \xi_l \leq C_{k,l}\}} \langle \varphi, a \rangle + \langle \xi, b \rangle.$$

We can also define a c-transform of a vector $\varphi \in \mathbb{R}^N$ as:

$$(\varphi^c)_l = \min_{k \in \{1,\dots,N\}} C_{k,l} - \varphi_k,$$

and consequently consider the following optimization problem.

Problem 2.6 : Solve

$$\max_{\varphi \in \mathbb{R}^N} \langle \varphi, a \rangle + \langle \varphi^c, b \rangle.$$

2.2.3 Wasserstein distance

The Wasserstein distance is a specific case of Kantorovich's formulation where $\mathcal{X} = \mathcal{Y}$ and $c(x, y) = d(x, y)^q$ where d is a distance.

Definition 2.29 : Let $(\mathcal{X}, d)$ be a Polish space, let $\mu, \nu \in \mathcal{P}(\mathcal{X})$, $c : (x, y) \mapsto d(x, y)^q$ with $q \in \mathbb{R}$, the q-Wasserstein distance writes:

$$W_q(\mu, \nu) \coloneqq \left(\inf_{\gamma \in \Pi(\mu,\nu)} \int_{(\mathbb{R}^d)^2} d(x, y)^q d\gamma(x, y) \right)^{1/q}. \tag{2.6}$$

By Villani [192], W_q defines a distance between probability measures of finite q-th moment on $\mathcal{X}$.
In the following and for the rest of the thesis, unless specified, we will assume $\mathcal{X} = \mathbb{R}^d$ for $d \geq 1$.

Definition 2.30 : Let $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$, $c : (x, y) \mapsto \|x - y\|^q$ with $q \in \mathbb{R}$ and $\| \cdot - \cdot \|$ the Euclidean distance, the q-Wasserstein distance writes:

$$W_q(\mu, \nu) \coloneqq \left(\inf_{\gamma \in \Pi(\mu,\nu)} \int_{(\mathbb{R}^d)^2} \|x - y\|^q d\gamma(x, y) \right)^{1/q}. \tag{2.7}$$

2.2.3.1 Optimal transport on the real line

In the specific case where μ and ν are two probability measures defined on $\mathbb{R}$ (or a space isomorphic to $\mathbb{R}$), then the q-Wasserstein distance is well known and has an explicit formula.

Proposition 2.9 : For $\mu \in \mathcal{P}(\mathbb{R})$ and $\nu \in \mathcal{P}(\mathbb{R})$, denoting F_μ, F_ν their cumulative distribution function and F_μ^{-1},F_ν^{-1} their generalized inverse (or quantiles function), the q-Wasserstein distance writes

$$W_q(\mu,\nu) = \left(\int_{[0,1]} |F_\mu^{-1}(t) - F_\nu^{-1}(t)|^q dt \right)^{1/q}. \tag{2.8}$$

This means that the Wasserstein distance between one dimensional probability measures is obtained by comparing their quantiles in an increasing order.

2.2.3.2 Balanced discrete case

This part provides some insight into the Wasserstein distance when μ and ν are both discrete. In the following and in this thesis, unless specified, μ and ν have the same number of points support with uniform weight. We call this setting a *balanced optimal transport problem.*

Definition 2.31 : We denote $\mu = \frac{1}{K}\sum_{k=1}^{K} \delta_{x_k}$, $\nu = \frac{1}{K}\sum_{k=1}^{K} \delta_{y_k}$ with $x = \{x_1, \dots, x_K\}$, $y = \{y_1, \dots, y_N\}$ two sets of $N \geq 1$ points in $\mathbb{R}^d$. We also denote i_x and i_y the sets of indices of x and y respectively. The q-Wasserstein distance writes:

$$W_q(\mu,\nu) = \left(\inf_{\psi : i_x \xrightarrow{bij} i_y} \frac{1}{K} \sum_{k=1}^{K} \|x_k - y_{\psi(k)}\|^q \right)^{1/q}. \tag{2.9}$$

Eq. 2.9 is thus a displacement problem under the constraint of having bijections as optimal transport plans.

Now recalling that the p-Wasserstein distance admits an explicit formula when dealing with probability measures on $\mathbb{R}$, this formula (Eq. 2.8) becomes

$$W_q(\mu,\nu) = \left(\frac{1}{K} \sum_{k=1}^{K} \|x_{\sigma(k)} - y_{\tau(k)}\|^q \right)^{1/q},$$

where σ and τ are permutations of $\{1, \dots, N\}$ respectively ordering the sets $\{x_1, \dots, x_K\}$ and $\{y_1, \dots, y_K\}$ on $\mathbb{R}$.

Remark 2.3 : Note that in this last case, the Monge's problem and its Kantorovich's relaxation are equivalent.

2.2.3.3 Probabilistic interpretation

Eq. 2.6 can also be interpreted in a probabilistic way, and is equivalent to the following formulation:

$$W_q(\mu,\nu)^q = \min_{X\sim\mu, Y\sim\nu} \mathbb{E}[d(X,Y)^q]. \tag{2.10}$$

2.2.4 Wasserstein barycenter

This section details a nonlinear interpolation between probability measures on $\mathbb{R}^d$. The same way one obtains a barycenter between points $\{x_1, \dots, x_m\}$ in $\mathbb{R}^d$ with barycentric weights $(\lambda_1, \dots, \lambda_m)$ as

$$\arg\min_{x \in \mathbb{R}^d} \sum_{i=1}^{m} \lambda_i \|x - x_i\|^2,$$

one can obtain such barycenter by replacing the Euclidean distance by the 2-Wasserstein distance and the x_i by probability measures μ_i.

Definition 2.32 (Agueh & Carlier [5]) **:** Let $\mu_1, \dots, \mu_m$ be m probability measures on $\mathbb{R}^d$ with finite second moments, let $\lambda_1, \dots, \lambda_m$ be positive coefficients summing to 1. A Wasserstein barycenter μ^* is then defined as

$$\mu^* \in \arg\min_{\mu} \sum_{i=1}^{m} \lambda_i W_2^2(\mu, \mu_i). \tag{2.11}$$

2.3 Topological Data Analysis

This part presents the backbone of a modern method for comparing topological manifolds. The first works on comparison of topological spaces showed that connectivity arguments can be used. Indeed the image of a connected space by a continuous function is a connected space, thus two spaces that have different numbers of connected component cannot be similar (they are not homeomorphics). Further works then took their interests on the boundary of a manifold, the following observation being the foundation of homology theory: the boundary of a topological manifold is a manifold with an empty boundary. This simple observation was followed by the algebraic formalization of a boundary, and consequently what has an empty boundary which is commonly called a "hole". This section is based on the following general references: *"Elements of Algebraic Topology"* by J.Munkres [124], *"Algebraic Topology"* by A.Hatcher [86], *"Topologie Algébrique: Une Introduction et Au Delà* by C.Lerustre [111] , *"Computational Topology: An Introduction"* by H.Edelsbrunner & J.Harer [58], and *"Fréchet Means for Distributions of Persistence Diagrams"* by K.Turner & .al [187].

2.3.1 Simplicial Homology

The theory of homology groups formalizes those notion of "holes" in an algebraic way, and is commonly used to compare topological manifolds. In this thesis we detail one way to define homology based on simplicial complexes (or triangulations). It is called simplicial homology and it is the most concrete among the homology theories.

2.3.1.1 Simplicial complex

Definition 2.33 (Simplex) **:** For $P = \{p_0, p_1, \dots, p_k\} \subset \mathbb{R}^d$ a set of $(k+1)$ points affinely independent, i.e. they do not belong on a common affine line, the k-simplex σ with

vertices P is

$$\sigma = \left\{ \sum_{i=0}^{k} \lambda_i p_i \mid \forall i,\ \lambda_i \geq 0,\ \sum_{i=0}^{k} \lambda_i = 1 \right\}. \tag{2.12}$$

In this setting we denote $\sigma = [p_0, p_1, \ldots, p_k]$ the simplex spanned by P, and k its dimension.

Remark 2.4 :
- A 0-simplex is a point, a 1-simplex is a segment, a 2-simplex is a filled triangle, and a 3-simplex is a filled tetrahedron.
- The faces of σ are the simplices spanned by the subsets of P.

Definition 2.34 (Simplicial complexes) **:** A simplicial complex $\mathcal{K}$ in $\mathbb{R}^d$ is a collection of simplices s.t.:

- a face of a simplex of $\mathcal{K}$ is a simplex of $\mathcal{K}$.
- the intersection of two simplexes of $\mathcal{K}$ is either empty or a common face.

We also introduce two definitions that will be useful in the following.

Definition 2.35 : We denote by $|\mathcal{K}|$ the *underlying space*, which is the union of all the simplices of $\mathcal{K}$ with the topology inherited by $\mathbb{R}^d$.

Fig. 2.1 illustrates a simple example of a simplicial complex, and a collection of simplices that is not a simplicial complex.

Definition 2.36 : Let M be a subcollection of $\mathcal{K}$, if M contains all simplices of $\mathcal{K}$ spanned by its element then it is a subcomplex of $\mathcal{K}$. A particular subcomplex is the k-skeleton, containing all simplex of $\mathcal{K}$ of at most dimension k.

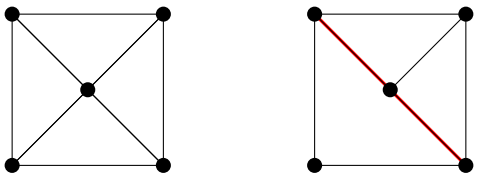


Figure 2.1: On the left we have a simplicial complex, on the right we do not have a simplicial complex as there is an intersection of simplices that is the union of two simplices (colored in red), thus is not a face in this example.

2.3.1.2 Homology

Definition 2.37 (**Chain complexes**) **:** Let $\mathcal{K}$ be a simplicial complex and k an integer. A k-chain is a finite formal sum of L k-simplices in $\mathcal{K}$. The standard notation for this is $c = \sum_{i=1}^{L} a_i \sigma_i$ where σ_i are k-simplices of $\mathcal{K}$ and a_i coefficients in $\mathbb{Z}/2\mathbb{Z}$.

Remark 2.5 : In topological data analysis, we mostly study the setting where the coefficients a_i belong to the group $\mathbb{Z}/2\mathbb{Z}$. But chain complexes can also be defined using coefficients belonging to other groups, rings or fields (in general $\mathbb{Z}$).

Definition 2.38 : The sum of two k-chains $c = \sum_i a_i\sigma_i$ and $c' = \sum_i b_i\sigma_i$ is defined as follows:

$$c + c' = \sum_i (a_i + b_i)\sigma_i, \text{ with } 1 + 1 = 0 \text{ in } \mathbb{Z}/2\mathbb{Z}.$$

This formal sum defines an inner law between k-chains, thus it defines a group structure (associativity, identity element, inverse element).

Definition 2.39 : For $k \in \mathbb{N}$, the k-chains space over a simplicial complex $\mathcal{K}$ is the abelian group spanned by the formal sums of k-simplices. Notation $C_k = C_k(\mathcal{K})$ denotes such group.

For $k < 0$ and $k > \dim(\mathcal{K})$, this group is trivial $C_k = \{0\}$. To relate these groups we define the boundary of a k-simplex as the sum of its $(k-1)$ dimensional faces.

Definition 2.40 : Writing $\sigma = [v_0, v_1, \ldots, v_k]$ for the simplex spanned by the listed vertices, its boundary is:

$$\partial_k\sigma = \sum_{i=0}^{k} [v_0, \ldots, \hat{v}_i, \ldots, v_k],$$

where $[v_0, \ldots, \hat{v}_i, \ldots, v_k]$ is the i-th face of σ by omitting v_i.

In Fig. 2.2, we have a simple visualization of a boundary of a 2-simplex.

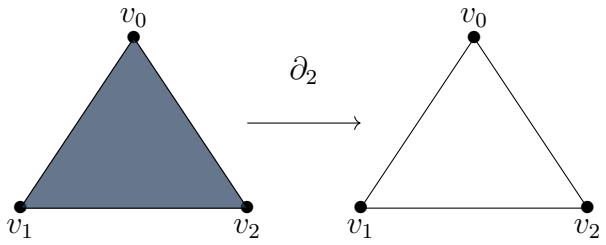


Figure 2.2: The 2-simplex, which is a filled triangle, can be written as $\sigma = [v_0, v_1, v_2]$. Its boundary is $\partial_2\sigma = [v_0, v_1] + [v_0, v_2] + [v_1, v_2]$ which is an empty triangle spanned by the sum of the three 1-simplices.

This definition of a simplex boundary leads us to define ∂ as a linear operator:

Definition 2.41 : the boundary operator or the boundary homomorphism is defined as

$$\begin{aligned} \partial_k : C_k &\longrightarrow C_{k-1} \\ c &\longmapsto \partial_k c = \sum_{\sigma \in c} \partial_k \sigma. \end{aligned}$$

The boundary homomorphism links the k-chains groups the following way:

$$\ldots \xrightarrow{\partial_{k+1}} C_k(\mathcal{K}) \xrightarrow{\partial_k} C_{k-1}(\mathcal{K}) \xrightarrow{\partial_{k-1}} \ldots \xrightarrow{\partial_2} C_1(\mathcal{K}) \xrightarrow{\partial_1} C_0(\mathcal{K}) \xrightarrow{\partial} \{0\}.$$

This sequence leads us to the definition of the kernel and image of the boundary operator.

Definition 2.42 (**k-cycles and k-boundaries**) **:** A k-cycle is a k-chain with a null boundary, i.e. the set of k-cycles of $\mathcal{K}$ is defined as:

$$Z_k(\mathcal{K}) = Z_k = \mathrm{Ker}(\partial_k) = \{c \in C_k \mid \partial_k c = 0\}.$$

We define a k-boundary as a k-chain that is the boundary of a $(k+1)$-chain, i.e. the set of k-boundaries of $\mathcal{K}$ is defined as:

$$B_k(\mathcal{K}) = B_k = \mathrm{Im}(\partial_{k+1}) = \{c \in C_k \mid \exists c' \in C_{k+1},\ \partial_{k+1} c' = c\}.$$

Those two sets are groups as they are respectively the kernel and the image of a homomorphism.

Fig. 2.3 shows a case where a chain is both a cycle and a boundary.
Let us now introduce the fundamental lemma carried by the famous quote:

"The boundary of a boundary is empty."

Lemma 2.1 (**Fundamental lemma of Homology**) **:**

$$\partial_{k-1}\partial_k \sigma = 0,\ \forall k \in \mathbb{N} \setminus \{0\},\ \forall\ \sigma \in C_k.$$

Proof. As $\partial_{k-1}\partial_k$ is linear, we just need to show that $\partial_{k-1}\partial_k\sigma = 0$ for a k-simplex σ. Let $\sigma = [v_0, \ldots, v_k]$ be a k-simplex, we have:

$$\partial_{k-1}(\partial_k \tau) = \partial_{k-1} \sum_{i=0}^{k} [v_0, \ldots, \hat{v}_i, \ldots, v_k] = \sum_{i=0}^{k} \partial_{k-1}[v_0, \ldots, \hat{v}_i, \ldots, v_k]$$

$$= \sum_{i=0}^{k} \sum_{i \neq j} [v_0, \ldots, \hat{v}_j, \hat{v}_i, \ldots, v_k] = \sum_{j<i} [v_0, \ldots, \hat{v}_j, \hat{v}_i, \ldots, v_k] + \sum_{j>i} [v_0, \ldots, \hat{v}_j, \hat{v}_i, \ldots, v_k].$$

As the last two sums are the same with different permutations, the sum of the two is equal to 0. □

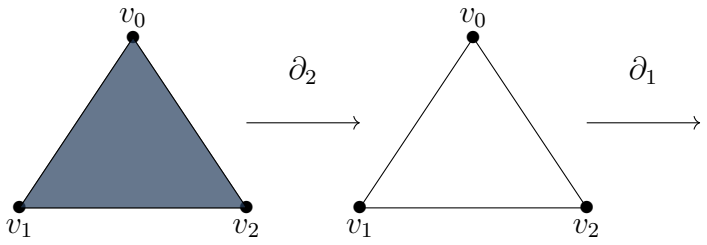


Figure 2.3: Initially we have $\sigma = [v_0, v_1, v_2]$. Then we apply the boundary operator $\partial_2\sigma = [v_0, v_1] + [v_0, v_2] + [v_1, v_2]$, and $\partial_1\partial_2\sigma = [v_0] + [v_1] + [v_0] + [v_2] + [v_1] + [v_2] = 0$.

This lemma gives us this direct corollary:

Corollary 2.1 : B_k is a normal subgroup of Z_k.

B_k being a normal subgroup of Z_k ensures that taking the quotient of Z_k by B_k gives a group, and this group is the so-called *homology group*.

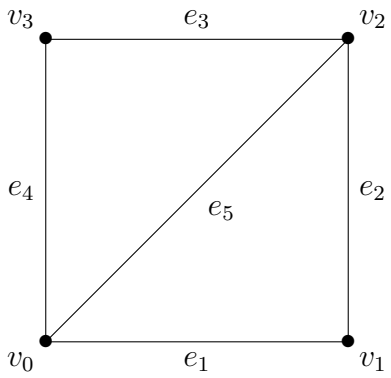
Definition 2.43 : The k-th *homology group* is the k-th cycle group modulo the k-th boundary group, or the group of cosets of the k-th boundary group:

$$H_k(\mathcal{K}) = H_k = Z_k/B_k.$$

The k-th *Betti number* [146] is defined as the rank of the k-th homology group: $\beta_k = \text{rk}(H_k)$.

A cycle σ defines an equivalence class $[\sigma]$ in H_k, and its class is called *homology class*. Looking at the definition, a cycle σ is a boundary if and only if $[\sigma] = 0$ (Rk. 2.2). Before giving an interpretation of homology groups and the Betti numbers, let us compute the 1-st homology groups in two examples.

Example 2.8 : We consider the following complex $\mathcal{K}$ comprised of the elementary 1-chains e_1, e_2, e_3, e_4, e_5. Let us take the following chain $\sigma = \sum_{i=1}^{5} a_i e_i$.

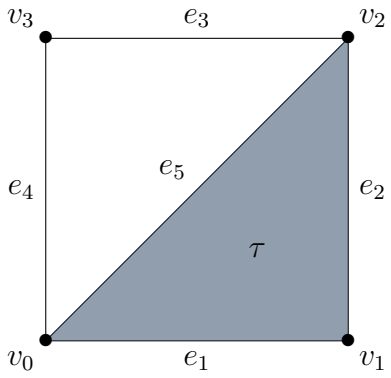


We have $\partial_1\sigma = 0 \iff \partial_1 \sum_{i=1}^{4} a_i e_i = 0 \iff \begin{cases} a_1 + a_2 = 0 \\ a_3 + a_4 = 0 \\ a_1 + a_4 + a_5 = 0 \\ a_2 + a_3 + a_5 = 0 \end{cases}$. This system has 2 degrees of freedom, choosing values for a_1 and a_3 induces the values of a_2, a_4, a_5. We keep in mind that the a_i take their values in $\mathbb{Z}/2\mathbb{Z}$. There are tree cases:

- $a_1 = 1$ and $a_3 = 0$: we have $a_2 = 1$, $a_4 = 0$ and $a_5 = 1$, giving us the chain $e_1 + e_2 + e_5$.
- $a_1 = 0$ and $a_3 = 1$: we have $a_2 = 0$, $a_4 = 1$ and $a_5 = 1$, giving us the chain $e_3 + e_4 + e_5$.
- $a_1 = 1$ and $a_3 = 1$: we have $a_2 = 1$, $a_4 = 1$ and $a_5 = 0$, giving us the chain $e_1 + e_2 + e_3 + e_4$.

We also have $e_1 + e_2 + e_3 + e_4 = e_1 + e_2 + e_5 + e_3 + e_4 + e_5$. This means that Z_1 is generated by the first two chains, and is of rank 2. $B_1 = \{0\}$ as there are no 2-simplices in the complex, meaning that $H_1 = Z_1/B_1 = Z_1/\{0\} \simeq Z_1$. Thus $\beta_1 = 2$.

Example 2.9 : Now we consider the following chains $\sigma = \sum_{i=1}^{5} a_i e_i$ and $\tau = [v_0, v_1, v_2]$.



The same way as in the previous example, Z_1 is generated by the chains $e_1 + e_2 + e_5$ and $e_3 + e_4 + e_5$. However $e_1 + e_2 + e_5 = \partial_2 \tau \in B_1$, meaning that its homology class is $[0]$, thus $H_1 = \langle [e_3 + e_4 + e_5] \rangle$. So we have $\beta_1 = \text{rk}(H_1) = 1$.

Notice that we let H_0 on the side. This homology group is easy to compute thanks to this following result [124]:

Theorem 2.2 : Let $\mathcal{K}$ be a complex, $H_0(\mathcal{K})$ is generated by taking one vertex from each connected component of $\mathcal{K}$.

This means that one only has to look at the connected components to find the 0-th homology classes and β_0. Following this theorem, we have the following property:

Proposition 2.10 : Let $\mathcal{K} = \{x\}$ with x in $\mathbb{R}^d$, then $H_k(\mathcal{K}) = 0$ for all $k \geq 1$ and $H_0(\mathcal{K}) = 1$.

For terminology, we say that two k-chains c and c' are *homologous* if there is a boundary $b \in B_1$ such that $c' = c + b \iff c' \in c + B_1$. Recall the complex in Ex. 2.9, we take $c = e_3 + e_4 + e_5$ (the generator of H_1) and $c' = e_1 + e_2 + e_3 + e_4$. Those two chains are homologous as $c' = e_1 + e_2 + e_3 + e_4 + e_5 + e_5 = e_3 + e_4 + e_5 + e_1 + e_2 + e_5 = c + \partial_2 \tau$. In other words, to obtain a k-homology class $[c]$, one has to add all k-boundaries to its representative, $c + B_k$ with c a k-cycle.

Informally speaking, the homology groups represents the k-dimension "holes" in a simplicial complex, more formally called *"handles"*. Thus they are topological descriptors of a simplicial complex. As such, they are invariant by homotopy and homeomorphism.

Proposition 2.11 : Homology groups are invariant by homotopy and homeomorphism.

Until now we have detailed the theory behind simplicial homology without taking into account the case for common topological spaces. We introduce a definition linking the two cases.

Definition 2.44 : Let $\mathcal{M}$ be a r-submanifold of $\mathbb{R}^d$, a triangulation of $\mathcal{M}$ is a simplicial complex $\mathcal{K}$ such that its underlying space $|\mathcal{K}|$ is homeomorphic to $\mathcal{M}$. We say that $\mathcal{M}$ is triangulable it there exists such a triangulation.

This means that to find the homology groups of a topological space, one can triangulate such a space with a simplicial complex (if it is triangulable) and compute the homology groups on it.

Remark 2.6 : We have not detailed the theory of homology groups for general topological spaces as it would be too much of a detour for this thesis. However the intuition and definition are the same (the way leading to it is not...), the homology groups represent the "handles" of a topological space.

2.3.2 Persistent homology

The persistent homology theory is a natural extension of the homology theory. Its goal is the same, comparing topological spaces in an efficient and quick way. It goes further than the homology theory. Indeed while homology theory permits one to know about the homology groups of a topological manifold, i.e. its topological invariants, persistent homology allows one to "track" the evolution homology groups, during a sweep, of a manifold based on scalar values or data values.

Intuition: Let $\mathcal{M}$ be a surface "oriented vertically". Take a scalar function $f : \mathcal{M} \to \mathbb{R}$ representing a height function. Now take a value $a \in \mathbb{R}$, one can track the evolution of the topology of $\mathcal{M}$ when varying a by studying the preimage of $]-\infty, a]$ by f. Thus one can track the evolution of the homology groups of $f^{-1}(]-\infty, a]$ while sweeping a.

2.3.2.1 Filtration of a simplicial complex

Definition 2.45 : Let $\mathcal{M}$ be a r-submanifold ($\mathcal{C}^p$) of $\mathbb{R}^d$. Let $f : \mathcal{M} \to \mathbb{R}$ be a scalar function. For $a \in \mathbb{R}$, we call its preimage $f^{-1}(a) = \{x \in \mathcal{M} \ f(x) = a\}$ a level set. We also call $\mathcal{M}_a = f^{-1}(]-\infty, a]) = \{x \in \mathcal{M} \mid f(x) \leq a\}$ a sublevel set.

In some cases it is required for the function to be $\mathcal{C}^\infty$, that its critical points are non-degenerate (its Hessian admits an inverse) and that all critical points have different critical values. The second condition prevents any plateau around the critical points and in most cases is the only condition that is not discarded.

However, for obvious reasons, such a function cannot be defined in practice on a submanifold $\mathcal{M}$. So we go around this diffculty by defining a similar function on a simplicial complex approximating $\mathcal{M}$.

Definition 2.46 : Let $\mathcal{K}$ be a triangulation of $\mathcal{M}$, with vertices in $\mathcal{M}$ that have scalar values specified. A piecewise linear (PL) function $f : |\mathcal{K}| \to \mathbb{R}$ is a function such that $f(x) = \sum_{i=1}^{k} \lambda_i(x) f(u_i)$ for $x \in \sigma = [u_1, \ldots, u_k]$ a simplex, $\lambda_i(x)$ the barycentric coordinates as in Eq. 2.12 and $f(u_i)$ the values already assigned beforehand.

Now that we have a proper function on a simplicial complex, the goal is to build nested subcomplexes $K_0 \subset K_1 \subset \ldots \subset K_l = K$ so that the homology groups of each K_i can be computed, allowing their evolution to be tracked. This type of nested spaces, especially topological spaces, is called a filtration:

Definition 2.47 : A filtration is an indexed set $(X_i)_{i \in I}$ of subspaces of a topological space X such that $X_i \subset X_j$ if $i \leq j$.

Before defining a filtration of a simplicial complex, we need two simple notions first.

Definition 2.48 : Let v be a vertex of a simplicial complex $\mathcal{K}$. The star of v is the union of the interiors of the simplices of K that have v as a vertex. It is denoted $St(v)$. Its closure is denoted $\overline{St(v)}$ and is called the closed star of v (Fig. 2.4).
We define the lower star, noted $LSt(v)$, as the set of simplices in $St(v)$ such that $f(v)$ is the maximum:

$$LSt(v) = \{\sigma \in St(v) \mid u \in \sigma \implies f(u) \leq f(v)\}.$$

Now denote all the vertices of a simplicial complex $\mathcal{K}$ $(u_i)_{i\in\{1,\dots,l\}}$ and order them such that $f(u_i) < f(u_j)$ for $i \leq j$, taking $K_i = \bigcup_{j\leq i} LSt(u_j)$ defines a filtration of $\mathcal{K}$ and $\bigcup_{i=1}^{l} K_i = \mathcal{K}$. This filtration is called the *lower star filtration* of $\mathcal{K}$. Moreover, taking a such that $f(u_i) \leq a < f(u_{i+1})$, one can see that $|\mathcal{K}|_a = f^{-1}(]-\infty, a])$ is homotopy equivalent to $|K_i|$, the retractions $t \mapsto (1-t)y + tu_i$ and $t \mapsto (1-t)u_i + ty$, with $f(y) = a$, provides the result. This ensures a continuous tracking of the homology groups by the sublevel sets.

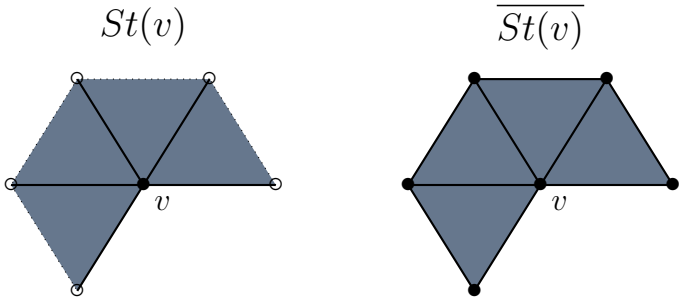


Figure 2.4: On the left we have the star of a vertex v in a simplicial complex. The dashed edges are the simplices that are not in $St(v)$. The colored triangles represent the interiors of the 2-simplices that have v as a vertex. On the right we have the closed star of v.

The inclusions between the $|K_i|$ induce the following natural inclusion applications:

$$\emptyset = |K_0| \hookrightarrow |K_1| \hookrightarrow \dots \hookrightarrow |K_l| = |\mathcal{K}|.$$

Those inclusion applications induce themselves natural homomorphisms between the $H_k(K_i)$.

2.3.2.2 Evolution of homology

Proposition 2.12 : Let $(K_i)_{i\in\{1,\dots,l\}}$ be a lower star filtration of a simplicial complex $\mathcal{K}$ associated to a PL function f. The inclusion applications between the $|K_i|$ induce the following homomorphisms denoted $f_k^{i,j} : H_k(K_i) \to H_k(K_j), \quad [\sigma]_i \mapsto [\sigma]_j$ for $i \leq j$.

$[\sigma]_i$ denotes the homology class in K_i, and $[\sigma]_i$ can be non-trivial with $[\sigma]_j$ being null. Thanks to those morphisms, we have the following sequences:

$$\{0\} = H_k(K_0) \xrightarrow{f_k^{0,1}} H_k(K_1) \xrightarrow{f_k^{1,2}} \dots \xrightarrow{f_k^{l-1,l}} H_k(K_l) = H_k(\mathcal{K}),$$

for all dimensions k.

Those sequences allow the tracking of the homology classes. This is the foundation of persistent homology, tracking when a homology class "appears" in the sequence and when it "disappears" in the sequence.

Definition 2.49 (Edelsbrunner & al [58]) **:** The k-th persistent homology groups are defined as the image of the induced homomorphisms:

$$H_k^{i,j} = \mathrm{Im}(f_k^{i,j}),\ \forall\ 0 \leq i \leq j \leq l.$$

As for the Betti number, the k-th persistent Betti numbers are defined as the ranks of these groups $\beta_k^{i,j} = \mathrm{rk}(H_k^{i,j})$.

For terminology, we say that a homology class $[\sigma]_i$ in $H_k(K_i)$ is *born* at K_i if $[\sigma]_i$ is not an image by $f_k^{i-1,i} : H_k(K_{i-1}) \to H_k(K_i)$, i.e. $[\sigma]_i \notin H_k^{i-1,i}$. Informally, it means that $[\sigma]_i$ does not come from a previous subcomplex prior to K_i (included). With the same reasoning, we say that $[\sigma]_i$ *dies* at K_j if it has the same image with an older class coming from K_{i-1} when going from K_{j-1} to K_j. Formally, $[\sigma]_i$ merges with another class at K_j if:

- $f_k^{i,j-1}([\sigma]_i)$ is not an image by $f_k^{i-1,j-1}$, meaning that it does not come from K_{i-1} onward to K_{j-1}, i.e. $f_k^{i,j-1}([\sigma]_i) \notin H_k^{i-1,j-1}$.
- There is $[\gamma]_j$ that is an image by $f_k^{i-1,j}$ such that $f_k^{i,j}([\sigma]_i) = [\gamma]_j = f_k^{i-1,j}([\gamma]_{i-1})$, i.e. $f_k^{i,j}([\sigma]_i) \in H_k^{i-1,j}$.

This process is called the *Elder Rule*, and Fig. 2.5 illustrates the process of a merging of two classes during a filtration.

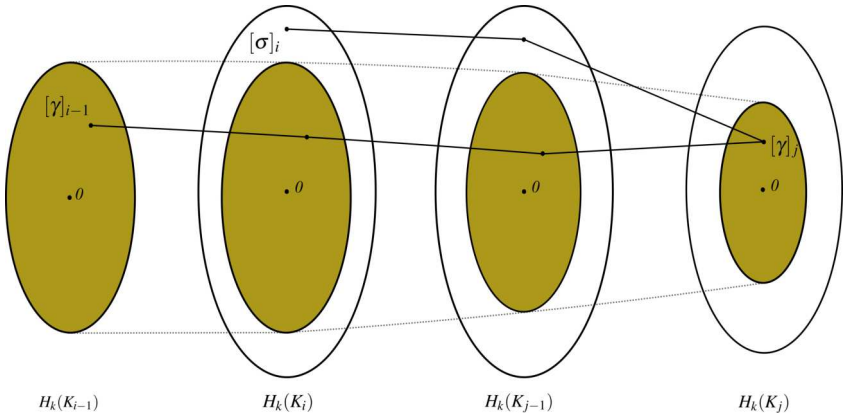


Figure 2.5: $[\gamma]_{i-1}$ was born at K_{i-1} and "lives" through the images of $f_k^{i-1,i}$, $f_k^{i-1,j-1}$ and $f_k^{j-1,j}$ (shaded areas). Another class $[\sigma]_i$ is born at K_i then lives through the images $f_k^{i-1,j-1}$ and $f_k^{j-1,j}$, until $f_k^{i,j}([\sigma]_i) = [\gamma]_j = f_k^{i-1,j}([\gamma]_{i-1})$.

2.3.2.3 Critical points

Recall that for $a \in \mathbb{R}$ such that $f(u_i) \leq a < f(u_{i+1})$, $|K_a|$ has the same homotopy as $|K_i|$, and in particular the homology remains the same by Prop. 2.11. This means that the only moment we can possibly observe the *birth* or the *death* of a homology group is at a u_i. Before characterizing such vertices, we need two simple notions.

Definition 2.50 : Let v be a vertex of a simplicial complex $\mathcal{K}$. Recalling the star of v, $St(v)$, and its closure $\overline{St(v)}$, we define the link of v (Fig. 2.6) as:

$$Lk(v) = \overline{St(v)} - St(v).$$

Similarly to the lower star, the lower and upper link of v is defined as:

$$LLk(v) = \{\sigma \in Lk(v) \mid u \in \sigma \implies f(u) < f(v)\},$$
$$ULk(v) = \{\sigma \in Lk(v) \mid u \in \sigma \implies f(u) > f(v)\}.$$

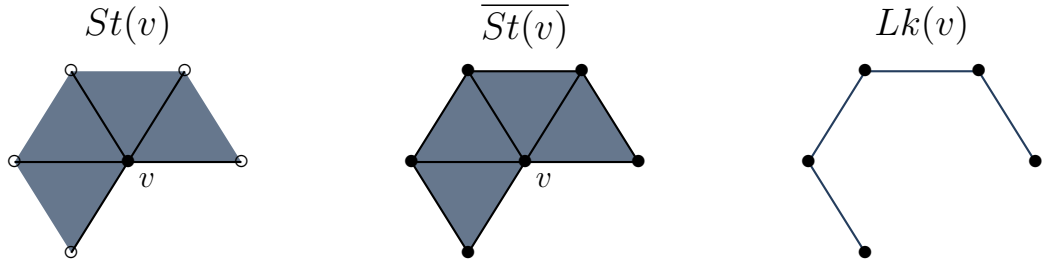


Figure 2.6: On the left we have the star of a vertex v, in the middle its closed star and on the right its link.

We can now classify the vertices of a triangulation $\mathcal{K}$ of a r-submanifold using the reduced homology of their lower links, see Fig. 2.7 for an illustration.

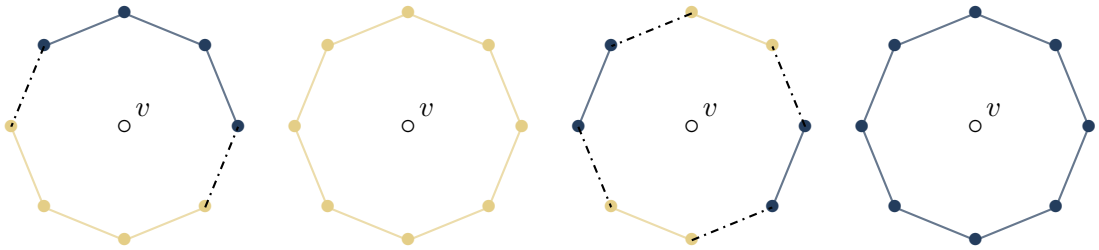
Definition 2.51 : Let $\mathcal{K}$ be a triangulation of a r-submanifold $\mathcal{M}$ with a PL function f on $\mathcal{K}$. Let v be a vertex of $\mathcal{K}$, it is said that:

- v is a *PL regular vertex* if $LLk(v) \neq \emptyset$ and $ULk(v) \neq \emptyset$ with $H_k\big(ULk(v)\big) = 1$ and $H_k\big(ULk(v)\big) = 0$ for $1 \leq k \leq r$.
- Otherwise, v is called a *PL critical vertex of index* q for $0 \leq q \leq r$:
  - If $LLk(v)$ is empty, then v is called a minimum and is a critical vertex of index 0. By convention we identify $\mathbb{S}^{-1} = \emptyset$.
  - If $ULk(v)$ is empty, then v is called a maximum and is a critical vertex of index r.
  - Otherwise, it is called a saddle and is a critical vertex of index $1 \leq q \leq r-1$.

Now that we have a classification of the vertices of a triangulation. We can now say when the homotopy of the lower star filtration changes.

Proposition 2.13 : Let $\mathcal{K}$ be a triangulation of a r-submanifold with simplices $(u_i)_{i\in\{1,\dots,l\}}$. Let $f : \mathcal{K} \to \mathbb{R}$ be a PL function such that $f(u_1) < \ldots < f(u_l)$. Taking the lower star filtration $K_0 \subset \ldots \subset K_l = \mathcal{K}$, a homology class $[\sigma]$ is either born at K_i or dies at K_i if and only if u_i is a PL critical vertex of $\mathcal{K}$.

The lifetime of a homology class during the filtration is important for topological data analysis.



Regular Vertex | Minimum | 1-saddle | Maximum

Figure 2.7: We have examples of a lower link of vertex v in a 2-submanifold when it is: a regular vertex, a minimum, a saddle and a maximum. The simplices are dashdotted if they are not in either of them. The simplices in blue are in $LLk(v)$, and the yellow ones are in $ULk(v)$. For the first example, v being a regular vertex means $LLk(v)$ and $ULK(v)$ are homotopic to a point, thus $H_0\big(LLk(v)\big) = H_0\big(ULk(v)\big) = 1$ and $H_k\big(LLk(v)\big) = H_k\big(ULk(v)\big) = 0$ for $k \in \{1, 2\}$. For the second case, v is a minimum if $LLk(v) = \emptyset$. v is a 1-saddle, its lower link and upper link are not empty and not connected. Finally, v is a maximum as $ULk(v) = \emptyset$.

Definition 2.52 : Let $\mathcal{K}$ be a triangulation of a r-submanifold $\mathcal{M}$, with vertices $(u_i)_{i\in\{1,\dots,l\}}$ along with a PL function on $|\mathcal{K}|$. Consider the lower star filtration $K_0 \subset \ldots \subset K_l$, and let u_i and u_j be two critical vertices of f such that a homology class $[\sigma]$ is born at K_i and dies at K_j. The lifetime of $[\sigma]$ is called the *persistence* and $pers([\sigma]) = f(u_j) - f(u_i)$. If $[\sigma]$ is born at K_i but never merges with another class, then its persistence is infinite.

There is always at least one class that has infinite persistence, it is the first class born at the global minimum. Fig. 2.8 provides a simple example of a filtration of a 1D scalar function. Now we have everything to define the mathematical object studied in this thesis.

2.3.3 Persistence diagrams

We have shown that the lower star filtration allows us to track the evolution of homology classes of a triangulation $\mathcal{K}$ of a r-submanifold $\mathcal{M}$. We have also characterized the *birth* and the *death* of a homology class during the filtration. Following Prop. 2.13, a homology class can be identified to a pair of critical points $\big(f(u_i), (u_j)\big)$. The persistence diagram is a mathematical object embedding each pair $\big(f(u_i), (u_j)\big)$ to a point in $\mathbb{R}^2$ and the persistence is still encoded as $f(u_j) - f(u_i)$.

Definition 2.53 (*Persistence Diagram* [58]) **:** The k-th persistence diagram of a filtration is defined as a set $\{(x, y) \in \mathbb{R}^2 \mid x > y\}$ (points counted with multiplicity) with x and y encoding respectively the *birth* and *death* of a homology class of the k-th persistent homology group, along with the diagonal $\Delta = \{(x, y) \in \mathbb{R}^2 \mid x = y\}$ counted with infinite multiplicity.

A persistence diagram encodes all the information about the persistent homology groups of a filtration. We know that there is at least one class that has infinite persistence, resulting in theory to have a pair in the persistence diagram of the form $(x, +\infty)$.

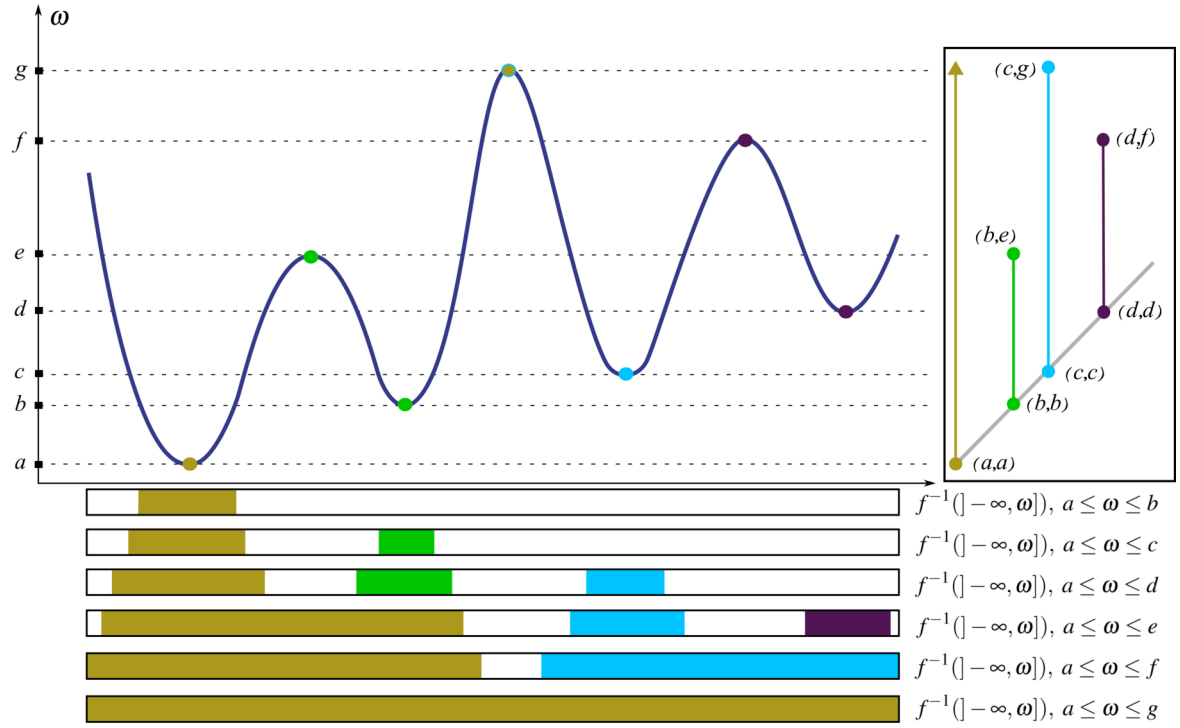


Figure 2.8: This figure present a simple filtration on a 1D function on a compact interval of $\mathbb{R}$. This function has 7 critical points associated to 7 critical values a, b, c, d, e, f and g. There are seven main events during the filtration. First when $\omega = a$ (colored in yellow) a connected component is born (colored in yellow). Second, when $\omega = b$ (green), a second connected component (green) is born while the yellow one grows. Third, at $\omega = c$ (cyan) a third component (cyan) appears with yellow and green still growing. Fourth, at $\omega = d$ (purple) the last component (purple) comes next, with the other three still growing. Fifth, when ω reaches e (green), the yellow and green components merge, the green dies and only the yellow one is still growing according to the elder rule. The lifetime of the green component being $e - b$. Second to last, with $\omega = f$ (purple), the cyan and purple components merge, and its lifetime equals $f - d$ with only the cyan one growing. Last, the last two component merge, letting only the yellow one, and this lifetime is $g - c$. Those different ranges of values gives the lifetimes of the 4 connected components, those lifetimes – called *persistence* – are represented as the lengths of vertical bar codes encoded with the same colors of the critical values (and component) on the right. In particular, this construction on the right is a formal representation of a persistence diagrams in the *"birth/death"* space, the bottom part of the bar codes being on the diagonal, the top part having coordinates (v_{min}, v_{max}) and the persistence encoded as $v_{max} - v_{min}$.

However, in practice, we cut this infinite value of those pairs to the global maximum. This means there is at least on pair involving the global minimum and global maximum, whose persistence is the global range of the scalar function.

A particular persistence diagrams that is interesting to look at is the 0-th persistence diagram. Indeed this diagram tracks the evolution of the 0-th homology groups at all time, and we know that the 0-th homology groups correspond to the connected components of the lower star filtration. This diagram is composed of pairs (x, y) where x is a minimum

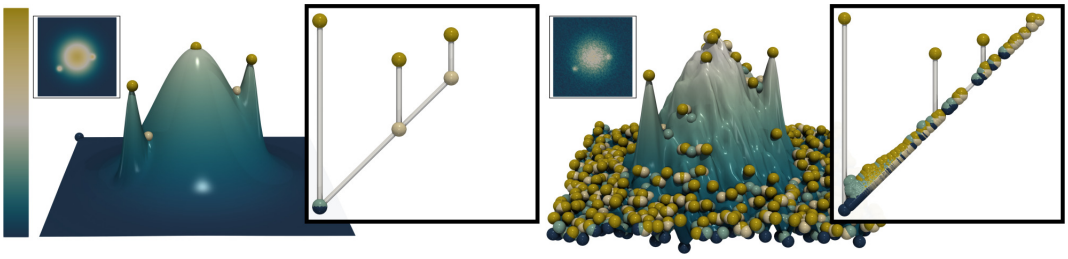


Figure 2.9: Persistence diagrams of a clean (left) and noisy (right) terrain (dark blue spheres: minima, dark yellow: maxima, other: saddles). The three main hills are clearly represented with long bars in the persistence diagrams. In the noisy persistence diagram, small bars encode noise.

point and y a saddle point. Those pairs are interesting because they represent *drainage basins*, or a *pit*, of values in the data. The r-th persistence diagrams is also of interest, this diagrams is composed of pairs (x, y) with x being a saddle point and y a maximum point. Indeed, we have seen that the 0-th persistence diagrams is composed of min-saddle pairs of a PL function f, but if we take $-f$ then a min-saddle pair becomes a saddle-max pair. So if the min-saddle pairs represent drainage basins, then the saddle-max represent *mountains*, or *peaks*, of values in the data.

Pairs with small persistence are more often associated with noise, while the pairs with high persistence correspond to the structures of interests as we can see in Fig. 2.9. However, there are cases where those pairs encode the relevant information of the topology of the initial data (Fig. 3.12). But mostly, we often remove the pairs with small persistence in the diagrams to simplify further computations on them.

2.3.3.1 Wasserstein distance between Persistence Diagrams

There are several metric used to formally compare persistence diagrams [44]. The most commonly used is the Wasserstein distance, which we already introduced in Eq. 2.9. In practice, a persistence diagram is only represented by its pairs outside the diagonal Δ, so we can consider it as a discrete probability measure on $\mathbb{R}^2$. However, recall that we introduced the Wasserstein distance when the optimal transport problem is balanced, and two persistence diagrams can have different numbers of pairs outside of Δ. So we have to pre-process the two of them to make the transport problem balanced.

For a points $v \in \mathbb{R}^2$, we denote by $\pi_\Delta(v)$ its projection onto Δ with respect to the norm $\|.\|$. For $D(f)$ and $D(g)$ two persistence diagrams (from two PL functions f and g), we consider the following *augmented* diagrams:

$$D'(f) = D(f) \cup \{\pi_\Delta(v) \mid v \in D(g)\},$$
$$D'(g) = D(g) \cup \{\pi_\Delta(v) \mid v \in D(f)\}.$$

We see that $|D'(f)| = |D'(g)| = N$. Now that we have balanced the transport problem, let us discuss about the choice of the cost. Indeed, Δ represents the null persistent homology class that is always present in the filtration. Thus two points on the diagonal are associated to the same homology class $[0]$ and are considered as equal. To take that into account we consider the following cost $c_q : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}_+, \quad (x, y) \mapsto \begin{cases} \|x - y\|^q & \text{if } x \notin \Delta \text{ or } y \notin \Delta \\ 0 & \text{otherwise} \end{cases}$

with $q > 1$. Denoting $D'(f) = \frac{1}{K}\sum_{k=1}^{K} \delta_{x_k}$ and $D'(f) = \frac{1}{K}\sum_{k=1}^{K} \delta_{y_k}$, the Wasserstein distance (Eq. 2.9) between the diagrams $D(f)$ and $D(g)$, denoted $W_q^{\mathcal{D}}$,writes as

$$W_q^{\mathcal{D}}\big(D'(f), D'(g)\big) = \left(\inf_{\psi : D'(f) \xrightarrow{bij} D'(g)} \sum_{k=1}^{K} c_q\big(x_k, \psi(x_k)\big) \right)^{1/q},$$

Where ψ, the optimal transport plan, matches persistence pairs of same critical indices. In practice, the Wasserstein distance used is the 2-Wasserstein distance $W_2^{\mathcal{D}}$. Fig. 2.10 illustrates an example of the matching returned by W_2 between two persistence diagrams.

Despite those modifications, the usual Wasserstein distance and the Wasserstein distance for diagrams are equivalent.

Proposition 2.14 : For all D_1, D_2 persistence diagrams (that have been augmented) and $q \in [1, \infty[$:

$$W_q^{\mathcal{D},q}(D_1, D_2) \leq W_q^q(D_1, D_2) \leq 2W_q^{\mathcal{D},q}(D_1, D_2). \tag{2.13}$$

Proof. Let D_1, D_2 be two persistence diagrams supposed augmented. We consider ψ the optimal transport plan with respect to W_q^q. With an abuse of notation we will write $x \in D_1$ to designate a point of the persistence diagram D_1. We have:

$$\begin{aligned} W_q^q(D_1, D_2) &= \sum_{x \in D_1} \|x - \psi(x)\|^q \\ &= \sum_{x \notin \Delta \text{ or } \psi(x) \notin \Delta} \|x - \psi(x)\|^q + \sum_{x \in \Delta \ \& \ \psi(x) \in \Delta} \|x - \psi(x)\|^q. \end{aligned}$$

Then we naturally have:

$$\begin{aligned} \sum_{x \in D_1} c_q\big(x, \psi(x)\big) &\leq \sum_{x \notin \Delta \ \& \ \psi(x) \notin \Delta} \|x - \psi(x)\|^q + \sum_{x \in \Delta \ \& \ \psi(x) \in \Delta} 0 \\ &\leq \sum_{x \notin \Delta \ \& \ \psi(x) \notin \Delta} \|x - \psi(x)\|^q + \sum_{x \in \Delta \ \& \ \psi(x) \in \Delta} \|x - \psi(x)\|^q, \end{aligned}$$

with $c_q(x, y) = \|x - y\|^q$ if either x or y are not on Δ and 0 otherwise. Thus by definition of $W_2^p D$ we have:

$$W_2^{\mathcal{D},q}(D_1, D_2) = \inf_{\phi : D_1 \to D_2} \sum_{x \in D_1} c_q\big(x, \phi(x)\big) \leq \sum_{x \in D_1} c_q\big(x, \psi(x)\big) \leq W_q^q(D_1, D_2). \tag{2.14}$$

For the other inequality, let γ be the optimal transport plan for $W_2^{\mathcal{D},q}(D_1, D_2)$. We have:

$$W_2^{\mathcal{D},q}(D_1, D_2) = \sum_{x \in D_1} c_q\big(x, \gamma(x)\big) = \sum_{x \notin \Delta \text{ or } \gamma(x) \notin \Delta} \|x - \gamma(x)\|^q + \sum_{x \in \Delta \ \& \ \gamma(x) \in \Delta} 0,$$

we then take ϕ the modification of γ such that the summation on $\{x \notin \Delta \text{ or } \gamma(x) \notin \Delta\}$ does not change, and regarding the sum on Δ it rearranges the terms the following way:

- considering $x \notin \Delta$, we denote $\pi_\Delta(x)$ its projection on Δ and $\gamma(x)$ its assigned by γ not in Δ,

- we have $\phi\big(\pi_\Delta(x)\big) = \pi_\Delta\big(\gamma(x)\big)$.

By definition we have

$$\sum_{x\notin\Delta \text{ or } \gamma(x)\notin\Delta} \|x-\gamma(x)\|^q + \sum_{x\in\Delta \,\&\, \gamma(x)} 0 = \sum_{x\notin\Delta \text{ or } \phi(x)\notin\Delta} \|x-\phi(x)\|^q + \sum_{x\in\Delta \,\&\, \phi(x)} 0.$$

Now we notice that:

$$\sum_{x\in\Delta \,\&\, \phi(x)\in\Delta} \|x-\phi(x)\|^q \leq \sum_{x\in\Delta \,\&\, \phi(x)\in\Delta} \|\pi_\Delta^{-1}(x) - \pi_\Delta^{-1}\big(\phi(x)\big)\|^q \leq \sum_{x\notin\Delta \text{ or } \phi(x)\notin\Delta} \|x-\phi(x)\|^q.$$

Thus:

$$\begin{aligned}\sum_{x\notin\Delta \text{ or } \phi(x)\notin\Delta} \|x-\phi(x)\|^q + \sum_{x\in\Delta \,\&\, \phi(x)\in\Delta} \|x-\phi(x)\|^q \\ \leq 2 \sum_{x\notin\Delta \text{ or } \phi(x)\notin\Delta} \|x-\phi(x)\|^q = 2 \sum_{x\notin\Delta \text{ or } \gamma(x)\notin\Delta} \|x-\gamma(x)\|^q.\end{aligned}$$

By definition we finally have:

$$W_q^q(D_1,D_2) = \inf_{\tau:D_1\to D_2} \sum_{x\in D_1} \|x-\tau(x)\|^q \leq 2W_2^{\mathcal{D},q}(D_1,D_2).$$

□

In light of this result, we will drop the notation $W_q^{\mathcal{D}}$ and simply use W_q to denote the q-Wasserstein distance between persistence diagrams for the remainder of this thesis, as both distances are equivalent.

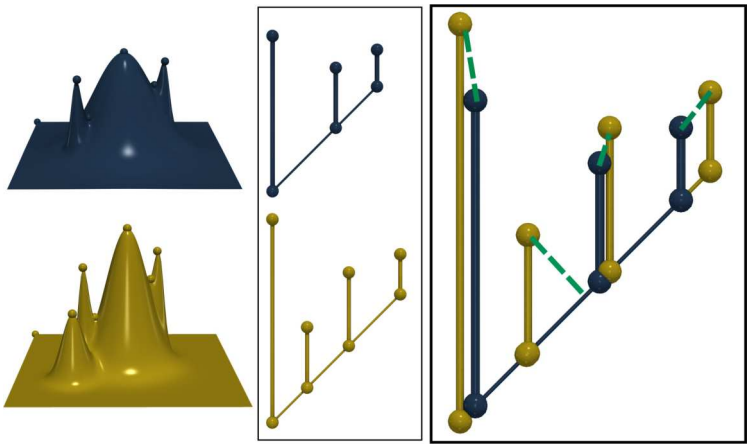


Figure 2.10: Optimal matching (green dashes, right) with regard to W between the two persistence diagrams (center) of two terrains (left).

2.3.3.2 Stability of the Wasserstein distance

One important property, justifying using the Wasserstein distance between persistence diagrams of functions, is the stability of the Wasserstein distance [46].

Definition 2.54 : Let $\mathcal{M}$ be a compact metric space. We say that $\mathcal{M}$ implies bounded degree-k total persistence if there is a constant $C_{\mathcal{M}}$, depending only on $\mathcal{M}$ (and its dimension), such that for every tame (has finite number of separated critical points) Lipschitz continuous function f with Lipschitz constant C_f, we have:

$$\mathrm{Pers}_k(f) = \sum_{x \in D(f)} \mathrm{pers}(x)^k \leq C_f^l C_{\mathcal{M}}.$$

Proposition 2.15 ([46]) **:** Let $\mathcal{M}$ be a compact metric space that implies bounded degree-k total persistence. Let f, g be two tame Lipschitz continuous functions. Then we have for $q > k$ and $p \in [1, \infty[$:

$$W_q^{\mathcal{D}}\big(D'(f), D'(g)\big) \leq C_{\mathcal{M}} \max(C_f, C_g)^k \|f - g\|_\infty^{1-\frac{k}{q}}.$$

This result ensures that the persistence diagrams of two functions cannot be farther to each other than the original functions are; Fig. 2.9 illustrates this stability property. Regarding the constant $C_{\mathcal{M}}$ bounding the total persistence, we have the following expression in a specific case for the 0-th and r-th persistence diagrams when $\mathcal{M}$ is a compact r-submanifold.

Proposition 2.16 : Let $\mathcal{M}$ be a compact smooth r-submanifold in $\mathbb{R}^d$ with negative curvature. Then $\mathcal{M}$ implies bounded degree-r total persistence with $C_{\mathcal{M}} = \frac{\Gamma(\frac{r}{2}+1)}{\pi^{r/2}}\mathrm{Vol}(\mathcal{M})$.

Proof. Without any loss of generality, we suppose $\mathcal{M}$ to be connected, the proof can be generalized by reasoning on each component $\mathcal{M}_i$ otherwise and sum them. Let $f : \mathcal{M} \to \mathbb{R}$ be a tame Lipschitz continuous function.

Let us look at a draining basin A with local minimal value b and saddle value d. We take $f^{-1}(b)$ and $f^{-1}(d)$ the level sets accordingly, we also note that $f^{-1}(b)$ is a singleton as f is tame.

By Hopfin-Rinow theorem [35], there is a minimizing geodesic $\alpha : [0, 1] \to \mathcal{M}$ between $f^{-1}(b)$ and $f^{-1}(d)$. In other words, α is a path such that $\alpha(0) = f^{-1}(b)$, $\alpha(1) \in f^{-1}(d)$ and $\mathrm{Length}(\alpha) = \min_{x \in f^{-1}(d)} d_{\mathcal{M}}\big(x, f^{-1}(b)\big) = d_{min}$. f is Lipschitz continuous, thus $\gamma = f \circ \alpha$ is a Lipschitz and rectifiable curve on $f(A)$. We have

$$\begin{aligned}\mathrm{Length}(\gamma) = \mathrm{Length}(f \circ \alpha) &= \int_0^1 \overline{\lim}_{s \to t} \frac{|f\big(\alpha(s)\big) - f\big(\alpha(t)\big)|}{|s - t|} dt \\ &\leq \int_0^1 C_f \overline{\lim}_{s \to t} \frac{d_{\mathcal{M}}\big(\alpha(s), \alpha(t)\big)}{|s - t|} dt = C_f \mathrm{Length}(\alpha).\end{aligned}$$

The last equality holds because α is a geodesic. As α realizes the geodesic between $f^{-1}(b)$ and $f^{-1}(d)$, A contains a geodesic r-ball of center $f^{-1}(b)$ and radius d_{min}, thus $\mathrm{Vol}\Big(B^k\big(f^{-1}(b), d_{min}\big)\Big) \leq \mathrm{Vol}(A)$.

Introducing the gamma function $\Gamma : \mathbb{R}_+^* \to \mathbb{R}_+^*, \quad z \mapsto \int_{\mathbb{R}_+^*} t^{z-1}e^{-t}dt$, we have:

$$(d-b)^r \leq \text{Length}(\gamma)^r \leq C_f^r d_{min}^r \leq C_f^r \frac{\Gamma(\frac{r}{2}+1)}{\pi^{r/2}} \text{Vol}\Big(B^r\big(f^{-1}(b), d_{min}\big)\Big)$$
$$\leq C_f^r \frac{\Gamma(\frac{r}{2}+1)}{\pi^{r/2}} \text{Vol}(A) = C_f^r \frac{\Gamma(\frac{r}{2}+1)}{\pi^{r/2}} \int_A 1 d\mu_{\mathcal{M}}(s).$$

The third inequality comes from the fact that $\mathcal{M}$ has a negative curvature. The same proof can be done for peaks of values, indeed for d a local maximal value and b a local saddle value for f, then we have $b' = -d$ being a local minimal value and $d' = -b$ being local saddle value for $-f$.

We sum over all the draining basins and peaks, and we have the following:

$$\sum_{x \in D(f)} \text{pers}(x)^r \leq C_f^r \frac{\Gamma(\frac{r}{2}+1)}{\pi^{r/2}} \int_{\mathcal{M}} 1 d\mu_{\mathcal{M}}(s) = C_f^r \frac{\Gamma(\frac{r}{2}+1)}{\pi^{r/2}} \text{Vol}(\mathcal{M}).$$

□

2.3.3.3 Wasserstein barycenter of Persistence Diagrams

The same way a Wasserstein barycenter between probability measures is defined, one can use W_2^2 to define a barycenter of persistence diagrams [187]. Recalling Def. 2.32, a barycenter of persistence diagrams $\mathcal{D} = \{a_1, \ldots, a_m\}$ with barycentric weights $\boldsymbol{\lambda} = \lambda_1, \ldots, \lambda_m$ is defined as

$$Y(\boldsymbol{\lambda}, \mathcal{D}) \in \arg\min_D \sum_{i=1}^m \lambda_i W_2^2(D, a_i). \tag{2.15}$$

Turner & al [187] introduce a simple and efficient way to estimate this barycenter. $Y(\boldsymbol{\lambda}, \mathcal{D})$ is firstly initialized as one element of $\{a_1, \ldots, a_m\}$. Then m optimal transport plans $\psi_1, \ldots, \psi_m$, between each a_i and $Y(\boldsymbol{\lambda}, \mathcal{D})$, are computed. Next, we update $Y(\boldsymbol{\lambda}, \mathcal{D}) = \{p_1, \ldots, p_K\}$ by setting $p_k = \sum_{i=1}^m \lambda_i \psi_i(p_k)$ for all $1 \leq k \leq K$. This sequence of events is iterated until attaining a fixed diagram $Y(\boldsymbol{\lambda}, \mathcal{D})$, which also means that the ψ_i remain unchanged between two iterations. This algorithm results in a diagram as in Fig. 2.11. This algorithm is accelerated by Vidal & al [191] by integrating tailored approximations throughout the computation. Specifically, it approximates the optimal assignments ψ_i with the fast *Auction* optimization [19] (instead of the traditional, yet prohibitive, *Munkres* algorithm [123]). Further, it improves performance with a mechanism called *price memorization*, which enables the initialization of the *Auction* optimization with the assignments ψ_i computed in the previous *Assignment* step. This allows the barycenter optimization to resume the assignment optimization instead of re-computing it from scratch at each iteration. This approach also includes a strategy for the adaptive increase of the *accuracy* parameter of the *Auction* optimization, allowing for fast assignments in the early iterations of the barycenter algorithm, and slower but more accurate assignments towards its convergence.

The computation of the barycenter $Y(\boldsymbol{\lambda}, \mathcal{D})$ requires generalizing the pairwise augmentation described in Sec. 2.3.3.1. Specifically, each non-diagonal point of each diagram a_i is projected to the diagonal of all the other diagrams a_j (with $i \neq j$). After this first augmentation, each diagram a_i contains $K = \sum_{i=1}^{m} |a_i|$ points (where $|a_i|$ is the number of non-diagonal points in a_i). Then $Y(\boldsymbol{\lambda}, \mathcal{D})$ is typically initialized on the diagram a_* which initially minimizes the Fréchet energy. Let $|Y(\boldsymbol{\lambda}, \mathcal{D})| = |a_*|$ be the number of non-diagonal points of a_*. Then, all the non-diagonal points of all the atoms are projected on the diagonal of $Y(\boldsymbol{\lambda}, \mathcal{D})$, and reciprocally, all the non-diagonal points of $Y(\boldsymbol{\lambda}, \mathcal{D})$ are projected on the diagonal of each atom. Thus, at this stage, after this second augmentation, each diagram a_i and the candidate barycenter $Y(\boldsymbol{\lambda}, \mathcal{D})$ contains $K = \sum_{i=1}^{m} |a_i| + |Y(\boldsymbol{\lambda}, \mathcal{D})|$ points (mostly on the diagonal).

Fig. 2.11 gives a simple example of a Wasserstein barycenter of persistence diagrams.

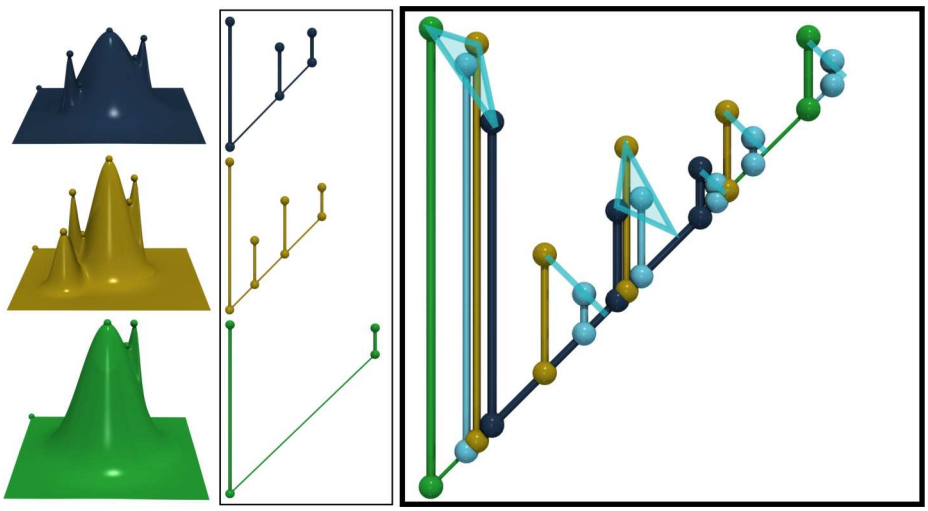


Figure 2.11: Wasserstein barycenter (cyan, uniform weights) of 3 persistence diagrams (center) of 3 terrains (left). Each barycenter point (cyan sphere) is the barycenter of its matched points in the inputs (cyan triangle).

Chapter 3

Wasserstein Dictionaries of Persistence Diagrams

This chapter presents a computational framework for the concise encoding of an ensemble of persistence diagrams, in the form of weighted Wasserstein barycenters [187,191] of a dictionary of *atom diagrams*. We introduce a multi-scale gradient descent approach for the efficient resolution of the corresponding minimization problem, which interleaves the optimization of the barycenter weights with the optimization of the *atom diagrams*. Our approach leverages the analytic expressions for the gradient of both sub-problems to ensure fast iterations and it additionally exploits shared-memory parallelism. Extensive experiments on public ensembles demonstrate the efficiency of our approach, with Wasserstein dictionary computations in the orders of minutes for the largest examples. We show the utility of our contributions in two applications. First, we apply Wassserstein dictionaries to *data reduction* and reliably compress persistence diagrams by concisely representing them with their weights in the dictionary. Second, we present a *dimensionality reduction* framework based on a Wasserstein dictionary defined with a small number of atoms (typically three) and encode the dictionary as a low dimensional simplex embedded in a visual space (typically in 2D). In both applications, quantitative experiments assess the relevance of our framework. Finally, we provide a C++ implementation that can be used to reproduce our results.

The work presented in this chapter has been published in the journal IEEE Transactions on Visualization and Computer Graphics 2024 [173]. It is certified replicable by the Graphics Replicability Stamp Initiative (`https://www.replicabilitystamp.org/index.html#https-github-com-keanu-sisouk-w2-pd-dict`). Our implementation is available at `https://github.com/Keanu-Sisouk/W2-PD-Dict`.

3.1 Context

In addition to the challenge of increased geometrical complexity (discussed above), a new difficulty has recently emerged in many applications, with the notion of *ensemble dataset*. These representations describe a given phenomenon not only with a single dataset, but with a *collection* of datasets, called *ensemble members*. In that context, the topological analysis of an ensemble dataset consequently results in an ensemble of corresponding topological descriptors (e.g. one persistence diagram per ensemble member). Some prominent examples of such ensembles, that are analyzed in this chapter are: the *Isabel* ensemble [148] which is an ensemble of 3D datasets which are obtained from a simulation of the Isabel storm and the *Ionization Front 3D* ensemble [148] which members are describing the movement of ions in a gaseous environment through simulations.

Then, a major challenge consists in developing practical tools for such an ensemble of topological descriptors, to facilitate its processing, analysis and visualization. Such tools include compression approaches (to facilitate the manipulation of the ensemble of descriptors) or visualization methods (for instance, with planar layouts, where each point encodes a descriptor and the distance between a pair of points encodes the intrinsic differences between the corresponding descriptors).

To enable the above tools, a key research question deals with the definition of a concise, yet informative, encoding of the ensemble of descriptors. A promising research direction consists in defining a *dictionary* (i.e. a set of reference descriptors, or *atoms*), such that the topological descriptors of the ensemble can be concisely encoded by expressing them as a specific *function* of the atoms (e.g. a linear combination). At a technical level, this requires to accurately capture and model the implicit relations (i.e. the possible functions) which link the different descriptors of the ensemble.

A series of recent works started the exploration of this overall direction, in particular with the notion of *average topological representation* [104, 149, 187, 191, 199]. These techniques can produce a topological descriptor which nicely summarizes the ensemble. However, they do not capture the implicit relations between the different topological descriptors.

This chapter addresses this issue by introducing a simple and efficient approach for the estimation of linear relations between persistence diagrams on their associated Wasserstein metric space. Inspired by previous work on histograms [167], our approach provides a linear encoding of the input ensemble, where each diagram is represented as a weighted Wasserstein barycenter [187, 191] of a *dictionary* of automatically optimized diagrams called *atom diagrams*. We introduce a novel multi-scale gradient descent algorithm (Sec. 3.4) for the efficient resolution of the corresponding minimization problem (Sec. 3.3), for which we interleave the optimization of the barycenter weights (Sec. 3.3.2) with the optimization of the atom diagrams (Sec. 3.3.3). Extensive experiments (Sec. 3.6) on public ensembles demonstrate the efficiency of our approach, with Wasserstein dictionary computations in the orders of minutes for the largest examples. We illustrate the relevance of our contributions for the visual analysis of ensemble data with two applications, data reduction (Sec. 3.5.1) and dimensionality reduction (Sec. 3.5.2).

3.1.1 Related Work

The literature related to our work can be classified into three main classes: *(i)* uncertainty visualization, *(ii)* ensemble visualization, and *(iii)* topological methods for ensembles.

***(i)* Uncertainty visualization:** Data variability can be represented in the form of *uncertain* datasets, by considering the data at each point of the domain as a random variable, associated with an explicit probability density function (PDF). The analysis and visualization of uncertain data has been recognized as a major challenge in the visualization community [1, 27, 92, 116, 138, 155]. Several techniques have been proposed either dealing with the entropy of the random variables [154], or their correlation [145] or gradient variation [143]. The effect of data uncertainty on feature extraction has also been studied (for instance for level set extraction [14, 15, 144, 151–153, 166]), for various interpolation schemes and PDF models (e.g. Gaussian [113, 134, 135, 141] or uniform [21, 77, 179] distributions). In general, a central limitation of existing methods for uncertain data is their design dependence on a specific PDF model (Gaussian, uniform, etc). This challenges their usability for ensemble data, where the PDFs estimated from the ensemble can follow an arbitrary, unknown model. Moreover, most of these techniques do not consider multimodal PDFs, which are however essential when multiple trends appear in the ensemble.

***(ii)* Ensemble visualization:** Another approach to model data variability consists in using ensemble datasets. In this context, the variability is encoded by a sequence of empirical observations (*i.e* the members of the ensemble). Established techniques typically compute geometrical objects, such as level sets or streamlines, thereby capturing the main features for each member of the ensemble. From there, a *representative* of the resulting ensemble of geometrical objects can be computed. For this task, a few methods have been introduced. For instance spaghetti plots [55] are used in the case of level-set variability, more particularly for weather data [156, 165], and box-plots [120, 194] for the variability of contours and curves. In the case of trend variability, Hummel et al. [91] conceived a Lagrangian framework for classification purposes in flow ensembles. More specifically, clustering techniques have been used to identify the main trends in ensemble of streamlines [65] and isocontours [66]. However, only few techniques have applied this strategy to topological objects. Favelier et al. [62] and Athawale et al. [16] respectively introduced techniques to analyze the geometrical variability of critical points and gradient separatrices. Overlap-based heuristics have been studied for estimating a representative contour tree from an ensemble [101, 197]. In the context of ensembles of histograms, Schmitz et al. [167] introduced a dictionary encoding approach based on optimal transport [48]. However, this method is not directly applicable to persistence diagrams. It focuses on a fundamentally different object (histograms). Thus, the employed distances, geodesics and barycenters are defined differently (in particular in an entropic form [48, 49]) and the algorithms for their computations are drastically different (based on Sinkhorn matrix scaling [172]). In contrast, our work focuses on *Persistence diagrams* (Sec. 3.2.1), whose associated metric space is also inspired from optimal transport, but with various formal and computational specificities (Sec. 3.2.2). Moreover, our approach is based on gradient descent which, from our experience, provides better practical convergence for this kind of problems than quasi-Newton techniques. Finally, we contribute a multi-scale progressive optimization algorithm, which provides improved solutions in comparison to a naive optimization.

***(iii)* Topological methods for ensembles:** To analyze the relations between the per-

sistence diagrams of an ensemble, several key low level notions are required, such as the notion of distance and barycenters between diagrams, for which we review the literature here. Inspired by optimal transport [95, 121], the *Wasserstein* distance between persistence diagrams [58] (Sec. 3.2.2) has been extensively studied [44, 46]. It relies on a bipartite assignment problem, for which exact [123] and approximate [19, 97] implementations are available in open-source [184]. Based on this distance, several approaches have explored the possibility to define a *representative* diagram of an ensemble of persistence diagrams, with the notion of *Wasserstein* barycenter. Turner et al. [187] introduced the first approach for the computation of such a barycenter. Lacombe et al. [104] presented an approach based on entropic transport [48, 49]. However, it requires a pre-vectorization step which is subject to several parameters, and which is not conducive to visualization tasks (features can no longer be individually tracked beyond the pre-vectorization step). In contrast, Vidal et al. [191] introduced a vectorization-free approach which maintains the feature assignments explicitly. It is based on a progressive scheme, which greatly accelerates computation in practice. These concepts have been recently investigated for other topological descriptors, such as merge trees [149, 199]. Recently, several authors have investigated another compact representation of ensembles of topological descriptors, via a basis of representative descriptors. For instance, Li et al. [112] introduce a vectorization for merge trees, which was subsequently used by matrix sketching procedures [195] to create a basis of representative merge trees. In contrast, our work focuses on persistence diagrams (which can encode different features). Also, it directly operates on the Wasserstein metric space of persistence diagrams, thereby avoiding the typical technical difficulties associated with vectorizations (e.g. quantization and/or linearization artifacts, potential stability issues, possible inaccuracies in vectorization reversal, etc.). Pont et al. [150] introduced the notion of principal geodesic analysis of merge trees (and persistence diagrams), with the same overall goal of characterizing the relations between the topological descriptors of an ensemble. In this work, we introduce a different formulation of the problem, which is both simpler (based on the construction of weighted Wasserstein barycenters) and more flexible (our optimization is not subject to complicated constraints such as geodesic orthogonality). This results in a simpler implementation and slightly faster computations (Sec. 3.6).

3.1.2 Contributions

This work makes the following new contributions:

1. *A simple approach for the linear encoding of Persistence Diagrams:* We formulate the linear encoding of an ensemble of persistence diagrams on their associated Wasserstein metric space as a dictionary optimization (Sec. 3.3), which simply optimizes, simultaneously, *(i)* the barycentric weights (Sec. 3.3.2) and *(ii)* the atoms of the dictionary (Sec. 3.3.3).

2. *A multi-scale algorithm for the computation of a Wasserstein dictionary of Persistence Diagrams:* We introduce a novel, efficient algorithm for the optimization of the above dictionary encoding problem. Our algorithm leverages the analytic expressions of the gradient of both of the above sub-problems, to ensure fast iterations. Moreover, in comparison to a naive optimization, our algorithm reaches

solutions of improved energy thanks to a multi-scale strategy. Finally, we leverage shared-memory parallelism to further improve performances.

3. *An application to data reduction:* We present an application to data reduction (Sec. 3.5.1), where the persistence diagrams of the input ensemble are significantly compressed, by solely storing their barycentric weights as well as the atom diagrams.

3.2 Notations

This section presents the notations used for the formalization of this chapter. We first redefine the notation used ton enumerate the elements in the topological data representation that we use - the persistence diagram (Sec. 3.2.1) -, second we recall its associated metric (Sec. 3.2.2) using the new notation. Then we define the notion of Wasserstein barycenter of persistence diagrams (Sec. 3.2.3), which is a core component of our approach (Sec. 3.3).

3.2.1 Persistence diagrams

We refer to Sec. 2.3 for a complete definition of a persistence diagram.

For the sake of simplicity, in the remainder of this *chapter*, we enumerate the points of a diagram X with indices such that $X = \{x^1, \dots, x^K\}$ and we note $i_X = \{1, \dots, K\}$ the set of indices (i.e. the set of all integers going from 1 to K) .

3.2.2 Wasserstein distance

We again refer to Sec. 2.3.3.1 for a complete definition and characterization of the Wasserstein distance between persistence diagrams. We still rewrite the Wasserstein distance with the notation introduced in Sec. 3.2.1. Let us consider $X_1 = \{x_1^1, \dots, x_1^{K_1}\}$ and $X_2 = \{x_2^1, \dots, x_2^{K_2}\}$, recall that persistence diagrams have to be augmented before using the Wasserstein distance. Given an off-diagonal point x (i.e. $b < d$), let Δ be its diagonal projection, specifically: $\Delta = (\frac{b+d}{2}, \frac{b+d}{2})$. Let P_1 and P_2 be the sets of the diagonal projections of the points of X_1 and X_2 respectively. Then, X_1 and X_2 are augmented into $X_1{}'$ and $X_2{}'$ by considering $X_1{}' = X_1 \cup P_2$ and $X_2{}' = X_2 \cup P_1$. This ensures that $|X_1^{'}| = |X_2^{'}| = K$ (which eases distance evaluation). We consider in the remainder of this chapter that the notations X_1 and X_2 refer to *augmented diagrams* (i.e. $|X_1| = |X_2| = K$).

Using those notations, given two persistence diagrams X_1 and X_2 the 2-Wasserstein distance between them is defined as:

$$W(X_1, X_2) = W_2(X_1, X_2) = \min_{\psi : i_X \xrightarrow{bij} i_X} \sqrt{\sum_{j=1}^{K} c(x_1^j, x_2^{\psi(j)})}, \tag{3.1}$$

where ψ, the matching, in this chapter is a bijection of the index set i_X towards itself (i.e. ψ is a permutation of i_X) and $c = c_2$ as in Sec. 2.3.3.1.

3.2.3 Wasserstein barycenter

We give again the definition of a Wasserstein barycenter using the notation introduced previously in Sec. 3.2.1 and Sec. 3.2.2. Given a set of persistence diagrams $\mathcal{D} = \{a_1, \dots, a_m\}$

(which we will call in the remainder *dictionary*), a Wasserstein barycenter (Fig. 2.11) – or Fréchet mean – of the dictionary $\mathcal{D}$ with barycentric weights $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)$ is a diagram, which we note $Y(\boldsymbol{\lambda}, \mathcal{D})$ in the following, which minimizes the Fréchet energy $E_F(B)$:

$$E_F(B) = \sum_{i=1}^{m} \lambda_i W^2(a_i, B).$$

$\boldsymbol{\lambda}$ is such that $\lambda_i \geq 0$ and $\sum_{i=1}^{m} \lambda_i = 1$. We denote Σ_m the simplex of such vectors. Intuitively, $Y(\boldsymbol{\lambda}, \mathcal{D})$ is a diagram which minimizes the above linear combination, given $\boldsymbol{\lambda}$, of its squared Wasserstein distances to the diagrams of the dictionary $\mathcal{D}$.

3.3 Wasserstein Dictionary Encoding

This section formalizes our approach for the Wasserstein dictionary encoding of an ensemble of persistence diagrams. Sec. 3.3.1 provides an overview of our approach, which interleaves barycentric weight optimization ($\boldsymbol{\lambda}$) with atom optimization ($\mathcal{D}$). Finally, Secs. 3.3.2 and 3.3.3 detail the gradient estimation for both sub-problems.

3.3.1 Overview

Let $\{X_1, \ldots, X_N\}$ be the input ensemble of N persistence diagrams. The goal of our approach is to jointly optimize two sub-problems:

- Optimize a set $\mathcal{D}$ of m *reference* persistence diagrams, called the *atoms* of the *Wasserstein dictionary* $\mathcal{D}$;
- Optimize for each input diagram X_n a vector of m barycentric weights $\boldsymbol{\lambda}_n \in \Sigma_m$, in order to accurately approximate X_n with a Wasserstein barycenter $Y(\boldsymbol{\lambda}_n, \mathcal{D})$ (Sec. 3.2.3).

This can be formalized as a joint optimization, where one wishes to find the optimal barycentric weights $\Lambda^* = \boldsymbol{\lambda}_1^*, \ldots, \boldsymbol{\lambda}_N^*$ and the optimal Wasserstein dictionary $\mathcal{D}_* = \{a_1^*, \ldots, a_m^*\}$ (with $m \ll N$), in order to minimize the following *dictionary energy*:

$$E_D(\Lambda, \mathcal{D}) = \sum_{n=1}^{N} W^2\big(Y(\boldsymbol{\lambda}_n, \mathcal{D}), X_n\big). \tag{3.2}$$

Our overall strategy for optimizing Eq. 3.2 consists in iteratively interleaving two sub-optimizations:

1. For a fixed dictionary $\mathcal{D}$, the set of barycentric weights Λ is optimized with one step of gradient descent (Sec. 3.3.2);
2. For a fixed set of barycentric weights Λ, the dictionary $\mathcal{D}$ is optimized with one step of gradient descent (Sec. 3.3.3).

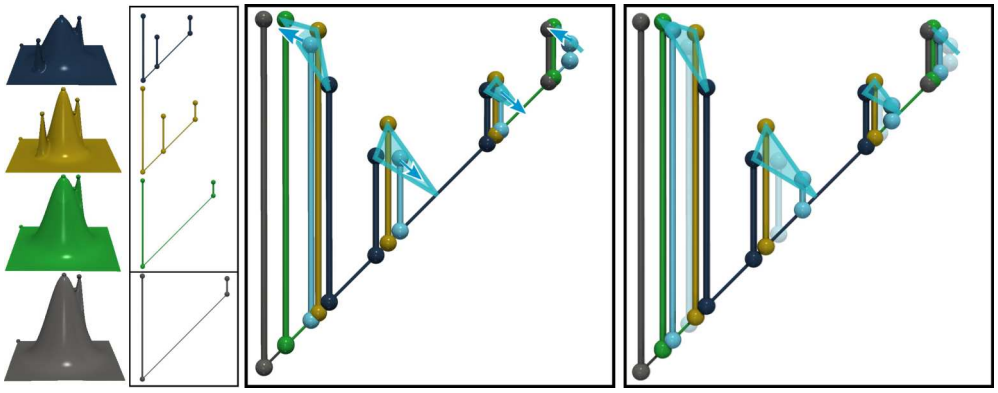


Figure 3.1: Optimizing the weights of the barycenter $Y(\boldsymbol{\lambda})$ (cyan diagram) to improve its approximation of X (grey diagram), given a fixed Wasserstein dictionary $\mathcal{D}$ of 3 atoms (dark blue, yellow, green). At a given iteration t (center), a step $\rho_{\boldsymbol{\lambda}}$ is made along the gradient of the weight energy E_W (cyan arrows), resulting in an improved estimation at iteration $t+1$ (right).

Then, this sequence of two sub-procedures is iterated until a pre-defined stopping condition is reached (Sec. 3.4.2).

Finally, the output of our approach is the optimized Wasserstein dictionary $\mathcal{D}_*$ (a set of m atom diagrams) and, for each input diagram X_n, a vector of weights $\boldsymbol{\lambda}_n^* \in \Sigma_m$, which can be interpreted as the barycentric coordinates of X_n in $\mathcal{D}_*$ (thus capturing linear relations between the input diagrams on the Wasserstein dictionary).

3.3.2 Weight optimization

This section details the optimization of the barycentric weights $\Lambda = \boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_N$. Let $\mathcal{D} = \{a_1, \ldots, a_m\}$ be a fixed dictionary of atom diagrams, with $m > 0$. Let X be a diagram of the input ensemble. For a given set of weights $\boldsymbol{\lambda} = \big(\lambda_1, \ldots, \lambda_m\big)$, let $Y(\boldsymbol{\lambda}) = \big\{y^1(\boldsymbol{\lambda}), \ldots, y^K(\boldsymbol{\lambda})\big\}$ be the barycentric approximation of X, with $K \in \mathbb{N}^*$ the size of $Y(\boldsymbol{\lambda})$, relative to $\mathcal{D}$ (i.e. each point $y^j(\boldsymbol{\lambda})$ of $Y(\boldsymbol{\lambda})$ approximates a point in X).

We recall that after augmentation (Sec. 2.3.3.3), $Y(\boldsymbol{\lambda})$ and the atoms contain $\sum_{i=1}^m |a_i| + |Y(\boldsymbol{\lambda})|$ points each, where $|a_i|$ and $|Y(\boldsymbol{\lambda})|$ denote the number of *non-diagonal* points in a_i and $Y(\boldsymbol{\lambda})$ respectively. Then, in order to compare it to X, $Y(\boldsymbol{\lambda})$ is further augmented by projecting on its diagonal the $|X|$ non-diagonal points of X. Then, at this stage, the size K of $Y(\boldsymbol{\lambda})$ is given by $K = \sum_{i=1}^m |a_i| + |Y(\boldsymbol{\lambda})| + |X|$. We augment similarly X (i.e. by projecting the non-diagonal points of $Y(\boldsymbol{\lambda})$ to its diagonal) and the m atoms (i.e. by projecting the non-diagonal points of X to their diagonals). Then, at this point, $Y(\boldsymbol{\lambda})$, X, and the m atoms a_i all have the same size $K = \sum_{i=1}^m |a_i| + |Y(\boldsymbol{\lambda})| + |X|$.

In this section, we describe a gradient descent on $\boldsymbol{\lambda}$ to minimize the *weight energy*:

$$E_W(\boldsymbol{\lambda}) = W^2\big(Y(\boldsymbol{\lambda}), X\big). \tag{3.3}$$

A step of the corresponding gradient descent is illustrated in Fig. 3.1.

Given the set of optimal matchings $\phi_1, \ldots, \phi_m$ between $Y(\boldsymbol{\lambda})$ and the m atoms, the

j^{th} point of $Y(\boldsymbol{\lambda})$, noted $y^j(\boldsymbol{\lambda})$, is given by:

$$\forall j \in \{1, \dots, K\},\ y^j(\boldsymbol{\lambda}) = \sum_{i=1}^{m} \lambda_i a_i^{\phi_i(j)}. \tag{3.4}$$

In other words, the j^{th} point $y^j(\boldsymbol{\lambda})$ of the diagram $Y(\boldsymbol{\lambda})$ is a linear combination (with the weights $\boldsymbol{\lambda}$) of the m points it matches to in the atoms (one point per atom a_i), under the optimal assignments ϕ_i (i.e. minimizing Eq. 3.1).

For a fixed set of assignments $\phi_1, \dots, \phi_m$, the Wasserstein distance Eq. 3.1 between X and its approximation $Y(\boldsymbol{\lambda})$ is then:

$$E_W(\boldsymbol{\lambda}) = W^2\big(Y(\boldsymbol{\lambda}), X\big) = \sum_{j=1}^{K} c\big(y^j(\boldsymbol{\lambda}), x^{\psi(j)}\big),$$

where ψ denotes the optimal assignment Eq. 3.1 between X and its approximation $Y(\boldsymbol{\lambda})$. When $y^j(\boldsymbol{\lambda})$ and $x^{\psi(j)}$ are not both diagonal points, the cost $c\big(y^j(\boldsymbol{\lambda}), x^{\psi(j)}\big)$ is given by their squared Euclidean distance in the birth/death space (it is zero otherwise, see Sec. 2.3.3.1). Then, by exploiting Eq. 3.4, $E_W(\boldsymbol{\lambda})$ can be re-written as:

$$\begin{aligned} E_W(\boldsymbol{\lambda}) = W^2\big(Y(\boldsymbol{\lambda}), X\big) &= \sum_{j=1}^{K} \|y^j(\boldsymbol{\lambda}) - x^{\psi(j)}\|^2 \\ &= \sum_{j=1}^{K} \left\| \left(\sum_{i=1}^{m} \lambda_i a_i^{\phi_i(j)} \right) - x^{\psi(j)} \right\|^2. \end{aligned}$$

Since $\sum_{i=1}^{m} \lambda_i = 1$, $E_W(\boldsymbol{\lambda})$ can finally be re-written as:

$$E_W(\boldsymbol{\lambda}) = W^2\big(Y(\boldsymbol{\lambda}), X\big) = \sum_{j=1}^{K} \left\| \sum_{i=1}^{m} \lambda_i (a_i^{\phi_i(j)} - x^{\psi(j)}) \right\|^2. \tag{3.5}$$

Intuitively, this energy measures the error (in terms of Wasserstein distance) induced by approximating the input diagram X with its barycentric approximation $Y(\boldsymbol{\lambda})$. In Eq. 3.5, it is computed for each j^{th} point $y^j(\boldsymbol{\lambda})$ of the diagram $Y(\boldsymbol{\lambda})$, by considering the birth/death distances between the points $y^j(\boldsymbol{\lambda})$ maps to, in the atoms on one hand and in the input diagram X on the other.

Then, by applying the chain rule on Eq. 3.5, the gradient of the weight energy Eq. 3.3 is given by:

$$\nabla E_W(\boldsymbol{\lambda}) = 2 \sum_{j=1}^{K} \begin{bmatrix} (a_1^{\phi_1(j)} - x^{\psi(j)})^T \\ \vdots \\ (a_m^{\phi_m(j)} - x^{\psi(j))^T} \end{bmatrix} \big(\lambda_i (a_i^{\phi_i(j)} - x^{\psi(j)})\big). \tag{3.6}$$

Now that the gradient of the weight energy is available Eq. 3.6, we can proceed to gradient descent. Specifically, the barycentric weights at the iteration $t+1$ (noted $\boldsymbol{\lambda}^{t+1}$) are obtained by a step $\rho_{\boldsymbol{\lambda}}$ from the weights at the iteration t (noted $\boldsymbol{\lambda}^t$) along the gradient:

$$\boldsymbol{\lambda}^{t+1} = \Pi_{\Sigma_m}\big(\boldsymbol{\lambda}^t - \rho_{\boldsymbol{\lambda}} \nabla E_W(\boldsymbol{\lambda}^t)\big), \tag{3.7}$$

where Π_{Σ_m} is the projection onto the simplex of admissible barycentric weights (i.e. positive and summing to 1, c.f. Sec. 3.2.3). ∇E_W is L-Lipschitz as stated in the following proposition.

Proposition 3.1 : Let X be a persistence diagram and $\mathcal{D} = (a_1, \dots, a_m)$ a Wasserstein dictionary of persistence diagrams. If the optimal matchings are constant, then $E_W(\boldsymbol{\lambda}) = W^2\big(Y(\boldsymbol{\lambda}), X\big)$ is convex and ∇E_W is L-Lipschitz on Σ_m.

Proof. Let $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m) \in \Sigma_m$, $Y(\boldsymbol{\lambda}) = \big(y^1(\boldsymbol{\lambda}), \dots, y^K(\boldsymbol{\lambda})\big)$ the barycenter computed and $\phi_{\boldsymbol{\lambda},1}, \dots, \phi_{\boldsymbol{\lambda},m}$ the matchings between $Y(\boldsymbol{\lambda})$ and each atom $(a_1, \dots, a_m)$:

$$\forall j \in \{1, \dots, K\},\ y^j(\boldsymbol{\lambda}) = \sum_{i=1}^{m} \lambda_i a_i^{\phi_{\boldsymbol{\lambda},i}(j)}.$$

We suppose the optimal matchings to be constant, thus we write $\phi_i = \phi_{\boldsymbol{\lambda},i}$. We consider the following gradient:

$$\nabla y^j(\boldsymbol{\lambda}) = \begin{bmatrix} a_1^{\phi_1(j)} & \dots & a_m^{\phi_m(j)} \end{bmatrix}.$$

Now recall the following expression for:

$$W^2\big(Y(\boldsymbol{\lambda}), X\big) = \min_{\psi_{\boldsymbol{\lambda}} : i_X \xrightarrow{bij} i_X} \left(\sum_{j=1}^{K} \|y^j(\boldsymbol{\lambda}) - x^{\psi(j)}\|^2 \right).$$

This minimum is always attained, and with the hypothesis on the optimal matchings we write. Thus we rewrite:

$$W^2\big(Y(\boldsymbol{\lambda}), X\big) = \sum_{j=1}^{K} \|y^j(\boldsymbol{\lambda}) - x^{\psi(j)}\|^2 = \sum_{j=1}^{K} \left\| \sum_{i=1}^{m} \lambda_i (a_i^{\phi_i(j)} - x^{\psi(j)}) \right\|^2.$$

$W^2\big(Y(\boldsymbol{\lambda}), X\big)$ is convex with $\boldsymbol{\lambda}$ and the gradient follows naturally:

$$\nabla W^2\big(Y(\boldsymbol{\lambda}), X\big) = 2 \sum_{j=1}^{K} \begin{bmatrix} (a_1^{\phi_1(j)} - x^{\psi(j)})^T \\ \vdots \\ (a_m^{\phi_m(j)} - x^{\psi(j))^T} \end{bmatrix} \big(y^j(\boldsymbol{\lambda}) - x^{\psi(j)}\big).$$

For the following part we denote $H^j = \begin{bmatrix} a_1^{\phi_1(j)} - x^{\psi(j)} & \dots & a_m^{\phi_m(j)} - x^{\psi(j)} \end{bmatrix}$. The Hessian then writes as $H = H(\boldsymbol{\lambda}) = 2 \sum_{j=1}^{K} (H^j)^T H^j$. This shows that $\boldsymbol{\lambda} \mapsto W^2\big(Y(\boldsymbol{\lambda}), X\big)$ is convex. Indeed for $u \in \mathbb{R}^m$ we have:

$$u^T H u = 2 \sum_{j=1}^{K} u^T (H^j)^T H^j u = 2 \sum_{j=1}^{K} \|H^j u\|^2 \geq 0$$

This also shows that ∇E_W is L-Lipschitz with $L = \|H\|$. For numerical reasons, we bound L as follows:

$$L = \|H\| = 2 \left\| \sum_{j=1}^{K} (H^j)^T H^j \right\| \leq 2 \sum_{j=1}^{K} \|(H^j)^T H^j\| = 2 \sum_{j=1}^{K} \|H^j\|^2.$$

Thus for our algorithm, we consider the following gradient step:

$$\rho \leq \left[2 \sum_{j=1}^{K} \|H^j\|^2 \right]^{-1}.$$

□

E_W is also strictly convex by Prop. 3.2:

Proposition 3.2 : Let X be a persistence diagram and $\mathcal{D} = (a_1, \ldots, a_m)$ a Wasserstein dictionary of persistence diagrams. If the optimal matchings are constant and at least $m+1$ points in the atoms are affinely independent (they do not belong to a common affine line), then $E_W(\boldsymbol{\lambda}) = W^2\big(Y(\boldsymbol{\lambda}), X\big)$ is strictly convex.

Proof. Let $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m) \in \Sigma_m$, $Y(\boldsymbol{\lambda}) = \big(y^1(\boldsymbol{\lambda}), \ldots, y^K(\boldsymbol{\lambda})\big)$ the barycenter computed and $\phi_{\boldsymbol{\lambda},1}, \ldots, \phi_{\boldsymbol{\lambda},m}$ the matchings between $Y(\boldsymbol{\lambda})$ and each atom $(a_1, \ldots, a_m)$:

$$\forall j \in \{1, \ldots, K\},\ y^j(\boldsymbol{\lambda}) = \sum_{i=1}^{m} \lambda_i a_i^{\phi_{\boldsymbol{\lambda},i}(j)}.$$

We suppose the optimal matchings to be constant, thus we write $\phi_i = \phi_{\boldsymbol{\lambda},i}$. We have already computed (in Prop. 3.1) the Hessian $H = H(\boldsymbol{\lambda})$:

$$H = 2 \sum_{j=1}^{K} (H^j)^T H^j,$$

with $H^j = \begin{bmatrix} a_1^{\phi_1(j)} - x^{\psi(j)} & \ldots & a_m^{\phi_m(j)} - x^{\psi(j)} \end{bmatrix}$. It is clearly symmetric positive semi definite, and our goal is to show that its kernel $\mathrm{Ker}(H) = \{x \in \mathbb{R}^m \mid Hx = 0\}$ is null, implying that H is in fact symmetric positive definite and thus E_W is strictly convex. For that we will first need to prove the following lemma:

Lemma 3.1 : Let A, B be two symmetric positive semi definite matrices, we have $\mathrm{Ker}(A+B) = \mathrm{Ker}(A) \cap \mathrm{Ker}(B)$.

Proof. Let A and B be two symmetric positive semi definite matrices of size m. First let us take $x \in \mathrm{Ker}(A) \cap \mathrm{Ker}(B)$. We trivially have

$$(A+B)x = Ax + Bx = 0 + 0 = 0,$$

giving us $x \in \mathrm{Ker}(A+B)$, and consequently $\mathrm{Ker}(A) \cap \mathrm{Ker}(B) \subset \mathrm{Ker}(A+B)$.

Now let us take $x \in \mathrm{Ker}(A+B)$. We have

$$x^T(A+B)x = x^T Ax + x^T Bx = 0.$$

But $x^T Ax \geq 0$ and $x^T Ax \geq 0$, as they are positive semi definite, consequently $x^T Ax = 0$ and $x^T Bx = 0$. We will prove that $x \in \mathrm{Ker}(A)$, the proof being the same for B. A is symmetric positive semi definite, thus there exist Q an orthogonal matrix, i.e Q such

$QQ^T = Q^TQ = I_m$, and D a diagonal matrix with positive entries such that $A = QDQ^T$. We have

$$0 = x^TAx = x^T(QDQ^T)x = y^TDy,$$

with $Q^Tx = y$. Also, D being diagonal with positive entries, it admits a square root noted $D^{1/2}$ such that $(D^{1/2})^T = D^{1/2}$ and $D^{1/2}D^{1/2} = D$. This gives us:

$$0 = x^TAx = y^TDy = y^TD^{1/2}D^{1/2}y = \|D^{1/2}y\|^2,$$

and in particular $D^{1/2}y = 0$. This means that $y \in \mathrm{Ker}(D^{1/2})$, and additionally $y \in \mathrm{Ker}(D)$ as

$$Dy = D^{1/2}D^{1/2}y = D^{1/2}0 = 0.$$

Recall that $y = Q^Tx$, this directly implies that $DQ^Tx = 0$ and in particular

$$Ax = QDQ^Tx = Q0 = 0,$$

thus $x \in \mathrm{Ker}(A)$. The same way, we can also prove that $x \in \mathrm{Ker}(B)$, yielding the result $\mathrm{Ker}(A+B) \subset \mathrm{Ker}(A) \cap \mathrm{Ker}(B)$. □

A direct consequence of this lemma is

$$\mathrm{Ker}(H) = \mathrm{Ker}\left(\sum_{j=1}^{K}(H^j)^TH^j\right) = \bigcap_{j=1}^{K}\mathrm{Ker}\big((H^j)^TH^j\big).$$

Also $\mathrm{Ker}\big((H^j)^TH^j\big) = \mathrm{Ker}(H^j)$, indeed if $\zeta \in \mathrm{Ker}\big((H^j)^TH^j\big)$ then $0 = \zeta^T(H^j)^TH^j\zeta = \|H^j\zeta\|_2^2$ giving us $H^j\zeta = 0$, then $\zeta \in \mathrm{Ker}(H^j)$; and trivially $\zeta \in \mathrm{Ker}(H^j)$ yields $(H^j)^TH^j\zeta = (H^j)^T0 = 0$. Now recalling that $H^j = \begin{bmatrix} a_1^{\phi_1(j)} - x^{\psi(j)} & \dots & a_m^{\phi_m(j)} - x^{\psi(j)} \end{bmatrix} \in \mathbb{R}^{2\times m}$, $\zeta = (\zeta_1, \dots, \zeta_m) \in \mathrm{Ker}(H^j)$ if and only if $\zeta_1, \dots, \zeta_m$ are solutions of a system of two equations with m parameters

$$\begin{cases} (a_1^{\phi_1(j)} - x^{\psi(j)})[0]\zeta_1 + \ldots + (a_m^{\phi_m(j)} - x^{\psi(j)})[0]\zeta_m = 0, \\ (a_1^{\phi_1(j)} - x^{\psi(j)})[1]\zeta_1 + \ldots + (a_m^{\phi_m(j)} - x^{\psi(j)})[1]\zeta_m = 0. \end{cases}$$

This implies that $\zeta \in \bigcap_{j=1}^{K}\mathrm{Ker}\big((H^j)^TH^j\big)$ if it is solution of K systems of two equations with m parameters with $K >> m$. Unless at least $K-m$ systems are equivalent, meaning that at least $K-m$ vectors $a_k^{\phi_k(j)} - x^{\psi(j)}$ are colinear, which is not possible with our condition of having $m+1$ points affinely independent, the only solution is $\zeta = 0$. Thus

$$\mathrm{Ker}(H) = \mathrm{Ker}\left(\sum_{j=1}^{K}(H^j)^TH^j)\right) = \bigcap_{j=1}^{K}\mathrm{Ker}\big((H^j)^TH^j\big) = \{0\}.$$

□

Prop. 3.1 and Prop. 3.2 ensure that a gradient step will guarantee an energy decrease as long as we choose $\rho_{\boldsymbol{\lambda}}$ such that:

$$\rho_{\boldsymbol{\lambda}} \leq \left[2\sum_{j=1}^{K}\left\|\begin{matrix}(a_1^{\phi_1(j)} - x^{\psi(j)})^T \\ \vdots \\ (a_m^{\phi_m(j)} - x^{\psi(j)})^T\end{matrix}\right\|^2\right]^{-1} < \frac{1}{L}. \tag{3.8}$$

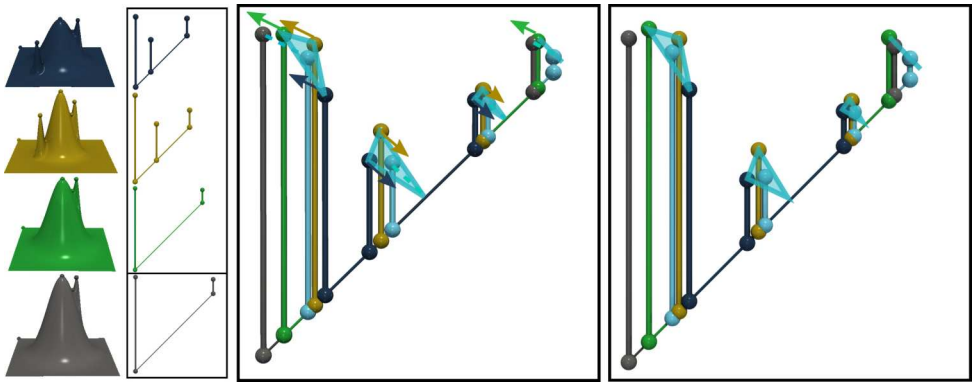


Figure 3.2: Optimizing the atoms of the Wasserstein dictionary $\mathcal{D}$ (dark blue, yellow and green diagrams). At a given iteration t (center), a step $\rho_{\mathcal{D}}$ is made along the gradient of the pointwise atom energy e_A (arrows on each triangle), resulting in a dictionary (right) that enables an improved barycentric approximation ($Y(\mathcal{D})$, cyan) of the input diagram X (grey).

Overall, for a given input diagram X, each iteration t of gradient descent for the optimization of E_W consists in the following steps:

1. Computing the Wasserstein barycenter $Y(\boldsymbol{\lambda}^t)$ (Sec. 3.2.3);
2. Computing the Wasserstein distance $W^2\big(Y(\boldsymbol{\lambda}^t), X\big)$ (Eq. 3.3);
3. Estimating the gradient $\nabla E_W(\boldsymbol{\lambda})$ (Eq. 3.6);
4. Applying one step $\rho_{\boldsymbol{\lambda}}$ of gradient descent (Eq. 3.7).

3.3.3 Atom optimization

This section details the optimization of the atoms of the dictionary $\mathcal{D} = \{a_1, \ldots, a_m\}$. Similarly to Sec. 3.3.2, let X be a diagram of the input ensemble and let $\boldsymbol{\lambda} \in \Sigma_m$ be its – fixed – vector of barycentric weights. For a given dictionary $\mathcal{D}$, let $Y(\mathcal{D}) = \big\{y^1(\mathcal{D}), \ldots, y^K(\mathcal{D})\big\}$ be the barycentric approximation of X, relative to $\boldsymbol{\lambda}$. In this section, we describe a step of gradient descent on $\mathcal{D}$ to minimize the following *atom energy*:

$$E_A(\mathcal{D}) = W^2\big(Y(\mathcal{D}), X\big).$$

A step of the corresponding gradient descent is illustrated in Fig. 3.2.

Given the set of optimal matchings $\phi_1, \ldots, \phi_m$ between $Y(\mathcal{D})$ and the m atoms, the j^{th} point of $Y(\mathcal{D})$, noted $y^j(\mathcal{D})$, is given by:

$$\forall j \in \{1, \ldots, K\},\ y^j(\mathcal{D}) = \sum_{i=1}^{m} \lambda_i a_i^{\phi_i(j)}.$$

This expression is identical to Eq. 3.4 (Sec. 3.3.2). However, y^j now depends on $\mathcal{D}$, which is the variable of the current optimization. Then, the gradient of $y^j(\mathcal{D})$ with regard to $\mathcal{D}$ is simply given by:

$$\nabla y^j(\mathcal{D}) = \begin{bmatrix} \lambda_1 & \cdots & \lambda_m \end{bmatrix}^T. \tag{3.9}$$

For a fixed set of assignments $\phi_1, \ldots, \phi_m$, the Wasserstein distance (Eq. 3.1) between X and its approximation $Y(\boldsymbol{\lambda})$ is then:

$$E_A(\mathcal{D}) = W^2\big(Y(\mathcal{D}), X\big) = \sum_{j=1}^{K} c\big(y^j(\mathcal{D}), x^{\psi(j)}\big),$$

where $\psi(j)$ denotes the optimal assignment between X and its barycentric approximation $Y(\mathcal{D})$. Similarly to Eq. 3.5 (Sec. 3.3.2), the above equation can be re-written as:

$$E_A(\mathcal{D}) = W^2\big(Y(\mathcal{D}), X\big) = \sum_{j=1}^{K} \left\| \sum_{i=1}^{m} \lambda_i (a_i^{\phi_i(j)} - x^{\psi(j)}) \right\|^2 .$$

Let $\mathcal{D}^j = [a_1^{\phi_1(j)}, \ldots, a_m^{\phi_m(j)}]^T$ be the $(m \times 2)$-matrix formed by the atom points matching to a given point $y^j(\mathcal{D})$ of $Y(\mathcal{D})$, via the fixed assignments $\phi_1, \ldots, \phi_m$. Specifically, the i^{th} line of this matrix refers to the point $a_i^{\phi_i(j)}$ in the atom a_i where $y^j(\mathcal{D})$ maps to (via the optimal assignment ϕ_i). For this line, the two columns of the matrix encode the birth/death coordinates of the point $a_i^{\phi_i(j)}$. Then, the *pointwise atom energy* of $y^j(\mathcal{D})$, noted $e_A(\mathcal{D}^j)$, is given by:

$$e_A(\mathcal{D}^j) = \left\| \sum_{i=1}^{m} \lambda_i (a_i^{\phi_i(j)} - x^{\psi(j)}) \right\|^2 . \tag{3.10}$$

Then, by applying the chain rule on Eq. 3.10 (using Eq. 3.9), the gradient of the pointwise atom energy is given by:

$$\nabla e_A(\mathcal{D}^j) = 2 \begin{bmatrix} \lambda_1 & \cdots & \lambda_m \end{bmatrix}^T \big(\sum_{i=1}^{m} \lambda_i (a_i^{\phi_i(j)} - x^{\psi(j)})^T \big). \tag{3.11}$$

Now that the gradient of the pointwise atom energy is available (Eq. 3.11), we can proceed to a step of gradient descent. Specifically, the matrix of atom points matched to $y^j(\mathcal{D})$ at the iteration $t+1$ (noted $\mathcal{D}_{t+1}^j$) is obtained by a step $\rho_{\mathcal{D}}$ from the same matrix at the iteration t (noted $\mathcal{D}_t^j$) along the gradient:

$$\mathcal{D}_{t+1}^j = \Pi_{\mathcal{X}}\big(\mathcal{D}_t^j - \rho_{\mathcal{D}} \nabla e_A(\mathcal{D}_t^j)\big), \tag{3.12}$$

where $\Pi_{\mathcal{X}}$ projects each atom point to an admissible region of the 2D birth/death space (i.e. above the diagonal and within the global scalar field range). Again ∇e_A is L-Lipschitz and e_A is strictly convex:

Proposition 3.3 : Let X be a persistence diagrams and $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m) \in \Sigma_m$. If the optimal matchings are constant, the functions e_A are convex and the gradients ∇e_A are L-Lipschitz.

Proof. Let $U = (u_1, \ldots, u_m) \in (\mathbb{R}^2)^m$, for $j \in \{1, \ldots, K\}$ we have:

$$e_A(U) = \left\| \sum_{i=1}^{m} \lambda_i (u_i - x^{\psi(j)}) \right\|^2$$

The gradient follows naturally:

$$\nabla e_A(U) = 2 \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_m \end{bmatrix} \left(u_i - x^{\psi(j)}\right)^T$$

Immediately we have the Hessian $H_j = H_{g_j}(U) = 2\boldsymbol{\lambda}\boldsymbol{\lambda}^T$, giving us the convexity of e_A and the L-Lipschitzianity of ∇e_A with $L = \|H_j\| \leq 2\|\boldsymbol{\lambda}\|^2 \leq 2m$. For numerical reasons, we consider the larger upper bound: $L \leq 4m$. Thus for our algorithm, we consider the following gradient step $\rho \leq (4m)^{-1}$. □

This ensures that a gradient step will guarantee an energy decrease as long as:

$$\rho_{\mathcal{D}} < (4m)^{-1} < L^{-1}$$

.

Note that, in order to control the final size S_m of the dictionary $\mathcal{D}$, after each iteration of atom optimization, each atom a_i is thresholded by removing its $\overline{K} = (mK - S_m)/m$ least persistent points (at the subsequent optimization iteration, all diagrams will be re-augmented again in a pre-preprocess, as detailed in Sec. 3.3.2).

Overall, for a given input diagram X, each iteration t of gradient descent for the optimization of E_A consists of the following steps:

1. Computing the Wasserstein barycenter $Y(\mathcal{D}_t)$ (Sec. 3.2.3);
2. Computing the Wasserstein distance $W^2\big(Y(\mathcal{D}_t), X\big)$ (Eq. 3.3);
3. For each point $y^j(\mathcal{D})$ of $Y(\mathcal{D})$:
   (a) Estimating the gradient $\nabla e_A(\mathcal{D}^j)$ (Eq. 3.11);
   (b) Applying one step $\rho_{\mathcal{D}}$ of gradient descent on $\mathcal{D}^j$ (Eq. 3.12);
4. Remove the $\overline{K}$ least persistent points from each atom a_i.

3.4 Algorithm

This section presents our overall algorithm for the resolution of the optimization formulated in Sec. 3.3. Sec. 3.4.1 details our initialization strategy. Our overall multi-scale scheme is presented in Sec. 3.4.2. Finally, shared-memory parallelism is discussed in Sec. 3.4.3.

3.4.1 Initialization

Our strategy for the initialization of the Wasserstein dictionary $\mathcal{D}$, illustrated in Fig. 3.3, is inspired by the celebrated *k-means++* strategy [42]. Specifically, we iteratively select the m atoms among the N input diagrams. At the first iteration, we select as first atom the diagram which maximizes the sum of its Wasserstein distances (Eq. 3.1) to all the

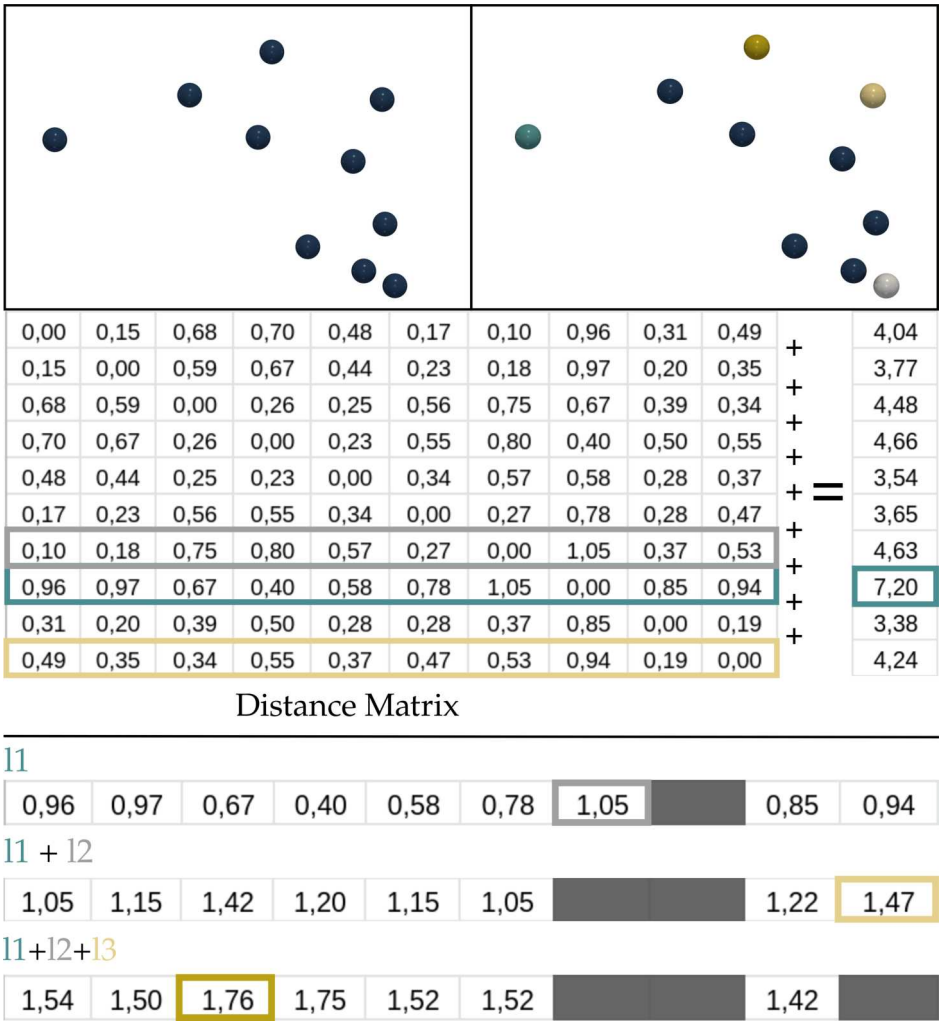


Figure 3.3: Illustration of our initialization strategy on a toy 2D point set (top left). First, the entries of the distance matrix of the input (middle) are summed on a per-line basis. The line maximizing this sum (cyan), noted l_1, identifies the first atom, noted a_1, as the point which is the *furthest away* from all the others (cyan sphere, top right). Next, the atom a_2 (grey sphere, top right) is selected as the point which maximizes its distance to a_1. At this point, the line l_2 (grey, corresponding to the point a_2) is added to the line l_1, to encode the distances to these two atoms (a_1 and a_2). Then, the point a_3 is selected as the maximizer of $l_1 + l_2$: it is the point which is the furthest away from all the previously selected atoms. Then, the corresponding line, l_3, is added to $l_1 + l_2$ and the process is iterated until the target number of atoms has been achieved.

input diagrams (cyan point in Fig. 3.3). Next, each iteration selects as the next atom the diagram which maximizes the sum of its Wasserstein distances to all the previously selected atoms. This process stops when the desired number of atoms, m, has been selected. As illustrated in Fig. 3.3 in the case of a toy 2D point set, this initialization strategy has the nice property that it tends to select atoms on the convex hull of the input point set, which ensures that the non-atom points can indeed be expressed as a convex combination of the atoms, hence leading to accurate initial barycentric approximations. As for the barycentric weights, these are uniformly initialized (i.e. to $1/m$).

3.4.2 Multi-scale optimization algorithm

In real-life data, persistence diagrams tend to contain many low-persistence features, which essentially encode the noise in the data (see Fig. 2.9, right). In this section, we present a multi-scale optimization strategy which addresses this issue by prioritizing the optimization on the most persistent pairs, which correspond to the most salient features of the data. As detailed in Sec. 3.6.2, this strategy leads the optimization to solutions of improved energy in comparison to a naive (non-multi-scale) approach.

Our multi-scale strategy consists in iterating our optimization procedure by progressively increasing the *resolution* (in terms of persistence) of the input diagrams. This is inspired by the progressive strategy by Vidal et al. [191] for the problem of Wasserstein barycenter optimization. Specifically, given an input diagram X of a scalar field $f : \mathcal{M} \to \mathbb{R}$ with $\mathcal{M}$ a manifold, let Δf be the span in scalar values in the corresponding ensemble member (i.e. $\Delta f = \max_{v \in \mathcal{M}} f(v) - \min_{v \in \mathcal{M}} f(v)$). Given a threshold $\tau \in [0, 1]$, we note $X^\tau = \{x \in X | \ d_x - b_x \geq \tau \Delta f\}$ the version of X at resolution τ. It is a subset of X which contains persistence pairs whose relative persistence is above τ. Note that the input diagrams are not normalized by persistence, which would prevent the capture of variability in data ranges within the ensemble. Instead, we normalize the above persistence threshold, by expressing it as a fraction $\tau \in [0, 1]$ of the scalar field range Δf.

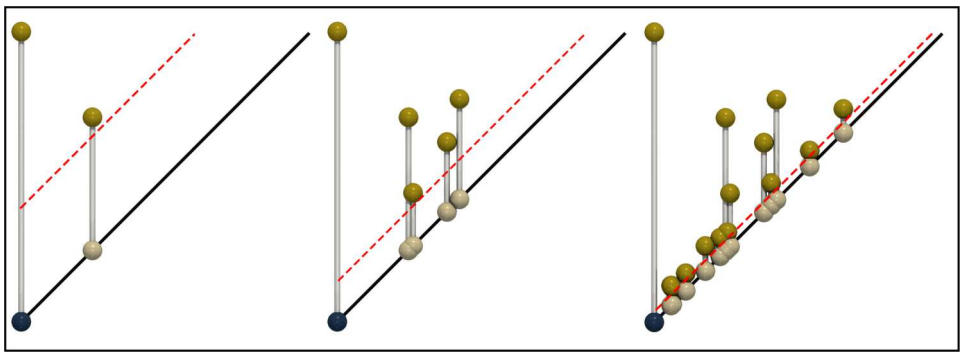


Figure 3.4: Multi-resolution representation of an input persistence diagram (taken from the *Isabel* ensemble). At a given resolution (from left to right), only the persistence pairs above a given persistence threshold (red dash line) are considered in the optimization.

Then, our multi-scale optimization will first consider the input diagrams at a resolution τ_0 and then will progressively consider finer resolutions $\tau_1, \ldots, \tau_r$ until the full diagrams are considered at $\tau_r = 0$. This multi-resolution strategy, based on a per-diagram normalized persistence threshold ($\tau \in [0, 1]$) prevents diagrams from being empty in the early resolutions in case of large variations in data range within the ensemble (which would occur for instance with a per-ensemble normalization). The multi-resolution is illustrated in Fig. 3.4. In our experiments, we set $\tau_0 = 0.2$ and decrease τ by 0.05 at each resolution (i.e. $\tau_1 = 0.15, \tau_2 = 0.10, \tau_3 = 0.05, \tau_4 = 0$). At each resolution, the solution for the previous resolution is used as an initialization. Note that alternative strategies were considered for decreasing τ (for instance by dividing it by 2 at each resolution), but the best experimental results were obtained for the above decrease strategy.

Alg. 1 summarizes our overall approach. For each sub-optimization (i.e. weight and atom optimization), although each gradient step is guaranteed to decrease the corresponding energy (see the end of Secs. Sec. 3.3.2 and Sec. 3.3.3), this is only true for fixed

Algorithm 1: Multi-scale Wasserstein Dictionary Optimization.

Input: Set of persistence diagrams $\{X_1, \ldots, X_N\}$;
Output 1: Wasserstein Dictionary $\mathcal{D}_*$;
Output 2: Barycentric weights $\boldsymbol{\lambda}_1^*, \ldots, \boldsymbol{\lambda}_N^*$;
for $\tau \in \{\tau_0, \ldots, \tau_r\}$ **do**
  if $\tau == \tau_0$ **then**
    Initialization (Sec. 3.4.1);
  end
  while E_D *(Eq. 3.2) decreases* **do**
    for $n \in \{1, \ldots, N\}$ **do**
      Perform a gradient step $\rho_{\boldsymbol{\lambda}_n}$ along ∇E_W relative to X_n (Sec. 3.3.2);
    end
    for $n \in \{1, \ldots, N\}$ **do**
      Perform a gradient step $\rho_{\mathcal{D}}$ along ∇E_A relative to X_n (Sec. 3.3.3);
    end
  end
end

assignments (between a diagram X and its barycentric approximation as well as between the barycentric approximation and the atoms). Since the assignments can change along the iterations of the optimization, the overall energy E_D (Eq. 3.2) may increase between consecutive iterations. Hence, pragmatic stopping conditions need to be considered. In practice, if E_D has not decreased for more than 10 iterations, we return the solutions $\boldsymbol{\lambda}^*$ and $\mathcal{D}_*$ reached by the optimization with the lowest energy E_D.

3.4.3 Parallelism

Our approach can be trivially parallelized with shared-memory parallelism. First, its most computationally demanding task, the N barycentric approximations of the input diagrams can be computed independently. Thus, for each barycentric approximation, we use one parallel task per input diagram. Next, the estimation of the gradient of E_W (Sec. 3.3.2) is done on a per input diagram basis, independently. Thus, we use one parallel task per input diagram. Regarding the estimation of the gradient of E_A (Sec. 3.3.3), given a barycentric approximation $Y(\mathcal{D})$ of an input diagram X, each of its points $y^j(\mathcal{D})$ defines independently a pointwise version of the gradient of the atom energy (see the last paragraph of Sec. 3.3.3). Thus, we use one parallel task per point $y^j(\mathcal{D})$ of a barycentric approximation $Y(\mathcal{D})$ of an input diagram X.

3.5 Applications

This section illustrates the utility of our approach in concrete visualization tasks: data reduction and dimensionality reduction.

3.5.1 Data reduction

Like any data representation, persistence diagrams can benefit from lossy compression. This can be useful in in-situ [17] use-cases, where time-steps are represented on permanent storage with topological signatures [36]. In such scenarios, lossy compression is useful to facilitate the manipulation (i.e. storage and transfer) of the resulting ensemble of persistence diagrams. We present now an application to data reduction where the input ensemble of persistence diagrams is compressed, by only storing to disk:

(i) the Wasserstein dictionary of persistence diagrams $\mathcal{D}_*$ and

(ii) the N barycentric weights $\boldsymbol{\lambda}_1^*, \ldots, \boldsymbol{\lambda}_N^*$.

The compression quality can be controlled with two input parameters *(i)* the number of atoms m and *(ii)* the maximum S_m of the total size of the atoms (i.e. $\sum_{i=1}^{m} |a_i|$). The reconstruction error (given by the energy E_D, Eq. 3.2.) will be minimized for large values of both parameters, while the compression factor will be maximized for low values. In our data reduction experiments, we set the number of atoms m to the number of ground-truth classes of each ensemble, as documented in the ensemble descriptions [149]. Moreover, we set S_m to $c_f^{-1} \sum_{i=1}^{N} |X_n|$, where c_f is a target compression factor and $|X_n|$ is the number of non-diagonal points in the input diagram X_n (see Sec. 3.6.2 for a quantitative evaluation).

Fig. 3.5 (left) provides a visual comparison between the diagram compressed with this strategy (bottom insets) and the original diagram (top insets), for three members of the *Isabel* ensemble. This experiment shows that diagrams can be significantly compressed ($c_f = 5.49$), while still faithfully encoding the main features of the data. Fig. 3.6 (left) provides a similar visual comparison for the *Ionization front (3D)* ensemble ($c_f = 2.9$).

We have applied our data reduction approach to topological clustering [191], where the main trends within the ensemble are identified by clustering the ensemble members based on their persistence diagrams. For the large majority of our test ensembles, the outcome of the clustering algorithm [191] was identical when used with the input diagrams or our compressed diagrams (Sec. 3.6.3 documents a counter-example). This confirms the viability and utility of our data reduction scheme.

3.5.2 Dimensionality reduction

Our framework can also be used to generate low-dimensional layouts of the ensemble, for its global visual inspection. Specifically, we generate 2D planar layouts by using $m = 3$ atoms and by embedding our Wasserstein dictionary $\mathcal{D}_*$ as a triangle in the plane, such that its edge lengths are equal to the Wasserstein distances between the corresponding atoms. Next, each diagram X of the input ensemble is embedded as a point in this triangle by using its barycentric weights $\boldsymbol{\lambda}^*$ as barycentric coordinates.

As illustrated in Figs. 3.5 (right) and 3.6 (right), our dimensionality reduction provides a planar overview of the ensemble which groups together diagrams which are close in terms of Wasserstein distances. Specifically, in both examples, the ground-truth classification of the ensemble is visually respected: the points of a given class (same color) indeed form a distinct cluster in the planar view.

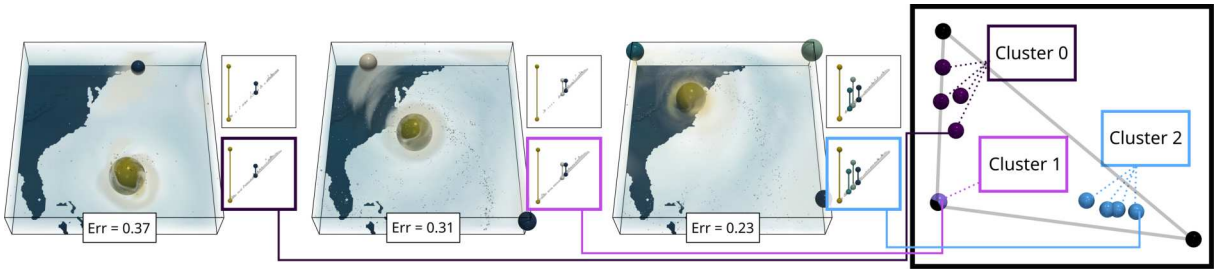


Figure 3.5: Visual comparison (left) between the input persistence diagrams (top insets, saddle-maximum persistence pairs only) and our compressed diagrams (bottom insets, Sec. 3.5.1, saddle-maximum persistence pairs only) for three members of the *Isabel* ensemble (one member per ground-truth class). For each member, the sphere color encodes the matching between the input and the compressed diagrams (for the meaningful persistence pairs, above 10% of the function range). This visual comparison shows that the main features of the diagrams (encoding the main hurricane wind gusts in the data) are well preserved by the data reduction, especially for the members coming from the cluster 2, for which a lower relative reconstruction error (*Err*) can be observed. The planar overview of the ensemble (right) generated by our dimensionality reduction (Sec. 3.5.2) enables the visualization of the relations between the different diagrams of the ensemble. Specifically, this illustration shows a larger disparity for two clusters.

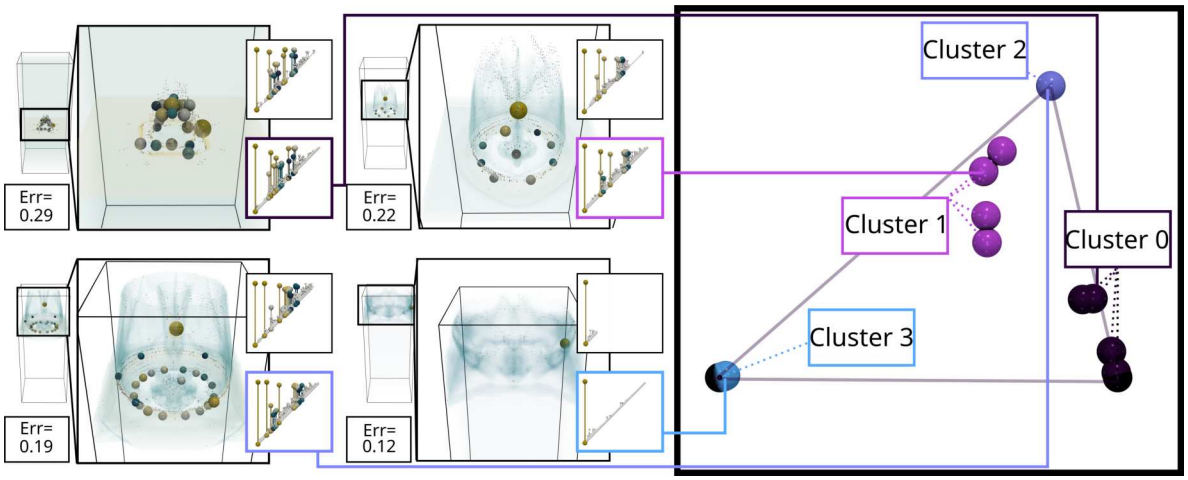


Figure 3.6: Visual comparison (left) between the input persistence diagrams (top insets) and our compressed diagrams (bottom insets, Sec. 3.5.1) for four members of the *Ionization front (3D)* ensemble (one member per ground-truth class). The color encoding is the same as in Fig. 3.5. This visual comparison shows that the main features of the diagrams (the extremities of the ionization front) are well preserved by the data reduction, especially for the members coming from the clusters 2 and 3, for which a lower reconstruction error (*Err*) can be observed. The planar overview of the ensemble (right) generated by our dimensionality reduction (Sec. 3.5.2) enables the visualization of the relations between the diagrams of the ensemble. Specifically, it shows a larger disparity for the clusters 0 and 1 (spread out purple and pink spheres), which are also the most difficult to reconstruct.

3.6 Results

This section presents experimental results obtained on a computer with two Xeon CPUs (3.2 GHz, 2x10 cores, 96GB of RAM). The input persistence diagrams were computed with the *Discrete Morse Sandwich* algorithm [75]. We implemented our approach in C++ (with OpenMP), as modules for TTK [184], [23]. Experiments were ran on the benchmark of public ensembles [148] described in [149], which includes simulated and acquired 2D and 3D ensembles from previous work and past SciVis contests [133]. The considered type of persistence pairs (i.e. the index of the corresponding critical points, Sec. 3.2.1) was selected on a per-ensemble basis, depending on the features of interest present in the ensemble. All types of pairs (i.e. minimum-saddle pairs, saddle-saddle pairs and saddle-maximum pairs) were considered for the following ensembles: *Cloud processes*, *Isabel*, *Starting Vortex*, *Sea Surface Height*, *Vortex Street*. Only the persistence pairs including extrema were considered for the ensembles *Ionization front (2D)* and *Ionization front (3D)*. Finally, only the persistence pairs containing maxima were considered for the remaining ensembles: *Asteroid Impact*, *Dark Matter*, *Earthquake*, *Viscous Fingering*, *Volcanic Eruptions*.

3.6.1 Time performance

The most computationally expensive part of our approach is the computation of the N Wasserstein barycenters, for which we use the algorithm by Vidal et al. [191]. Each iteration of barycenter optimization approximatively requires $\mathcal{O}(mK^2)$ steps in practice (where K is the size of the augmented diagrams, cf. Sec. 3.2.2). As discussed in Sec. 3.4.3, each barycenter is computed in parallel. The evaluations of the gradient of the weight energy (Sec. 3.3.2) and the atom energy (Sec. 3.3.3) both require $\mathcal{O}(NmK)$ steps. As described in Secs. 3.3.2 and 3.3.3, both evaluations can be run in parallel.

Tab. 3.1 evaluates the practical time performance of our multi-scale algorithm for the optimization of the Wasserstein dictionary. In sequential, the runtime is roughly a function of the number of input diagrams (N) as well as their average size ($|X|$). The parallelization of our algorithm (with 20 cores) induces a significant speedup (up to 18 for the largest ensembles), resulting in an average computation time below 5 minutes, which we consider to be an acceptable pre-processing time, prior to interactive exploration. In comparison to the principal geodesic analysis of persistence diagrams (Tab. 1 of [150]), on a per ensemble basis, our approach is 1.56 times faster on average (on the same hardware).

3.6.2 Framework quality

Tab. 3.2 reports compression factors and average relative reconstruction errors for our application to data reduction (Sec. 3.5.1). For each ensemble, the compression factor is the ratio between the storage size of the input diagrams and that of the Wasserstein dictionary $\mathcal{D}_*$ (the m atoms, of average size $|a|$, plus the N sets of barycentric weights). The relative reconstruction error is obtained by considering the Wasserstein distance between an input diagram and its barycentric approximation, divided by the maximum pairwise Wasserstein distance observed in the input ensemble. Then this relative reconstruction error is averaged over all the diagrams of the ensemble. Tab. 3.2 compares a naive optimization (Sec. 3.3) to our multi-scale strategy (Sec. 3.4.2). Specifically, for a given ensemble,

Table 3.1: Running times (in seconds) of our multi-scale algorithm (1 and 20 cores).

Dataset	N	$\|X\|$	1 core	20 cores	Speedup
Asteroid Impact (3D)	20	220	259	35	7.50
Dark matter (3D)	40	216	1,323	188	7.04
Earthquake (3D)	12	97	113	92	1.23
Ionization front (3D)	16	757	4,230	595	7.11
Isabel (3D)	12	1,310	1,609	270	5.96
Viscous Fingering (3D)	15	158	252	49	5.14
Cloud processes (2D)	12	1,176	914	64	14.28
Ionization front (2D)	16	186	145	45	3.22
Sea surface height (2D)	48	1,567	14,587	792	18.42
Starting vortex (2D)	12	125	140	24	5.83
Vortex street (2D)	45	43	1,061	241	4.40
Volcanic eruptions (2D)	12	860	2,798	706	3.96

Table 3.2: Comparison of the average relative reconstruction error (for a common target compression factor), between a naive optimization (Sec. 3.3) and our multi-scale strategy (Sec. 3.4.2). Our multi-scale algorithm improves the error by 30% on average over the naive approach.

Dataset	N	$\|X\|$	m	$\|a\|$	Factor	Error (Naive)	Error (Multi-Scale)
Asteroid Impact (3D)	20	220	4	493	2.20	0.09	0.06
Dark matter (3D)	40	216	4	215	10.87	0.15	0.12
Earthquake (3D)	12	98	3	120	3.05	0.16	0.04
Ionization front (3D)	16	757	4	1,044	2.90	0.29	0.20
Isabel (3D)	12	1,310	3	1,049	5.49	0.34	0.37
Viscous Fingering (3D)	15	158	3	41	2.78	0.15	0.11
Cloud processes (2D)	12	1,176	3	381	5.97	0.38	0.41
Ionization front (2D)	16	186	4	300	2.68	0.38	0.17
Sea surface height (2D)	48	1,567	4	534	20.98	0.54	0.61
Starting vortex (2D)	12	125	2	379	1.98	0.22	0.09
Vortex street (2D)	45	43	5	75	5.08	0.18	0.04
Volcanic eruptions (2D)	12	860	3	345	9.97	0.20	0.20

the same target compression factor was used for both approaches (by imposing the same upper boundary on the total size of the atoms, Sec. 3.5.1). Tab. 3.2 shows that our multi-scale strategy (Sec. 3.4.2) enables the optimization to progress towards better solutions, as assessed by the improvement in reconstruction error of 30% on average. In comparison to the principal geodesic analysis of persistence diagrams (Appendix D of [150]), for the same compression factors, the error induced by our approach is on average 1.79 times larger. However, our approach is simpler, more flexible (our optimization is not subject to restrictive constraints, such as geodesic orthogonality) and slightly faster (Sec. 3.6.1).

Fig. 3.7 provides a visual comparison between the 2D layouts obtained with our approach on the *Isabel* ensemble and those obtained with two typical dimensionality reduction techniques, namely MDS [102] and tSNE [189], directly applied on the distance matrix obtained by computing the Wasserstein distance between all the pairs of input diagrams. For a given technique, to quantify its ability to preserve the *structure* of the ensemble, we run k-means in the 2D layouts and evaluate the quality of the resulting clustering (given the ground-truth [149]) with the normalized mutual information (NMI)

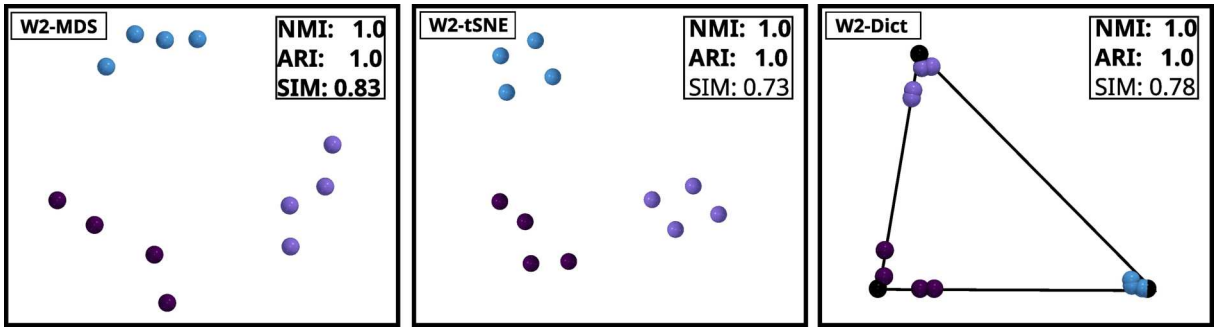


Figure 3.7: Comparison between the 2D layouts obtained with our approach (*W2-Dict*) and these obtained with typical dimensionality reduction approaches (*W2-MDS* [102], *W2-tSNE* [189]) on the *Isabel* ensemble (all persistence pairs are considered). Here, the three approaches preserve well the clusters of the ensemble (NMI/ARI). As expected, *W2-MDS* provides (by design) the best metric preservation (SIM, bold). Our approach constitutes a trade-off between *W2-MDS* and *W2-tSNE*.

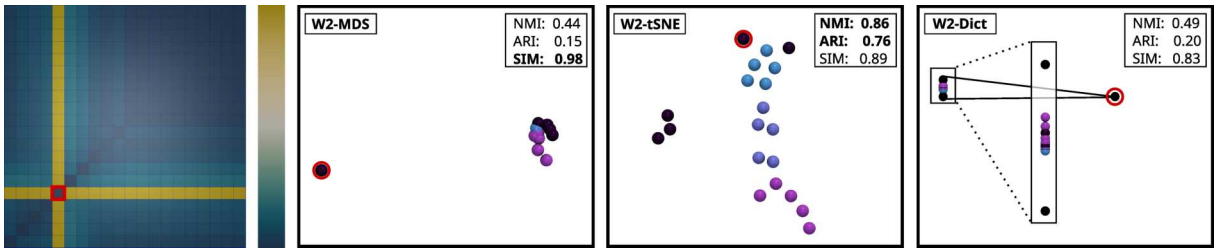


Figure 3.8: Comparison between the 2D layouts obtained with our approach (*W2-Dict*) and these obtained with typical dimensionality reduction approaches (*W2-MDS* [102], *W2-tSNE* [189]) on a *challenging* ensemble. In this example (*Asteroid Impact*), the presence of an outlier (time step of the actual impact, red entry in the distance matrix, left) challenges cluster preservation. While *W2-tSNE* provides the best cluster preservation scores (NMI/ARI), it fails at visually depicting the outlier (red circle) as being far away from the other ensemble members. In contrast, *W2-MDS* and *W2-Dict* do a better job at isolating this outlier (red circle), with *W2-Dict* providing slightly improved cluster preservation scores (NMI/ARI).

and adjusted rand index (ARI). To quantify its ability to preserve the *geometry* of the ensemble, we report the metric similarity indicator SIM [150], which evaluates the preservation of the Wasserstein metric in the 2D layout. All these scores vary between 0 and 1, with 1 being optimal. In Fig. 3.7, the three approaches preserve well the clusters of the ensemble (NMI/ARI) and our approach provides a trade-off between MDS and tSNE in terms of metric preservation (SIM). Fig. 3.8 provides another visual comparison on a *challenging* ensemble (*Asteroid Impact*). There, the presence of an outlier (time step of the actual impact) challenges cluster preservation. While tSNE provides the best cluster preservation (NMI/ARI), it fails at visually depicting the outlier (red circle) as being far away from the other ensemble members. In contrast, *MDS* and our approach do isolate this outlier (red circle), with our approach providing slightly improved cluster preservation (NMI/ARI) over *MDS*. This illustrates the viability of our dimensionality reductions for outlier detection. Appendix A extends this visual analysis to all our test ensembles.

Table 3.3: Detailed layout quality scores (i.e. bold: best values). On average (bottom row), our approach (*W2-Dict*) provides a trade-off between *W2-MDS* and *W2-tSNE*: it preserves the clusters (NMI/ARI) slightly better than *W2-MDS* and the metric (SIM) clearly better than *W2-tSNE*.

	NMI			ARI			SIM		
Dataset	W2-MDS	W2-tSNE	W2-Dict	W2-MDS	W2-tSNE	W2-Dict	W2-MDS	W2-tSNE	W2-Dict
Asteroid Impact (3D)	0.44	**0.86**	0.49	0.15	**0.76**	0.20	**0.91**	0.89	0.83
Dark Matter (3D)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.91**	0.68	0.84
Earthquake (3D)	**0.65**	0.61	**0.65**	0.37	**0.44**	0.37	**0.96**	0.72	0.91
Ionization Front (3D)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.86**	0.71	0.71
Isabel (3D)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.83**	0.73	0.78
Viscous Fingering (3D)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.91**	0.64	0.89
Cloud Processes (2D)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.79**	0.55	0.68
Ionization Front (2D)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	0.78	0.74	**0.83**
Sea Surface Height (2D)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.85**	0.73	0.79
Starting Vortex (2D)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**0.88**	0.72	0.84
Street Vortex (2D)	**1.00**	0.14	**1.00**	**1.00**	-2e-4	**1.00**	0.89	**0.96**	0.81
Volcanic Eruption (2D)	0.66	**1.00**	0.66	0.41	**1.00**	0.41	**0.81**	0.74	0.74
Average	0.896	0.884	**0.900**	0.827	**0.849**	0.832	**0.870**	0.734	0.804

Tab. 3.3 extends our quantitative analysis to all our ensembles. MDS preserves well the metric (high SIM), at the expense of mixing ground-truth classes (low NMI/ARI). tSNE behaves symmetrically (higher NMI/ARI, lower SIM). Our approach provides a trade-off between the extreme behaviors of MDS and tSNE, with a cluster preservation slightly improved over MDS (NMI/ARI), and a clearly improved metric preservation over tSNE (SIM).

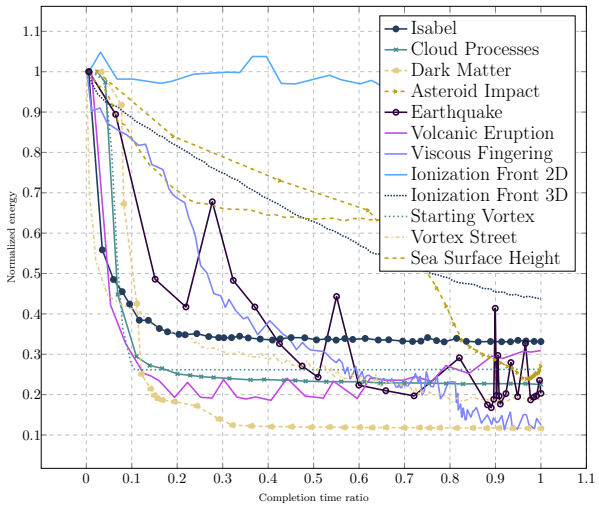


Figure 3.9: Evolution of the (normalized) energy E_D along the optimization, with a naive optimization (Sec. 3.3), for all our test ensembles.

Fig. 3.9 reports the evolution of the normalized energy E_D along the optimization for all test ensembles, for the naive optimization strategy (Sec. 3.3), by using a number of atoms equal to the number of ground-truth classes (cf. our application to data reduction, Sec. 3.5.1). In this figure, the energy is normalized on a per ensemble basis, based on its initial value. This figure shows that the energy does decrease for most ensembles, but still with large oscillations due to the non-convex nature of the dictionary energy E_D. In contrast, the energy evolution with our multi-scale strategy (Fig. 3.10) results in

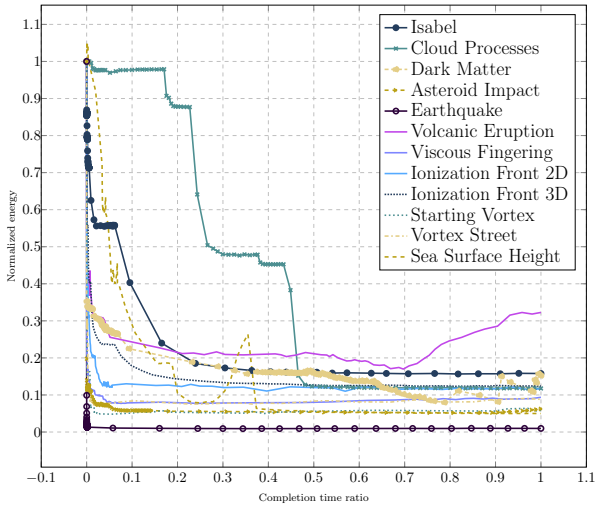


Figure 3.10: Evolution of the (normalized) energy E_D along the optimization, with our multi-scale strategy (Sec. 3.4.2), for all our test ensembles.

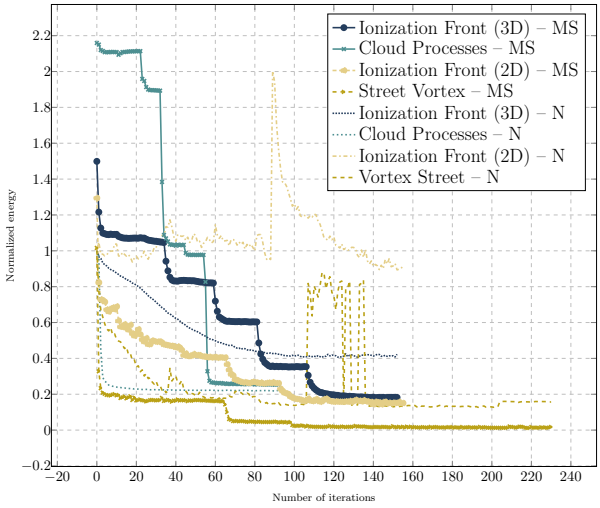


Figure 3.11: Comparison of the evolutions of the (normalized) energy E_D between the naive optimization (Sec. 3.3, *N*, dashed curves) and our multi-scale strategy (Sec. 3.4.2, *MS*, solid curves) for four ensembles. For this experiment, the energy has been normalized with regard to the initial energy of the naive optimization. The *Cloud Processes* ensemble is an example where the naive optimization reaches a solution of slightly lower energy. For the other three ensembles, our multi-scale strategy leads to solutions of much lower energy, through a sequence of characteristic, discontinuous decrease patterns (abrupt drop followed by a plateau) corresponding to the five persistence scales of our multi-scale strategy.

much less oscillations, which indicates the ability of this strategy to help the optimization explore in a more stable manner the locally convex areas of the energy (Appendix B discusses a counter-example). Specifically, in Fig. 3.10, one can observe sequences of discontinuous decrease patterns, characterized by an abrupt drop followed by a plateau. Each of these patterns corresponds to one persistence scale of our multi-scale strategy (this is particularly apparent on the *Cloud Processes* ensemble).

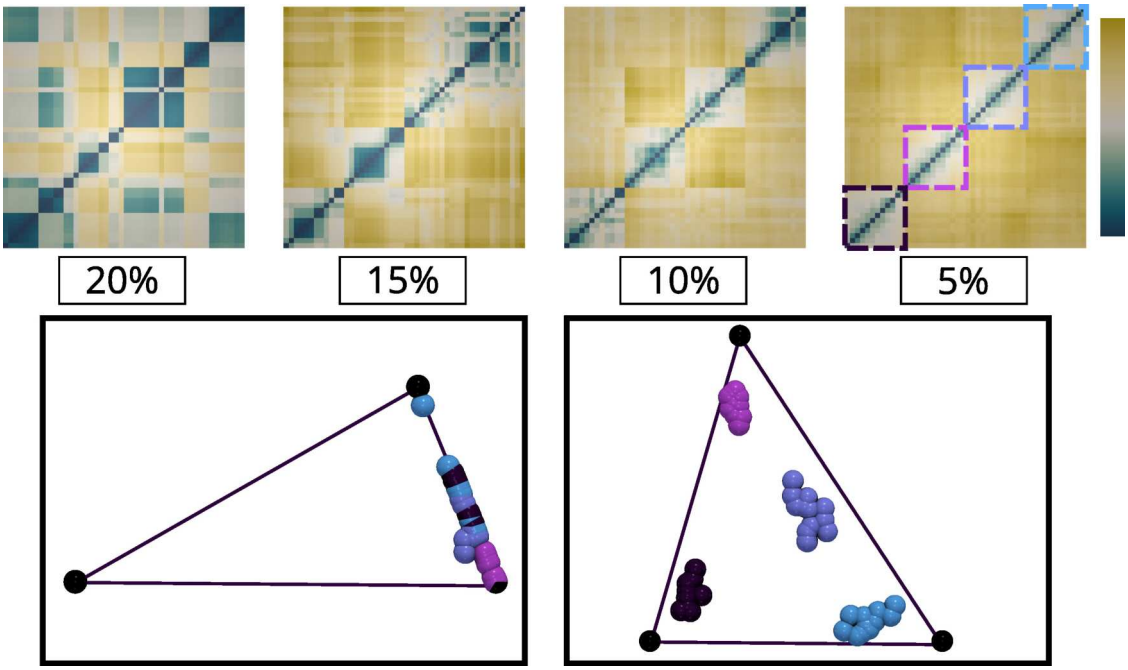


Figure 3.12: Counter-example for our multi-scale strategy (*Sea Surface Height* ensemble). Top: Wasserstein distance matrices for the first four persistence scales of our multi-scale strategy. The ground-truth classes only start to become visible in the distance matrix between the third and fourth scale (dashed sub-matrices in the fourth scale). As a result, our multi-scale strategy is attracted in the first scales towards a local minimum of the energy which does not encode well the ground-truth classes (dimensionality reduction, bottom left). In contrast, the naive optimization manages to reach a solution which separates well the ground-truth classes (dimensionality reduction, bottom right).

Fig. 3.11 provides a closer comparison between the two strategies on a selection of four ensembles. The *Cloud Processes* ensemble is an example where the naive optimization reaches a solution of slightly lower energy. For the other ensembles, our multi-scale strategy leads to solutions of much lower energy, visually confirming the conclusions of Tab. 3.2. In this figure, one can also observe the characteristic decrease patterns discussed above, particularly apparent on the *Ionization Front (3D)* ensemble, which correspond to the distinct scales of our multi-scale strategy.

3.6.3 Limitations

Similarly to other optimization problems based on topological descriptors [149, 150, 187, 191], our energy is not convex. Additionally, as shown in Fig. 3.9, the interleaving of the weight optimization (Sec. 3.3.2) with atom optimization (Sec. 3.3.3) can even lead to oscillations in the energy. As discussed in Sec. 3.6.2, our multi-scale strategy (Sec. 3.4.2) greatly mitigates both issues, with a more stable optimization than a naive approach (Sec. 3.3), which leads to relevant solutions which are exploitable in the applications (Sec. 3.5). However, we have found one example in our test ensembles (the *Sea Surface Height* ensemble), where our multi-scale strategy reached solutions which were arguably worse than these obtained with a naive solution, as described in details in Fig. 3.12. In this

example, the most persistent features in the diagrams are not particularly discriminative for the separation of the ground-truth classes. On the contrary, the variations between these classes seem mostly encoded by the *low* persistence features: in Fig. 3.12 clear separations in the distance matrices between the ground-truth classes only start to occur in the latest persistence scales (dashed sub-matrices, top right inset). This counter-intuitive observation goes against the rule of thumb traditionally used in topological data analysis, which states that the most persistent pairs encode the most important features in the data. For this example, when applying our framework to dimensionality reduction, the non-discriminative aspect of the early persistence scales eventually lead our multi-scale strategy towards a local minium which does not separate the ground-truth classes well (planar layout, bottom left) in comparison to the naive strategy (planar layout, bottom right). Thus, for this ensemble, we reported dimensionality reduction results (Tab. 3.3,Appendix A) obtained with the naive optimization. In general, this means that when users are confronted with ensembles where the most persistent pairs are not the most responsible for data variability (hence class separation), the naive optimization may need to be considered additionally as it might provide solutions which better encode the ground-truth classes.

Finally, as detailed in Appendix B, the presence of clear outliers can also challenge our optimization, especially when the selected number of atoms equals the number of ground-truth classes. Then, in this case, the best dictionary encoding will consequently be obtained by increasing the number of atoms, specifically, by considering that each outlier forms a singleton class.

3.7 Summary

In this chapter, we presented an approach for the encoding of linear relations between persistence diagrams, given the Wasserstein metric. Specifically, we introduced a dictionary based representation of an ensemble of persistence diagrams, inspired by previous work on histograms [167]. We first documented a naive optimization, which interleaves the optimization of the barycentric weights of the input diagrams with the optimization of the atoms of the dictionary (Sec. 3.3). Then, we presented a multi-scale strategy (Sec. 3.4.2) leading to more stable optimizations and relevant solutions (Sec. 3.6.2). We demonstrated the utility of our contributions in applications (Sec. 3.5) to data reduction and dimensionality reduction, where the visualizations generated by our framework enable the visual identification of the main trends in the ensembles (Figs. 3.5, 3.6), and the quick identification of outliers (Fig. 3.8). In contrast to previous work on persistence diagram encoding [150], our framework is simpler, less constrained and slightly faster in practice.

Chapter 4

Robust Barycenters of Persistence Diagrams

This chapter presents a general approach for computing robust Wasserstein barycenters [5, 187, 191] of persistence diagrams. The classical method consists in computing assignment arithmetic means after finding the optimal transport plans between the barycenter and the persistence diagrams. However, this procedure only works for the transportation cost related to the Wasserstein distance W_2. We exploit an alternative fixed-point method [181] to compute a barycenter for generic transportation costs, in particular those robust to outliers. We illustrate the utility of our work in two applications: *(i)* the clustering of persistence diagrams on their metric space and *(ii)* the dictionary-based encoding of ensembles of persistence diagrams [173]. In both scenarios, we demonstrate the added robustness to outliers provided by our generalized framework.

4.1 Introduction

To find a *representative* of an ensemble of persistence diagrams, notions from optimal transport [95, 121] were adapted to persistence diagrams. A central notion is the so-called *Wasserstein* distance [164]. The Wasserstein distance between persistence diagrams [58] (Sec. 4.2.1) has been studied by the TDA community [44, 45]. This distance is computed by solving an assignment problem, for which exact [124] and approximate [19, 97] implementation can be found in open-source [23, 69]. Using this distance, the *Wasserstein* barycenter is used to find a *representative* diagram of an ensemble, [5]. Turner et al. [187] first introduced an algorithm for the computation of such a barycenter for persistence diagrams, along with convergence results and theoretical properties of the persistence diagram space. Lacombe et al. [104] proposed a method to compute a barycenter based on the entropic formulation of optimal transport [48, 49]. However, this method requires a vectorization of persistence diagrams, which is not only subject to parameters, but which also challenges visual analysis and inspection. Indeed, in this case the features of interest cannot be tracked by the users during the analysis. Vidal et al. [191] proposed an approach allowing the tracking of the features. This method is based on a progressive framework, which accelerates the computation time compared to Turner et al.'s method [187]. To take it further, several authors have proposed methods to find a representation of ensembles of topological descriptors by a basis of *representative* descriptors. Li et al. [112] leveraged sketching methods [195] for vectorized merge trees. Pont et al. [150] introduced a principal geodesic analysis method for merge trees, and in Chapt. 3 we brought forth a Wasserstein dictionary encoding method for an ensemble of persistence diagrams. Both latter methods avoid the difficulties associated with vectorizations (e.g. quantization and linearization artifacts, inaccuracies in vectorization reversal). However, all of the above literature propose representatives that can be sensible to the presence of outliers. Turner et al. [186] studied the notion of median of a population of persistence diagrams, but no exact algorithm nor computation was proposed. In this work, we describe a general framework for computing a robust barycenter, by adapting a recent fixed point method [181] from generic probability measures to persistence diagrams. This robust barycenter is more stable to the presence of outliers, thereby enhancing other analysis frameworks such as clustering algorithms or dictionary-based encodings.

4.1.1 Contributions

This work makes the following contributions:

1. *A general framework for robust barycenters of persistence diagrams:* By adapting a recent approach [181] from probability measures to persistence diagrams, we show how barycenter diagrams can be reliably estimated, generic W_q distances (Sec. 4.3), despite outliers (Sec. 4.4).

2. *An application to clustering:* We present an application to clustering (Sec. 4.4.1), where our work yields an improved robustness to the outlier diagrams that are naturally present in ensembles used previously in the literature.

3. *An application to Wasserstein dictionary encoding:* We present an application to dictionary encoding (Sec. 4.4.2), where the added robustness of our generalized

barycenters is demonstrated over standard barycenters.

4. *Implementation:* We provide a Python implementation of our work that can be used for reproducibility purposes.

4.2 Preliminaries

This section presents the required theoretical foundations to our work.

4.2.1 Small reminder on the Wasserstein distance and barycenter and motivation

Recall that for two persistence diagrams X and Y, the q-Wasserstein distance W_q is defined as:

$$W_q(X,Y) = \min_{\phi:X\to Y} \left(\sum_{\ell=1}^{K} c_q\big(x_\ell, \phi(x_\ell)\big) \right)^{1/q}, \tag{4.1}$$

where $\phi : X \to Y$ is a bijection between X and Y. The *transportation cost* c_q, based on powered distances, is such that $c_q(x,y) = \|x-y\|_2^q$ if $x \notin \Delta$ or $y \notin \Delta$, and 0 otherwise. An optimal bijection ϕ^* minimizing Eq. 4.1 is called an optimal transport plan. In practice, q is often set to 2, yielding the Wasserstein distance, noted W_2. As in Sec. 2.3.3.3, a Wasserstein W_q barycenter of persistence diagrams $X_1, \ldots, X_m$ [187], is defined by minimizing the Fréchet energy:

$$\arg\min_{B} \sum_{i=1}^{m} \lambda_i W_q^q(B, X_i), \tag{4.2}$$

where $\lambda_i \geq 0$ and $\sum_i \lambda_i = 1$. Practical algorithms have been proposed for the computation of Wasserstein barycenters [187, 191], but only for the specific case where $q = 2$. When $q = 2$, a solution B^* of Eq. 4.2 has the following property: a point $x \in B^*$ is an arithmetic mean of m points each in $X_1, \ldots, X_m$, which considerably eases the optimization of Eq. 4.2. Fig. 4.1 (center) presents an example of a W_2 barycenter. A barycenter computed using the W_2 distance emulates the behavior of a mean of an ensemble of scalars. As such, it is prone to the influence of outliers. This motivates the use of a more general framework for computing robust barycenters, as detailed in Sec. 4.3.

4.3 Robust barycenter

This section presents a general framework for computing barycenters of persistence diagrams using W_q distances, for arbitrary q values such that $q > 1$.

4.3.1 Optimization

The optimization of Eq. 4.2 can be addressed by an iterative algorithm Alg. 3, where each iteration involves two steps. First, an *assignment step* computes the optimal assignment given the W_q metric between each input diagram and the current barycenter

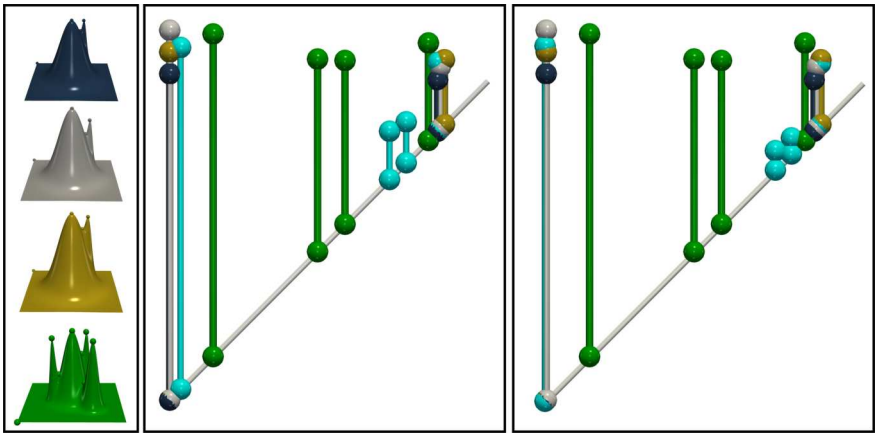


Figure 4.1: Comparison of barycenters computed with different values of q. On the left we have terrain views of four scalar fields colored in blue, gray, yellow and green, the latter being an outlier (featuring more peaks). The corresponding persistence diagrams are represented with matching colors and the barycenters are represented in cyan. The barycenter with $q = 2$ (center) is more sensitive to the presence of the green outlier, with two cyan bars of medium persistence, due to the outlier peaks in the green dataset. For $q = 1.5$ (right), the persistence of these two bars is significantly reduced, and so will be their importance in distance computations.

estimation. Second, in the *update step*, the Fréchet energy is minimized by computing, for each barycenter point, its *ground barycenter* in the birth/death plane. This is achieved by updating each barycenter point to its optimal location, given the assignments computed in the previous step.

For $q = 2$, the ground barycenter can be simply obtained by computing, for each barycenter point, the arithmetic mean of its assigned points in the input diagrams [187, 191]. However, for $q \neq 2$, such a simple update procedure cannot be considered. As illustrated in Fig. 4.2, an update based on the arithmetic mean may increase the Fréchet energy for $q \neq 2$, hence potentially preventing Alg. 3 from converging toward a satisfying result.

Instead, to generalize the computation of ground barycenters for $q \neq 2$, we consider the following function, representing the ground barycenter in the birth/death plane:

$$\mathfrak{b}_q : \begin{cases} \mathbb{R}^2 \times \ldots \times \mathbb{R}^2 \to \mathbb{R}^2 \\ (y_1, \ldots, y_m) \mapsto \arg\min_x \sum_{i=1}^{m} \lambda_i c_q(x, y_i) \end{cases} . \tag{4.3}$$

To optimize Eq. 4.3, as suggested by Tanguy et al. [181], we leverage a fixed-point optimization method Alg. 2, which we plug into Alg. 3 in the *update step* (ground barycenter computation line). In practice we give a maximum number T of overall iterations. We noticed that taking $T < 10$ is sufficient for convergence. Assumptions for achieving convergence are discussed in the next section.

Algorithm 2: Ground barycenter computation algorithm.

Input: Set of points in $\mathbb{R}^2$ $\{y_1, \ldots, y_m\}$, barycentric weights $\lambda_1, \ldots, \lambda_m$, and iteration number T.
Output : Ground barycenter $b_q^{(T)}$.
Initialization: $b_q^{(0)} = \sum_{i=1}^{m} \lambda_i y_i$.
for $0 \leq t \leq T-1$ **do**
  $z = 0$
  $w = (0, \ldots, 0)$
  for $i \in \{1, \ldots, m\}$ **do**
    $d_i = \|b_q^{(t)} - y_i\|^{2-q}$
    $w = w + \lambda_i * y_i / d_i$
    $z = z + \lambda_i / d_i$
  end
  $b_q^{(t+1)} = w/z$
end

Algorithm 3: Barycenter computation algorithm.

Input: Set of persistence diagrams $\{X_1, \ldots, X_m\}$, barycentric weights $\lambda_1, \ldots, \lambda_m$, and iteration number T.
Output : Wasserstein barycenter $B^{(T)} = \{x_1^{(T)}, \ldots, x_K^{(T)}\}$.
Initialization: $B^{(0)} = X_1$.
for $0 \leq t \leq T-1$ **do**
  // 1. Assignment step
  for $i \in \{1, \ldots, m\}$ **do**
    Compute $\phi_i \in \underset{\phi: B^{(t)} \to X_i}{\arg\min} \sum_{\ell=1}^{K} c_q\big(x_\ell^{(t)}, \phi(x_\ell^{(t)})\big)$.
  end

  // 2. Update step
  for $\ell \in \{1, \ldots, K\}$ **do**
    // Ground barycenter computation
    Find $x_\ell^{(t+1)} = \mathfrak{b}_q\big(\phi_1(x_\ell^{(t)}), \ldots, \phi_m(x_\ell^{(t)})\big)$.
  end
end

4.3.2 Convergence

Tanguy et al. prove, in the setup of probability measures, that their fixed-point method for minimizing Eq. 4.3 converged under certain assumptions [181]. In this section, we review these assumptions in the setup of persistence diagrams to argue the convergence of our overall approach.

Assumption 4.1 : For all $(y_1, \ldots, y_m) \in \mathbb{R}^2 \times \ldots \times \mathbb{R}^2$, for all $\lambda_1, \ldots, \lambda_m$ barycentric

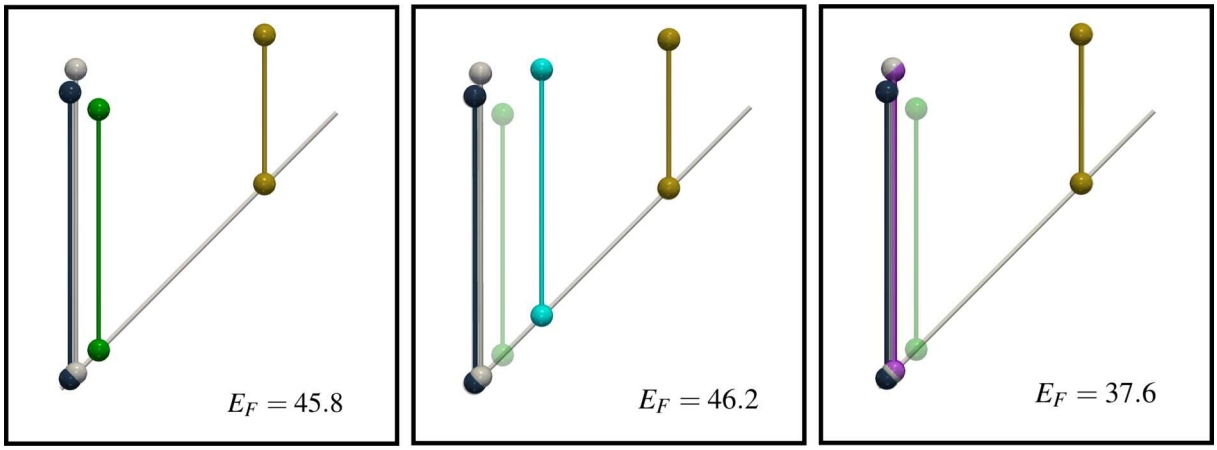


Figure 4.2: Simple example where computing the arithmetic mean instead of optimizing $\mathfrak{b}_q$ increases the Fréchet energy (noted E_F) for $q = 1$. We have three simple persistence diagrams, in dark blue, gray and yellow, each having a single point. For this problem, the transport plans are fixed and the barycenter has only one point. On the left we initialized the barycenter as the diagram encoded in green. In the middle, we have the candidate of the barycenter encoded in cyan when computing an arithmetic mean after one iteration. We can see that the Fréchet energy (for $q = 1$) increased. On the right, we have a candidate for the barycenter encoded in purple when optimizing $\mathfrak{b}_q$ instead, this time displaying a decrease of the Fréchet energy at one iteration.

coefficients, $\arg\min_x \sum_{i=1}^m \lambda_i c_q(x, y_i)$ is reduced to a single element.

We prove that this assumption is satisfied in our case under some conditions.

Conditions 4.1 : Let $y_1, \ldots, y_m \in \mathbb{R}^d$ and $(\lambda_1, \ldots, \lambda_m) \in (0,1)^m$ such that $\sum_i \lambda_i = 1$. Then for $q \in (1, +\infty)$, the function defined as:

$$V_q := x \longmapsto \sum_{i=1}^m \lambda_i \|x - y_i\|_2^q,$$

has a unique minimiser in $\mathbb{R}^d$. If $m \geq 3$ and there exists $i_1 < i_2 < i_3 \in \{1, \ldots, m\}$ such that the points y_{i_1}, y_{i_2} and y_{i_3} are not on a common affine line, then V_1 also has a unique minimiser.

Proof. — Step 1: Case $q > 1$.

For $q > 1$ and a fixed $y \in \mathbb{R}^d$, introduce the function $h_q := x \longmapsto \|x - y\|_2^q$. We begin by showing that h_q is strictly convex. Take $x_1, x_2 \in \mathbb{R}^d$ and $t \in (0,1)$. We re-write:

$$h_q(tx_1 + (1-t)x_2) = \|t(x_1 - y) + (1-t)(x_2 - y)\|_2^q.$$

Introduce $u := x_1 - y$ and $v := x_2 - y$. By convexity of $\|\cdot\|_2$, we have $\|tu + (1-t)v\|_2 \leq t\|u\|_2 + (1-t)\|v\|_2$. We consider the equality and strict inequality cases separately.

1. If $\|tu + (1-t)v\|_2 < t\|u\|_2 + (1-t)\|v\|_2$, we use consecutively the fact that $a \longmapsto a^q$ is increasing and convex on $\mathbb{R}_+$:

$$\begin{aligned}\|tu + (1-t)v\|_2^q &< (t\|u\|_2 + (1-t)\|v\|_2)^q \\ &\leq t\|u\|_2^q + (1-t)\|v\|_2^q.\end{aligned}$$

2. If $\|tu+(1-t)v\|_2 = t\|u\|_2+(1-t)\|v\|_2$, equality in the triangle inequality for $\|\cdot\|_2$ yields that u and v are positively co-linear, leading to the two following alternatives:

   (a) If $v = 0$ then we show the strict inequality as follows:

   $$\|tu\|_2^q = t^q\|u\|_2^q < t\|u\|_2^q,$$

   where the inequality comes from the fact that $q > 1$ and $t \in (0,1)$, with $u \neq 0$ (indeed, if $u = 0$ then we have $u = v = 0$ yielding $x_1 = x_2$ which is a contradiction).

   (b) If $v \neq 0$ then there exists $\alpha \geq 0$ such that $u = \alpha v$. This implies that $\|u\|_2 \neq \|v\|_2$: if equality held, then since $\alpha \geq 0$ we obtain $\alpha = 1$, then $u = v$ yields $x_1 = x_2$ which is a contradiction. Since $\|u\|_2 \neq \|v\|_2$, we can apply the strict convexity of $a \longmapsto a^q$ on $\mathbb{R}_+$, which shows:

   $$\begin{aligned}\|tu+(1-t)v\|_2^q &= (t\|u\|_2+(1-t)\|v\|_2)^q \\ &< t\|u\|_2^q + (1-t)\|v\|_2^q.\end{aligned}$$

In all cases, we obtain the inequality:

$$h_q(tx_1+(1-t)x_2) < th_q(x_1)+(1-t)h_q(x_2),$$

showing strict convexity of h_q. As a convex combination of strictly convex functions, V_q is strictly convex. Since V_q is coercive, we conclude that it admits a unique minimiser.

— *Step 2*: Case $q = 1$.

We now assume that $m \geq 3$ and that there exists $i_1 < i_2 < i_3 \in \{1,\dots,m\}$ such that y_{j_1}, y_{j_2} and y_{i_3} are not on a common affine line. We prove that V_1 is strictly convex: let $x_1 \neq x_2 \in \mathbb{R}^d$ and $t \in (0,1)$, by the triangle inequality (similarly to the case $q > 1$):

$$V_1\big(tx_1+(1-t)x_2\big) \leq \sum_{i=1}^m \lambda_i\big(\|t(x_1-y_i)\|_2 + \|(1-t)(x_2-y_i)\|_2\big).$$

Our objective is to show that in this case, the inequality is strict. There is equality if and only if for each $i \in \{1,\dots,m\}$, $x_2 - y_i = 0$ or there exists $\alpha_i \in \mathbb{R}_+$ such that $x_1 - y_i = \alpha_i(x_2 - y_i)$. We now reason by contradiction and assume that equality holds. We distinguish two cases concerning the points $(y_{i_k})_{k=1}^3$ from the assumption.

1. If there exists $k \in \{1,2,3\}$ such that $x_2 - y_{i_k} = 0$, without any loss of generality we take $y_{i_1} = x_2$, then $y_{i_1} = x_2$, furthermore the assumption on $(y_{i_k})_{k=1}^3$ implies that:

   $$\forall i \in \{i_2, i_3\},\ x_2 - y_i \neq 0.$$

   Using these properties, we deduce from the equality in the triangle inequality that there exists $\alpha_{i_k} \geq 0$ such that $x_1 - y_{i_k} = \alpha_{i_k}(x_2 - y_{i_k})$ for $k \in \{2,3\}$. Note that the $\alpha_{i_k} \neq 1$, otherwise $x_1 - y_{i_k} = x_2 - y_{i_k}$ which is impossible since $x_1 \neq x_2$. We can now re-rewrite the equality as

   $$y_{i_k} = x_1 + \frac{\alpha_{i_k}}{1-\alpha_{i_k}}(x_1 - x_2),$$

   concluding that y_{i_2}, y_{i_3} and x_2 are on the common affine line $x_1 + \mathbb{R}(x_1 - x_2)$, which is a contradiction as $x_2 = y_{i_1}$. We conclude that the equality cannot hold in the triangle inequality in this case.

2. If $\forall k \in \{1,2,3\},\ x_2 - y_{i_k} \neq 0$, then the triangle equality condition provides the existence of $\alpha_{i_k} \geq 0$ such that $x_1 - y_{i_k} = \alpha_{j_k}(x_2 - y_{i_k})$. Again $\alpha_{i_k} \neq 1$ for the same reasons discussed above. Similarly we re-write the equality as:

$$y_{i_k} = x_1 + \frac{\alpha_{i_k}}{1-\alpha_{i_k}}(x_1 - x_2),$$

concluding that the three points $(y_{i_k})_{k=1}^3$ are on the common affine line $x_1+\mathbb{R}(x_1-x_2)$ which is a contradiction, hence equality cannot hold in the triangle inequality.

In both cases, we have proven that equality in the triangle inequality cannot happen, yielding:

$$V_1\big(tx + (1-t)x_2\big) < tV_1(x) + (1-t)V_1(x_2),$$

which shows the strict convexity of V_1. Since V_1 is coercive, we can conclude that it admits a unique minimiser. □

Remark 4.1 : When $q = 1$ it is crucial that at least three points are not on a common affine line. For instance if we have only two points, and take $\lambda_1 = \lambda_2 = 1/2$, then $\arg\min_x \|x - y_1\|_2 + \|x - y_2\|_2 = \{ty_1 + (1-t)y_2 \mid t \in [0,1]\}$.

Notice that when $q = 1$, this assumption does not hold in general. (due to the presence of collinear points, in particular on the diagonal). In practice, this may lead to numerical instabilities when considering $q = 1$, especially when ground barycenters are not unique.

Under those assumptions, and denoting the Fréchet energy $E_F(B) = \sum_{i=1}^{m} \lambda_i W_q^q(B, X_i)$, Tanguy et al. [181] show that for two consecutive iterates $B^{(t)}$ and $B^{(t+1)}$, we have $E_F(B^{(t)}) \geq E_F(B^{(t+1)})$. This means that fixed-point iterations $(B^{(t)})$ decrease the energy optimized in Eq. 4.2.

For $q \in [1,2[$, a resulting fixed point is a barycenter that is more robust to the presence of an outlier, in the initial set $X_1, \ldots, X_m$, compared to a W_2 barycenter. Fig. 4.1 illustrates this difference when computing a barycenter with an outlier.

4.4 Results

This section presents two applications of our robust barycenters (Sec. 4.3) along with detailed experiments. The experimental results are obtained on a computer with a NVIDIA Geforce RTX 2060 (Mobile Q) with 6 GB of dedicated VRAM. Our methods were implemented on Python, using Pytorch for computations on the GPU. We ran some experiments on two public ensembles [148] described in [149]. One is an acquired 2D ensemble and the other is a simulated 3D ensemble, both selected from past SciVis contests [133]. For the experiments, only the persistence pairs containing maxima were considered.

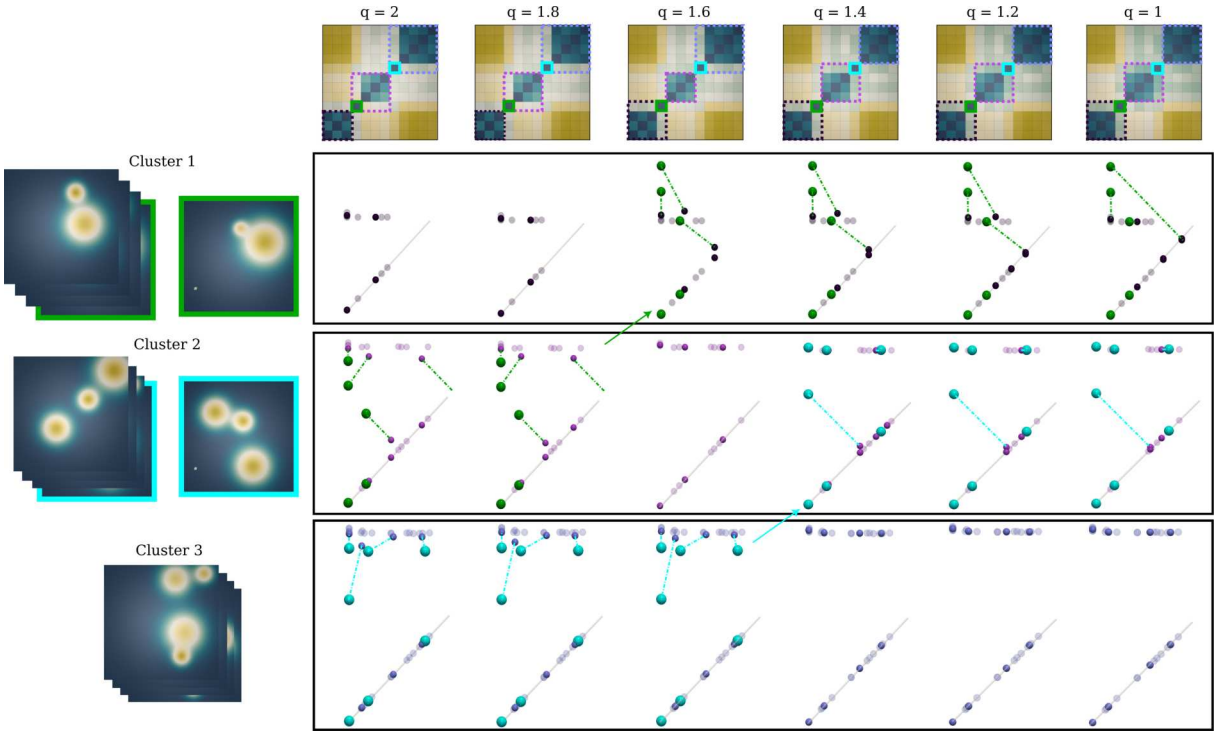


Figure 4.3: Comparison of clustering results on an ensemble of diagrams of Gaussian mixtures. On the left we have the 3 clusters: one cluster of 2 Gaussians (top), one cluster of 3 Gaussians (middle) and one cluster of 4 Gaussians (bottom). In the first and second clusters, we inserted an outlier (highlighted in green and cyan respectively) by setting one isolated pixel to an arbitrarily high value. Those pixels result in persistent pairs in the corresponding diagrams. On top we have the distance matrices of W_q for $q \in \{2, 1.8, 1.6, 1.4, 1.2, 1\}$. In the distance matrices, the clustering results are shown with dashed squares (clusters are colored in dark purple, purple and pale purple) while the outlier diagrams are indicated with a plain square (green and cyan). In the three frames; we visualize the evolution of each cluster and their barycenters for each q. Each frame corresponds to a cluster (top: cluster 1, middle: cluster 2, bottom: cluster 3). The outlier diagrams are colored in green and cyan. The barycenters are shown in opaque while the diagrams of each cluster are shown in transparent. We observe that for $q \in \{2, 1.8\}$ the green outlier is incorrectly assigned to the second cluster (as it exhibits the same number of persistence pairs, 3, as the entries of cluster 2). Similarly, given its number of persistence pairs, the cyan outlier is incorrectly assigned to the third cluster until $q = 1.6$. Beyond this value, the effect of the extra feature in these outlier diagrams is decreased, enabling their correct clustering.

4.4.1 Clustering on the persistence diagram metric space

The first natural application of our robust barycenters consists in using them for the problem of clustering an ensemble of persistence diagrams $X_1, \ldots, X_N$. In particular, clustering methods on ensembles of persistence diagrams group together subset of members that have similar topological structures, highlighting trends of topological features in the ensemble.

For this, we consider the classic clustering method, the k-means algorithm. This

is an iterative algorithm alternating between two phases: *computing k barycenters* and *labeling the elements into k clusters*. At first, k cluster barycenters B_j with $j \in \{1, \ldots, k\}$ are initialized as k diagrams in the initial input set $X_1, \ldots, X_N$, typically using the k-means++ method [42]. Then, the labeling phase consists in assigning each diagrams X_n to the closest barycenter B_j by using the W_q distance. After the labeling phase, the barycenters are updated by computing new barycenters based on the new k clusters using Alg. 3. The algorithm stops when reaching a maximum number of iterations or when converging (i.e., when the labels do not change anymore). However, when using W_2 as a distance, the presence of an outlier in an ensemble of persistence diagrams can incorrectly influence the barycenters in the k-means algorithm, and as a consequence the output labels of the clustering.

We leverage the robustness of the W_q barycenters for clustering problems when there are outliers in the ensemble to be clustered. We show a case in Fig. 4.3 where we artificially injected outlier pixels in an ensemble of synthetic scalar fields (a common degradation in real-life noisy data). This results in outlier diagrams in the input ensemble. We illustrate the clustering results with different $q \in \{2, 1.8, 1.6, 1.4, 1.2, 1\}$. This experiment shows that for q lower than 1.6, the resulting clustering put together the outliers into the correct groups, while for $q = 2$ the outliers are incorrectly clustered. This can also be seen in Fig. 4.4 where an outlier is naturally present in the ensemble of scalar fields. In this example, our generic barycenters enable the computation of the correct clustering, as off $q = 1.6$. Moreover, as illustrated on the right of Fig. 4.4 for the last cluster, our generic barycenters are more representative, visually, of the input diagrams as the importance of outlier persistence pairs is decreased in our framework.

4.4.2 Wasserstein dictionary encoding of persistence diagrams

Another application consists in using our robust barycenters as a core procedure for dictionary based encodings of ensembles of persistence diagrams [173]. Let $X_1, \ldots, X_N$ be an ensemble of persistence diagrams. A Wasserstein dictionary encoding aims at optimizing a set of persistence diagrams $\mathcal{D}^* = \{a_1^*, \ldots, a_m^*\}$ (called dictionary) and N vectors of barycentric coefficients $\Lambda^* = \{\boldsymbol{\lambda}_1^*, \ldots, \boldsymbol{\lambda}_N^*\}$ (i.e., N vectors of size m, with positive elements summing to 1) by solving:

$$\arg\min_{\mathcal{D},\Lambda} \sum_{\ell=1}^{N} W_2^2\big(B_2(\mathcal{D}, \boldsymbol{\lambda}_\ell), X_\ell\big), \tag{4.4}$$

where $B_2(\mathcal{D}, \boldsymbol{\lambda}_\ell)$ denotes a W_2 barycenter of $\mathcal{D}$ under barycentric coefficients $\boldsymbol{\lambda}_l$. Informally, this framework works as a lossy compression for persistence diagrams. The goal is to optimize a smaller set of persistence diagrams ($m \ll N$) and N vectors of barycentric weights such that the N Wasserstein barycenters defined by the barycentric weights are good approximations of the N input diagrams. This results in an encoding of much smaller size as only the dictionary and the N barycentric weights need to be stored to disk. This framework has two main applications: data reduction and dimensionality reduction. Naturally this framework can be extended to other Wasserstein distances. Then, Eq. 4.4 becomes:

$$\arg\min_{\mathcal{D},\Lambda} \sum_{l=1}^{N} W_q^q\big(B_q(\mathcal{D}, \boldsymbol{\lambda}_l), X_l\big), \tag{4.5}$$

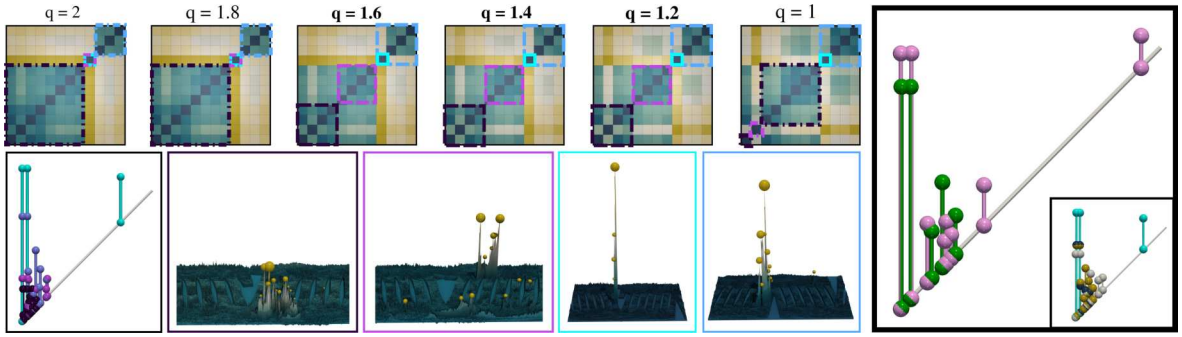


Figure 4.4: Visual comparison of distance matrices using W_q for $q \in \{2, 1.8, 1.6, 1.4, 1.2, 1\}$ on the *Volcanic Eruption* ensemble and the clustering results. Distance matrices are represented similarly to Fig. 4.3. This ensemble of 12 persistence diagrams has a natural outlier highlighted in cyan on the distance matrices. On the top, we can see that for $q \in \{2, 1.8\}$, the k-means algorithm keeps the outlier alone, groups the 8 first diagrams together and groups the last three together. Then starting from 1.6 to 1.2, the correct clusters are returned. But for $q = 1$, We observe that the clusters are not discriminated enough. On the bottom we have one representative scalar field for each cluster, and on the bottom left the corresponding diagrams, the cyan scalar field and diagram being the outlier. On the right, we have a visual comparison of two barycenters of the last cluster of four diagrams (represented on the bottom right of the square). The pink one encodes a W_2 barycenter, while the green one encodes a $W_{1.2}$ barycenter. We observe the influence of the outlier in the pink barycenter, as the two global pairs are higher than the green ones testifying the difference of scaling between the outlier (cyan) and the tree other diagrams in the cluster. Also, we notice the presence of an isolated pair above the diagonal that is generated by the isolated persistent pair in the outlier diagram.

where $B_q(\mathcal{D}, \boldsymbol{\lambda}_l)$ denotes a W_q barycenter returned by Alg. 3. For $q = 2$, in Chapt. 3 we introduced the analytic expression of the gradient $\nabla B_2(\mathcal{D}, \boldsymbol{\lambda}_l)$ with respect to $a_1, \ldots, a_m$ and $\boldsymbol{\lambda}_l$, enabling a simple gradient descent scheme for the optimization of Eq. 4.5. However, for $q \neq 2$, since ground barycenters are no longer obtained as arithmetic means, but by an interative, fixed-point method (Sec. 4.3.2), the gradient of the energy associated with Eq. 4.5 cannot be derived analytically. Instead, we rely on automatic differentiation (implemented in PyTorch) and use Adam [98] to optimize both the dictionary $\mathcal{D} = \{a_1, \ldots, a_m\}$ and the vectors of barycentric coefficients $\Lambda = \{\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_N\}$.

This extension results in a Wasserstein dictionary method that is more stable to the presence of outliers in the original input ensemble. Moreover, this extension lets us improve the initialization method for the dictionary. At first, the dictionary was chosen as the m elements that are the farthest to each other in the initial set [173]. But extending Eq. 4.4 to Eq. 4.5 allowed the dictionary to be initialized as barycenters issued from a k-means algorithm with $k = m$. From our experience, this initialization results in a better optimized energy (Eq. 4.5) and stability to the presence of outliers compared to the initial one [173]. We showcase this extension by applying a Wasserstein dictionary method using W_q. Fig. 4.5 shows a comparison of 2D planar layout of the barycenters, generated by the dictionaries and vectors optimized with Eq. 4.4 and Eq. 4.5, on the *Isabel* ensemble where we removed some entries to artificially inject an outlier (see the caption of Fig. 4.5 for more details). Specifically, this 2D planar layout is a direct application of the dictionary

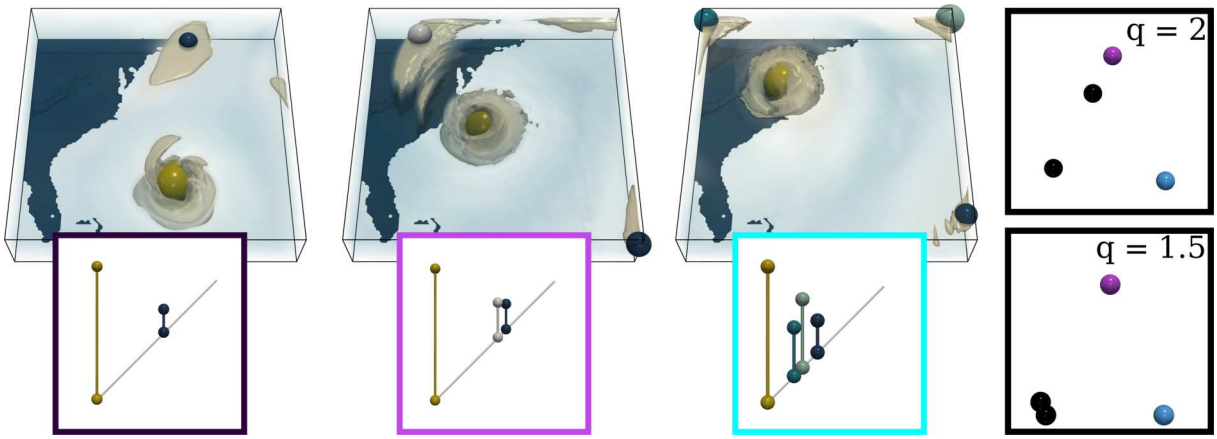


Figure 4.5: Visual comparison of 2D planar layouts (on the right) of Wasserstein barycenters after solving Eq. 4.4 and Eq. 4.5 when taking a dictionary with 3 diagrams. The initial ensemble *Isabel* is composed of 12 diagrams divided in 3 classes of 4 diagrams each. We removed 2 diagrams from the first and second classes, yielding an imbalanced ensemble (in terms of class size) of 8 diagrams. We show representative scalar field for each cluster, along with their diagram. The clusters are colored in dark purple, purple and cyan respectively. In the planar layouts, the points, representing the barycenters, are colored by their ground truth classification. For $q = 2$, the barycenter approximating an element of the first cluster (dark purple) is misplaced (i.e., located near the second cluster, purple). For $q = 1.5$, this same barycenter is correctly placed, thus yielding a planar projection that is more faithful to the ground-truth classification.

encoding when the dictionary has three atoms in it. After optimizing the dictionary, with the three Wasserstein distances between them, the cosine law can be used to form a triangle in $\mathbb{R}^2$. Then we use the vectors $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_m$ as vectors of barycentric coordinates in $\mathbb{R}^2$. This experiment shows that this extension is more stable to the presence of outliers, resulting in planar projections that are more faithful to the ground truth classification.

4.4.3 Computation time comparison

Table 4.1: Running times (in seconds) for computing a W_2 barycenter.

Method	$m = 4$	$m = 6$
Arithmetic Mean	1.9	24.9
$\mathfrak{b}_2$	6.3	29.0

In this section, we compare the time needed to compute a barycenter, with regard to the W_2 metric, using the arithmetic mean between points of $\mathbb{R}^2$, and the time when optimizing $\mathfrak{b}_2$. Our experiment consists in computing a barycenter of persistence diagrams for the Isabel ensemble (Sec. 4.2.1), where each diagram is thresholded by persistence to feature only $\sim$ 100 pairs and where T is set to 5. We report the resulting running times in Tab. 4.1. When considering $m = 4$ input diagrams, the computation based on the arithmetic mean is 3 times faster than the one based on $\mathfrak{b}_2$. However, for a larger

ensemble ($m = 6$), the difference is reduced to 4 seconds, yielding compatible run times between the two approaches.

4.5 Summary

In this chapter, we showcased the utility of a method for computing a robust Wasserstein barycenters of persistence diagrams. Specifically, we adapted a recent fixed-point method algorithm [181] to the case of persistence diagrams. We first gave a reminder on this fixed point framework to compute this robust barycenter. We also gave a formal proof of the necessary hypothesis for the convergence of this method. Then we presented two applications of this robust barycenter to clustering and dictionary encoding of persistence diagrams in the presence of outliers. We believe that our work on the robustness of Wasserstein barycenters is a useful step toward improving their applicability in the analysis of real-life ensembles of persistence diagrams.

Chapter 5

A User's Guide to Sampling Strategies for Sliced Optimal Transport

---

This chapter serves as a user's guide to sampling strategies for sliced optimal transport [28,159]. We provide reminders and additional regularity results on the Sliced Wasserstein distance. We detail the construction methods, generation time complexity, theoretical guarantees, and conditions for each strategy. Additionally, we provide insights into their suitability for sliced optimal transport in theory. Extensive experiments on both simulated and real-world data offer a representative comparison of the strategies, culminating in practical recommendations for their best usage.

The work presented in this chapter has been published in the journal Transaction on Machine Learning Research 2025 [174]. The python code used can be found at `https://github.com/Keanu-Sisouk/SW-Sampling-Guide`.

5.1 Context

The computational demands of the Wasserstein distance are, quite high, since evaluating the distance between two discrete distributions of N samples with traditional linear programming methods incurs a runtime complexity of $\mathcal{O}(N^3 \log N)$ [142]. This computational burden has motivated the development of alternative metrics sharing some of the desirable properties of the Wasserstein distance but with reduced complexity.

The Sliced Wasserstein (SW) distance [28,159], defined by slicing the Wasserstein distance along all possible directions on the hypersphere, is one of these efficient alternatives. Indeed, the SW distance maintains the core properties of the Wasserstein distance but with reduced computational overhead. For compactly supported measures, Bonnotte [29] showed for instance that the two distances are equivalent. Again, it has been successfully applied in various domains, such as domain adaptation [108], texture synthesis and style transfer [61,89], generative modeling [53,196], regularizing autoencoders [100], shape matching [106], and has even been adapted on Riemaniann manifolds [26].

The SW distance between two measures μ and ν can be written as the expectation of the one dimensional Wasserstein distance between the projections of μ and ν on a line whose direction is drawn uniformly on the hypersphere. It benefits from the simplicity of the Wasserstein distance computation in one dimension. In practice, computing the expectation on the hypersphere is unfeasible, so it is estimated thanks to numerical integration. The most common method for approximating the SW distance is to rely on Monte Carlo approximation, by sampling M random directions uniformly on the hypersphere and approximating the integral by an average on these directions. Since the Wasserstein distance in 1D between two measures of N samples can be computed in $\mathcal{O}(N \log N)$, computing this empirical version of Sliced Wasserstein has a runtime complexity of $\mathcal{O}(MN \log N)$. This complexity makes it a compelling alternative to the Wasserstein distance, especially when the number N of samples is high.

As a Monte Carlo approximation, the law of large numbers ensures that this empirical Sliced Wasserstein distance converges to the true expectation, with a convergence rate of $\mathcal{O}(\frac{1}{\sqrt{M}})$. This convergence speed is slow but independent of the space dimension. However, it is important to keep in mind that to preserve some of the properties of the SW distance, the number M of directions should increase with the dimension. For instance, it has been shown that for the empirical distance to almost surely separate discrete distributions (in the sense that if the distance between two distributions is zero then the two distributions are almost surely equal), the number of directions M must be chosen strictly larger than the space dimension [182].

Classical Monte Carlo with independent samples is not always optimal, since independent random samples do not cover the space efficiently, creating clusters of points and leaving holes between these clusters. In very low dimension, quadrature rules provide efficient alternative methods to classical Monte Carlo. On the circle for instance, the simplest solution is to replace the M random samples by the roots of unity $\{e^{i\frac{2k\pi}{M}} \mid 0 \leq k \leq M-1\}$: since the function that we wish to integrate is Lipschitz, this ensures that the integral approximation converges at speed $\mathcal{O}(\frac{1}{M})$. However, such quadrature rules are unsuitable for high-dimensional problems, as they require an exponential number of samples to achieve a given level of accuracy.

Another alternative sampling strategy is to rely on quasi-Monte Carlo (Q.M.C.) meth-

ods, which use deterministic, low-discrepancy sequences instead of independent random samples. Traditional Q.M.C. methods are designed for integration over the unit hypercube $[0,1]^d$. The quality of a Q.M.C. sequence is often measured by its discrepancy, which measures how uniformly the points cover the space. A lower discrepancy correlates with a better approximation, according to the Koksma-Hlawka inequality [30]. Examples of low-discrepancy sequences for the unit cube include for instance the Halton sequence [83], and the Sobol sequence [175], and different approaches have been investigated to project such sequences on the hypersphere. While quadrature rules are recommended for very small dimensions ($d = 1$ or 2 for instance), Q.M.C. integration is particularly effective in low to intermediate dimensions. A variant of low-discrepancy sequence is one where randomness is injected in the sequence while preserving its "low discrepancy" property. Such a sequence is called a randomized low-discrepancy sequence, and this is the foundation to randomized quasi-Monte Carlo (R.Q.M.C.) methods [137]. Q.M.C. methods do not only rely on low-discrepancy sequences, but can also use point sets of a given size directly optimized to have low-discrepancy, such as s-Riesz point configurations on the sphere [82]. However Q.M.C. and R.Q.M.C. methods on the sphere have a strong practical downside: they suffer from the curse of dimensionality. Indeed the higher the dimension the harder it is to generate samples with Q.M.C. and R.Q.M.C. approaches. Moreover, the higher the dimension, the slower the convergence rate, and the more regular the integrand needs to be to ensure fast convergence. The recent paper [127] already proposes an interesting comparison of such Q.M.C. methods to approximate the Sliced-Wasserstein distance in dimension 3, showing that such methods could provide better approximations that conventional M.C. in this specific dimensional setting.

All the sampling strategies mentioned above are designed to provide a good coverage of the space. However, they do not take into account the specific structure of the integrand, which is the Wasserstein distance between the one dimensional projections of the two measures μ and ν. More involved methods to improve Monte Carlo efficiency include importance sampling, control variates or stratification [13]. Such variance reduction techniques strategies can also be used in conjunction with quasi-Monte Carlo integration. Control variates have been explored for Sliced Wasserstein approximation in [128] and [109], showing interesting improvements in intermediate dimensions over classical Monte Carlo.

The goal of this survey is to provide a detailed comparison of these different sampling strategies for the computation of Sliced-Wasserstein in various dimensional settings. It is intended as a user-guide to help practitioners choose the appropriate sampling strategy for their specific problem, depending on the size and dimension of their data, and the type of experiments to be carried out (whether or not they need to compute numerous SW distances for instance). We will also look at the particularities of the different approaches, some being more appropriate than others depending on whether a given level of accuracy is desired (in which case an approach allowing sequential sampling is preferable to one requiring optimization of a point set) or, on the contrary, a given computation time is imposed. We will mainly focus on sampling strategies which are independent of the knowledge of the measures μ and ν, such as uniform random sampling [13], orthonormal sampling [162], low-discrepancy sequences mapped on the sphere [83, 175], randomized low-discrepancy sequences mapped on the spheres [137], Fibonacci point sets [85] and Riesz configuration point sets [82].

For the sake of completeness, we also include in our comparison the recent approach [109], which appears to be the most efficient among recent control variates approaches proposed to approximate Sliced Wasserstein.

The chapter is organized as follows. Sec. 5.2 introduces some reminders on the Sliced Wasserstein distance such as its definition and some regularity properties. Sec. 5.3 explores all the sampling methods considered in this work, hightlighting their theoretical guarantees, the conditions under which they can be used, and identifying which methods suffer from the curse of dimensionality. Sec. 5.4 provides a comparison of each sampling method's experimental results on different datasets. Then in Sec. 5.5, we detail how we can use the Sliced Wasserstein distance for our dictionary framework. Finally, in Sec. 5.6 we offer detailed recommendations for choosing and using these sampling methods effectively in practice.

5.2 Reminders on the Sliced Wasserstein Distance

5.2.1 Definition

In the following, we write $\langle \cdot \mid \cdot \rangle$ the Euclidean inner product in $\mathbb{R}^d$, $\|\cdot\|$ the induced norm, $\mathbb{S}^{d-1} = \{x \in \mathbb{R}^d \mid \|x\| = 1\}$ the unit sphere of $\mathbb{R}^d$. For $\theta \in \mathbb{S}^{d-1}$, we write $\pi_\theta : \mathbb{R}^d \to \mathbb{R}$ the map $x \mapsto \langle \theta | x \rangle$, s_{d-1} the uniform measure over $\mathbb{S}^{d-1}$. We also denote $\#$ the push-forward operation [1].

For two probability measures μ and ν supported in $\mathbb{R}^d$ and with finite moments of order 2, the Sliced Wasserstein Distance between μ and ν is defined as

$$SW_2^2(\mu, \nu) = \mathbb{E}_{\theta \sim \mathcal{U}(\mathbb{S}^{d-1})}[W_2^2(\pi_\theta \# \mu, \pi_\theta \# \nu)] = \int_{\mathbb{S}^{d-1}} W_2^2(\pi_\theta \# \mu, \pi_\theta \# \nu) ds_{d-1}(\theta). \tag{5.1}$$

This distance, introduced in [159], has been thoroughly studied and used as a dissimilarity measure between probability distributions in machine learning [28, 100, 125], and more generally as an alternative to the Wasserstein distance. Its simplicity stems from the fact that the Wasserstein distance between two probability measures in one dimension has an explicit formula. Recalling Eq. 2.8, for two probability measures ρ_1 and ρ_2 on the line, the Wasserstein distance $W_2(\rho_1, \rho_2)$ can be written

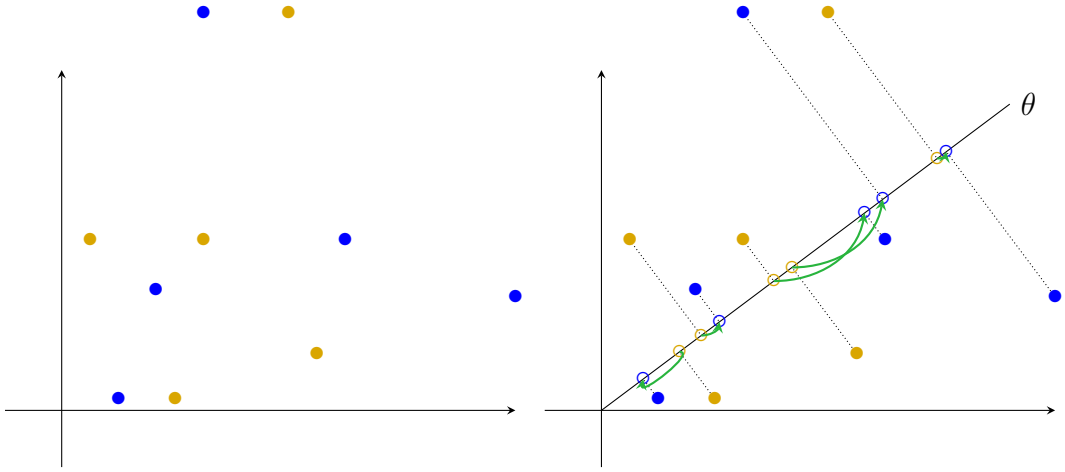
$$W_2^2(\rho_1, \rho_2) = \int_0^1 |F_1^{-1}(t) - F_2^{-1}(t)|^2 dt, \tag{5.2}$$

where F_1 and F_2 are the cumulative distribution functions of ρ_1 and ρ_2, and F_1^{-1} and F_2^{-1} are their respective generalized inverses (see [164] Proposition 2.17). For two one dimensional discrete measures $\rho_1 = \frac{1}{K}\sum_{k=1}^K \delta_{x_k}$ and $\rho_2 = \frac{1}{K}\sum_{k=1}^K \delta_{y_k}$, this distance becomes

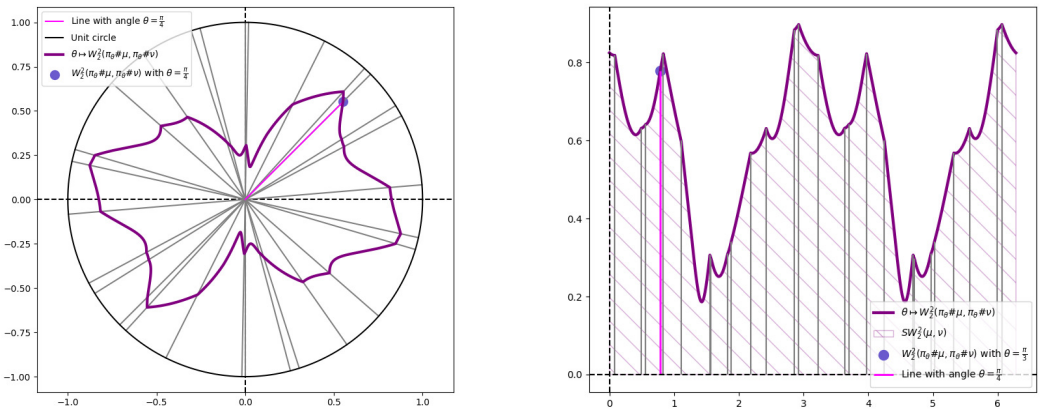
$$W_2^2(\rho_1, \rho_2) = \frac{1}{K}\sum_{k=1}^K |x_{\sigma(k)} - y_{\tau(k)}|^2, \tag{5.3}$$

where σ and τ are permutations of $[\![1, K]\!]$ which respectively order the sets $\{x_1, \dots, x_K\}$ and $\{y_1, \dots, y_K\}$ on the line.

[1] The push-forward of a measure μ on $\mathbb{R}^d$ by an application $T : \mathbb{R}^d \to \mathbb{R}^k$ is defined as a measure $T\#\mu$ on $\mathbb{R}^k$ such that for all Borel sets $B \in \mathcal{B}(\mathbb{R}^k), T\#\mu(B) = \mu(T^{-1}(B))$.



(a) On the left, we can see the two discrete distributions μ (blue points) and ν (yellow points). On the right, we have their projections $\pi_\theta\#\mu$ (blue circles) and $\pi_\theta\#\nu$ (yellow circles) along the direction θ. One then takes the increasing ordering of $\pi_\theta\#\mu$ and $\pi_\theta\#\nu$, to obtain the corresponding matchings (green arrows) and computes the cost following Eq. 5.3.



(b) On the left, we have a plot of $\theta \mapsto W_2^2(\pi_\theta\#\mu, \pi_\theta\#\nu)$ in polar coordinates, with the distributions μ and ν from Fig. 5.1a (top). The grey lines represent the angles where $\theta \mapsto W_2^2(\pi_\theta\#\mu, \pi_\theta\#\nu)$ is not differentiable, the magenta line is the line of angle $\theta = \frac{\pi}{4}$ and the blue dot is a specific value of $W_2^2(\pi_\theta\#\mu, \pi_\theta\#\nu)$ with the same angle. On the right, we have a $1D$ plot of $\theta \mapsto W_2^2(\pi_\theta\#\mu, \pi_\theta\#\nu)$, here the hashed area represents $SW_2^2(\mu, \nu)$ and again the vertical grey lines represent the values where $\theta \mapsto W_2^2(\pi_\theta\#\mu, \pi_\theta\#\nu)$ is not differentiable.

Figure 5.1: On the first row, Fig. 5.1a illustrates the computation of $W_2^2(\pi_\theta\#\mu, \pi_\theta\#\nu)$ for a fixed θ. On the second row, Fig. 5.1b gives a geometrical illustration of $SW_2^2(\mu, \nu)$ with μ, ν taken as in Fig. 5.1a.

As a consequence, the Sliced Wasserstein distance between two discrete probability measures $\mu = \frac{1}{K}\sum_{k=1}^{K} \delta_{x_k}$ and $\nu = \frac{1}{K}\sum_{k=1}^{K} \delta_{y_k}$ on $\mathbb{R}^d$ (i.e. with $(x_k)_{k=1,\dots,K}, (y_k)_{k=1,\dots,K} \in \mathbb{R}^d$)

can be rewritten:

$$SW_2^2(\mu,\nu) = \frac{1}{K}\sum_{k=1}^{K}\int_{\mathbb{S}^{d-1}}(\langle x_{\sigma_\theta(k)} - y_{\tau_\theta(k)},\theta\rangle)^2 ds_{d-1}(\theta) \tag{5.4}$$

$$= \frac{1}{K}\sum_{k=1}^{K}\int_{\mathbb{S}^{d-1}}(\langle x_k - y_{\tau_\theta\circ\sigma_\theta^{-1}(k)},\theta\rangle)^2 ds_{d-1}(\theta), \tag{5.5}$$

where σ_θ and τ_θ denotes respectively permutations which order the one dimensional point sets $(\langle x_k,\theta\rangle)_{k=1,\dots,N}$ and $(\langle y_k,\theta\rangle)_{k=1,\dots,N}$. Fig. 5.1 illustrates the computation of $W_2^2(\pi_\theta\#\mu,\pi_\theta\#\nu)$ for two discrete measures in two dimensions (Fig. 5.1a), and shows how this quantity varies when θ spans $[0,2\pi]$ (Fig. 5.1b).
Since the permutations σ_θ and τ_θ depends on the direction θ, the integrals in Eq. 5.1 and Eq. 5.4 do not have closed forms. For this reason, practitioners rely on Monte Carlo approximations of the form:

$$\frac{1}{KM}\sum_{k=1}^{K}\sum_{j=1}^{M}(\langle x_{\sigma_{\theta_j}(k)} - y_{\tau_{\theta_j}(k)},\theta_j\rangle)^2, \tag{5.6}$$

where $\theta_1,\dots,\theta_M$ are i.i.d. and follow a uniform distribution on the sphere. Classically, the convergence rate of such Monte Carlo estimations of SW is $\mathcal{O}(\frac{1}{\sqrt{M}})$ [84]. In this context, it is natural to question the optimality of sampling methods to approximate SW efficiently in different scenarios.

5.2.2 Regularity results on $\theta \mapsto W_2^2(\pi_\theta\#\mu,\pi_\theta\#\nu)$

The efficiency of sampling strategies used in numerical integration is highly dependent on the regularity of the functions to be integrated. For this reason, in the following we give some properties of the function (Fig. 5.1b):

$$f : \theta \mapsto W_2^2(\pi_\theta\#\mu,\pi_\theta\#\nu) \tag{5.7}$$

on the hypersphere $\mathbb{S}^{d-1}$. We first look at classical regularity properties of f.

Proposition 5.1 : f is Lipschitz on $\mathbb{S}^{d-1}$.

Proof. Let μ and ν be two probability measures with finite moments of order 2, and $\theta_1,\theta_2 \in \mathbb{S}^{d-1}$. The triangular inequality on W_2 yields

$$|W_2(\pi_{\theta_1}\#\mu,\pi_{\theta_1}\#\nu) - W_{,2}(\pi_{\theta_2}\#\mu,\pi_{\theta_2}\#\nu)| \leq W_2(\pi_{\theta_1}\#\mu,\pi_{\theta_2}\#\mu) + W_2(\pi_{\theta_1}\#\nu,\pi_{\theta_2}\#\nu).$$

Using Eq. 2.10, we also have

$$\begin{aligned} W_2^2(\pi_{\theta_1}\#\mu,\pi_{\theta_2}\#\mu) &= \inf_{X\sim\mu,Y\sim\mu}\mathbb{E}\left[|\langle\theta_1,X\rangle - \langle\theta_2,Y\rangle|^2\right] \\ &\leq \inf_{X\sim\mu}\mathbb{E}\left[|\langle\theta_1-\theta_2,X\rangle|^2\right] \leq \|\theta_1-\theta_2\|^2\mathbb{E}_{X\sim\mu}[\|X\|^2]. \end{aligned}$$

We can show similarly that $W_2^2(\pi_{\theta_1}\#\nu,\pi_{\theta_2}\#\nu) \leq \|\theta_1-\theta_2\|^2\mathbb{E}_{Y\sim\nu}[\|Y\|^2]$. Thus

$$|W_2(\pi_{\theta_1}\#\mu, \pi_{\theta_1}\#\nu) - W_2(\pi_{\theta_2}\#\mu, \pi_{\theta_2}\#\nu)| \leq \|\theta_1 - \theta_2\| \left(\sqrt{\mathbb{E}_{X\sim\mu}[\|X\|^2]} + \sqrt{\mathbb{E}_{Y\sim\nu}[\|Y\|^2]} \right).$$

□

Since f is Lipschitz continuous, it is differentiable almost everywhere. However the previous result does not give us the set where f is non differentiable. In the following we give a more complete proof when μ and ν are discrete following the notations introduced in Sec. 5.2.1.

Proposition 5.2 : When μ and ν are finite discrete measures, f is piecewise $\mathcal{C}^\infty$ ($\mathcal{C}^\infty_{pw}$) and Lipschitz on $\mathbb{S}^{d-1}$.

Proof. For discrete measures $\mu = \frac{1}{K}\sum_{k=1}^{K} \delta_{x_k}$ and $\nu = \frac{1}{K}\sum_{k=1}^{K} \delta_{y_k}$ on $\mathbb{R}^d$, f can be rewritten as

$$f(\theta) = \min_{\sigma\in\Sigma_K} f_\sigma(\theta), \text{ where } f_\sigma(\theta) = \sum_{k=1}^{K} \langle x_k - y_{\sigma(k)}|\theta\rangle^2, \tag{5.8}$$

where Σ_K is the set of permutations of $[\![1, K]\!]$. We assume that the $\{x_i\}$ (resp. $\{y_j\}$) are all distinct. In the following, we study the regularity of f as a function of $\mathbb{R}^d$ and deduce the regularity properties of its restriction $f_{|\mathbb{S}^{d-1}}$. Observe that each f_σ defines a quadratic function on $\mathbb{R}^d$ and f, as a minimum of a finite number of such functions, is continuous and also piecewise $\mathcal{C}^\infty$ on $\mathbb{R}^d$. Since f is continuous on $\mathbb{R}^d$, its restriction to $\mathbb{S}^{d-1}$ is also continuous. To show that this restriction to $\mathbb{S}^{d-1}$ is also in $\mathcal{C}^\infty_{pw}$, it is enough to observe that the set of points of $\mathbb{R}^d$ where f is not differentiable is included in the finite union of hyperplanes $\left(\cup_{i,j}\text{Span}(x_i - x_j)^\perp\right) \bigcup \left(\cup_{k,l}\text{Span}(y_k - y_l)^\perp\right)$, since these hyperplanes are the locations where the minimum in Eq. 5.8 jumps from a permutation σ to another one (see Fig. 5.2 as an illustration of those hyperplanes). Each of these hyperplanes intersect $\mathbb{S}^{d-1}$ on a great circle, and we call $\mathcal{U}$ the sphere minus this finite union of great circles. The open set $\mathcal{U}$ (which is dense in $\mathbb{S}^{d-1}$) can be written as the union $\bigcup_{k=1}^{p} V_k$ of a finite number of connected open sets V_l, such that on each V_l, the permutation σ which attains the minimum in Eq. 5.8 is constant and unambiguous. We write this permutation σ_l. On each V_l, $f_{|\mathbb{S}^{d-1}} = f_{\sigma_l}$, thus is $\mathcal{C}^\infty$ on V_l and its derivative can be obtained as the projection of ∇f_{σ_l} on the hypersphere. For $\theta \in \mathcal{U}$, writing σ_θ the permutation which attains the minimum in Eq. 5.8 for the direction θ, this derivative can be written

$$\nabla_{(d-1)} f(\theta) = 2\left(\sum_{k=1}^{K} \left(\langle x_k - y_{\sigma_\theta(k)}|\theta\rangle (x_k - y_{\sigma_\theta(k)}) - \langle x_k - y_{\sigma_\theta(k)}|\theta\rangle^2 \theta \right) \right). \tag{5.9}$$

Since these derivatives are upper bounded on the compact set $\mathbb{S}^{d-1}$, it follows that f is also Lipschitz on $\mathbb{S}^{d-1}$.
In the case where several x_i (or y_j) are equal, several of the functions f_σ coincide. For instance, if $x_1 = x_2$, the values of $\sigma(1)$ and $\sigma(2)$ can be exchanged without modifying f_σ. By eliminating all the redundant functions, we can make the same reasoning as before to show the same regularity results on f. In this case, all the pairs (x_i, x_j) with $x_i = x_j$ should be removed when constructing the set of great circles dividing the hypersphere. □

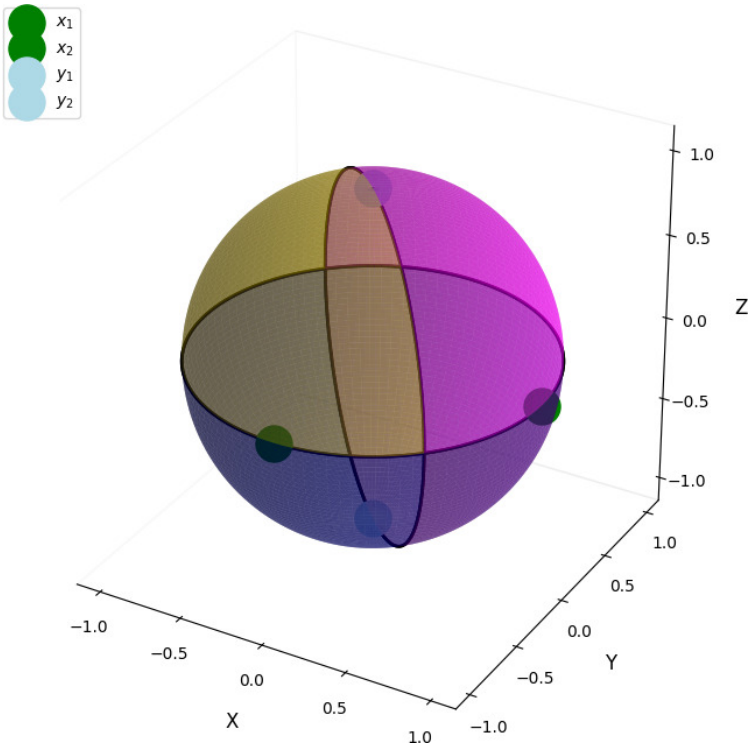


Figure 5.2: Illustration of the open subsets $\bigcup_{k=1}^{p} V_k$ and their intersection with the hyperplanes $\left(\cup_{i,j}\text{Span}(x_i - x_j)^\perp\right) \bigcup \left(\cup_{k,l}\text{Span}(y_k - y_l)^\perp\right)$, in the specific case of two measures made of two diracs $\mu = \frac{1}{2}\sum_{i=1}^{2} \delta_{x_i}$ with $x_1, x_2 = (1,0,0)^T, (0,-1,0)^T$ and $\nu = \frac{1}{2}\sum_{i=1}^{2} \delta_{y_i}$ with $y_1, y_2 = (0,0,1)^T, (0,0,-1)^T$. The hyperplanes divide the sphere into the colored sections where σ_θ and τ_θ are constant.

The following proposition will also be useful in the next sections.

Proposition 5.3 : $f \in H^1(\mathbb{S}^{d-1})$, where, for $\alpha \in \mathbb{N}$, the Sobolev space $H^\alpha(\mathbb{S}^{d-1})$ is defined as [87]

$$H^\alpha(\mathbb{S}^{d-1}) = \{h \in L^2(\mathbb{S}^{d-1}) \mid \partial^{|j|} h \in L^2(\mathbb{S}^{d-1}), 0 \leq |j| \leq \alpha\},$$

with j a multi-index and $\partial^{|j|}$ the partial mixed derivative of order $|j|$ on $\mathbb{S}^{d-1}$.

Proof. We have seen previously that f is continuous and piecewise $\mathcal{C}^\infty$, piecewise quadratic to be more precise. Thus its weak derivative is piecewise linear with discontinuities on a finite union of hyperplanes, which is L^2. □

5.3 Reminders on sampling strategies on the sphere and their theoretical guarantees

In this section, we present the different sampling methods for numerical integration on $\mathbb{S}^{d-1}$ considered in this work, before comparing them experimentally in Sec. 5.4. This part

addresses three main types of sampling: random sampling, discrepancy-based sampling, and a control variate approach. The first type includes the classical Monte Carlo (M.C.) method ([84], [110]) on the sphere and its variant called orthonormal sampling [162]. The second one relies on a concept called the discrepancy ([110], [54]) of a point set, which represents the number of points in a unit of volume, and can be divided into two categories: low-discrepancy sequences (or digital nets) and point sets (or lattices). Among the former category, we also investigate a method based on a spherical sliced-Wasserstein type discrepancy [25]. The last type details a control variates method [110] using spherical harmonics [122] for this purpose [109]. We will also determine which method, and under which conditions, is theoretically suitable based on the regularity properties established in Sec. 5.2.2. Tab. 5.1 presents a taxonomy of all the sampling methods explored in this work. It details which method's convergence rate result is **independent from the dimension** (i.e. the dimension does not appear in the asymptotic rate), which one can be **computed independently** (i.e. each sample can be generated independently from the others), and which one can be **computed and stored** in advance.

Sampling types	Dimension independence	Independent computation	Possible pre-computation
Random Sampling			
Uniform Sampling	x	x	x
Orthonormal Sampling	x	x	x
Based on discrepancy			
Riesz Point Set / Riesz Point Set Randomized			x
Fibonacci Point Set / Fibonacci Point Set Randomized			x
Sobol / Sobol Randomized mapped on $\mathbb{S}^{d-1}$		x	x
Halton / Halton Randomized on $\mathbb{S}^{d-1}$		x	x
Spherical Sliced Wasserstein Discrepancy	x		x
Control variates			
Spherical Harmonics Control Variates			

Table 5.1: Taxonomy of the three types of sampling methods investigated in this chapter.

Tab. 5.2 gives a summary of the convergence rate and computational complexity of each sampling method explored in this chapter. In this table $n_M = o\left(M^{1/\left(2(d-1)\right)}\right)$.

Sampling types	Theoretical convergence rate	Time complexity	Space complexity
Random Sampling			
Uniform Sampling	$\mathcal{O}(1/\sqrt{M})$	$\mathcal{O}(M)$	$\mathcal{O}(M)$
Orthonormal Sampling	None	$\mathcal{O}(M)$	$\mathcal{O}(M)$
Based on discrepancy			
Riesz Point Set / Riesz Point Set Randomized	$1/M$ on $\mathbb{S}^1$, Not applicable otherwise	$\mathcal{O}(M^2)$	$\mathcal{O}(M)$
Fibonacci Point Set / Fibonacci Point Set Randomized	Not applicable	$\mathcal{O}(M)$	$\mathcal{O}(M)$
Sobol / Sobol Randomized mapped on $\mathbb{S}^{d-1}$	None	$\mathcal{O}\left(M\log_b^2(M)\right)$	$\mathcal{O}(M)$
Halton / Halton Randomized on $\mathbb{S}^{d-1}$	None	$\mathcal{O}\left(M\log_b^2(M)\right)$	$\mathcal{O}(M)$
Spherical Sliced Wasserstein Discrepancy	None	$\mathcal{O}(M\log(M))$	$\mathcal{O}(M)$
Control variates			
Spherical Harmonics Control Variates	$\mathcal{O}\left(1/(n_M\sqrt{M})\right)$	$\mathcal{O}(M)$	$\mathcal{O}(M)$

Table 5.2: Convergence rate, time complexity and spacial complexity (w.r.t the sampling number) summary of the sampling methods studied in this chapter.

5.3.1 Random samplings

We first explore classical strategies for randomly generating points on the sphere: uniform sampling [84] and orthonormal sampling [162]. These strategies are the most commonly

used for estimating SW_2^2, and their convergence rates do not depend on the dimension of the input measures.

5.3.1.1 Classical Monte Carlo

The classical Monte Carlo method uses uniform random sampling to generate the projection angles. For $(\theta_M)_{M\in\mathbb{N}^*}$ i.i.d. samples of s_{d-1} [2], we write the Monte Carlo Estimator

$$X_M := \frac{1}{M}\sum_{i=1}^{M} f(\theta_i) \text{ with } M \in \mathbb{N}^*. \tag{5.10}$$

The law of large numbers ensures that X_M converges a.s. to $SW_2^2(\mu,\nu) = \mathbb{E}_{\theta\sim s_{d-1}}[f(\theta)]$ as M goes to infinity. Moreover, the rate of convergence for this unbiased estimator is given by

$$\sqrt{\mathbb{V}[X_M]} = \sqrt{\frac{\mathbb{V}[X_1]}{M}} = \frac{\sigma}{\sqrt{M}}, \tag{5.11}$$

where $\sigma^2 = \mathbb{V}[f(\theta)] = \int_{\mathbb{S}^{d-1}} f^2(\theta) ds_{d-1}(\theta) - SW_2^4(\mu,\nu) < +\infty$. This convergence rate in Eq. 5.11 does not depend on the dimension of the input measures. In order to derive confidence intervals for $SW_2^2(\mu,\nu)$, we can rely on the Central Limit Theorem [68] , which states that

$$\sqrt{M}\frac{[X_M - SW_2^2(\mu,\nu)]}{\sigma} \xrightarrow[M\to+\infty]{\mathcal{L}} \mathcal{N}(0,1),$$

This allows us to compute confidence intervals for $SW_2^2(\mu,\nu)$ by using the quantiles of the standard normal distribution. This means that for M large enough,

$$\mathbb{P}\left(X_M - SW_2^2(\mu,\nu) \in \left[-\frac{\sigma q_{1-\alpha/2}}{\sqrt{M}}, \frac{\sigma q_{1-\alpha/2}}{\sqrt{M}}\right]\right) \xrightarrow[M\to+\infty]{} 1-\alpha,$$

with α in $[0,1]$ and $q_{1-\alpha/2}$ the quantile of level $1-\alpha/2$ of $\mathcal{N}(0,1)$. One strategy for choosing M is taking M such that $\frac{\sigma q_{1-\alpha/2}}{\sqrt{M}} \leq \varepsilon$ with $\varepsilon \geq 0$ a chosen precision. The value of σ being unknown, a possibility is to plug a consistent estimator of σ^2, such as

$$\hat{\sigma}_M^2 = \frac{1}{M}\left[\sum_{i=1}^{M} f(\theta_i)^2 - X_M^2\right].$$

[198] provides an alternative criteria for choosing M, however it is quite impractical as it requires to compute the Wasserstein distance between μ and ν.

5.3.1.2 Orthonormal sampling

A variant of the uniform sampling covered in Sec. 5.3.1.1 was introduced by [162], which presents a simple variant for the previous Monte Carlo estimator X_M by sampling random orthonormal bases. This method is inspired by variance reduction techniques known as

[2] In practice, to simulate a random variable $\theta \sim s_{d-1}$, one takes a normal random variable $Z \sim \mathcal{N}(0, I_d) \neq 0$ and chooses $\theta = \frac{Z}{\|Z\|} \sim s_{d-1}$ [13].

stratification [110]. Let $O(d)$ be the orthogonal group in $\mathbb{R}^d$. For $(\Theta_P)_{P\in\mathbb{N}^*} \sim \mathcal{U}\big(O(d)\big)$, denoting $\theta_1, \dots, \theta_M$ all the columns of the matrices $\Theta_1, \dots, \Theta_K$, we define $Y_M = \frac{1}{M}\sum_{i=1}^{M} f(\theta_i)$. It is easy to show that each θ_i follows the uniform distribution on $\mathbb{S}^{d-1}$ [162]. As a consequence, the estimator Y_M is still unbiased. Although it is not possible to show that Y_M has a smaller variance than X_M in general, this estimator is most of the time more efficient than X_M in our experiments and shows an equivalent or better rate of convergence in practice. This might be due to the fact that the diversity of the samples is increased by the orthonormality constraint.

Remark 5.1 : Other fully random point processes on $[0,1]^2$ or $\mathbb{S}^2$ suitable for Monte Carlo integration are studied in the literature. Among them, we can mention Determinantal Point Process (DPP). Recent works, such as [64], have proposed DPP methods directly on the sphere $\mathbb{S}^2$. Unfortunately, due to the lack of publicly available implementations, we could not experiment efficiently with these methods.

5.3.2 Sampling strategies based on discrepancy

We examine in this section two different types of deterministic sampling based on discrepancy: low-discrepancy sequences (digital nets) and low-discrepancy point sets (lattices). They were developed to replace random sampling, expecting to have a better convergence rate than the classical Monte Carlo method.

5.3.2.1 Low-discrepancy sequences

Quasi-random sequences, better known as low-discrepancy sequences (L.D.S.), are sequences mimicking the behavior of random sequences while being entirely deterministic. To date, these sequences are only defined on the unit hypercube $[0,1]^d$. We introduce below a first definition of discrepancy ([110], [54]).

Definition 5.1 : The discrepancy of a set of points $P = \{u_1, \dots, u_M\}$ in $[0,1]^d$ is defined as

$$D_M(P) = \sup_{I\in\mathcal{I}} \left| \frac{|P\cap I|}{M} - \lambda^{\otimes d}(I) \right|,$$

where $|A|$ denotes the cardinal of a set A, $\lambda^{\otimes d}$ is the d-dimensional Lebesgue measure and $\mathcal{I} = \{\prod_{i=1}^{d} [a_i, b_i[\mid 0 \le a_i < b_i \le 1\}$. The star-discrepancy $D_M^*(P)$ is defined the same way with $\mathcal{I}^* = \{\prod_{i=1}^{d} [0, b_i[\mid 0 \le b_i \le 1\}$.

We can now provide a definition of a low-discrepancy sequence (L.D.S.).

Definition 5.2 : Let $(u_m)_{m\in\mathbb{N}^*}$ be a sequence in $[0,1]^d$. Denoting $P_M = \{u_1, \dots, u_M\}$ for any $M \in \mathbb{N}^*$, u is a L.D.S. if

$$D_M^*(P_M) \xrightarrow[M\to+\infty]{} 0.$$

Intuitively, a sequence is considered as a L.D.S. if the portion of points in the sequence falling into an area I is closed to the measure of I.

The notion of discrepancy is important because it is related to the error made when approximating an integral on the hypercube by its Monte Carlo approximation. This relation is made explicit by the Koksma-Hlawka inequality ([110]; [54]; [30]).

This inequality requires to introduce the notion of Hardy-Krause variation V_h of a function h on $[0,1]^d$ [8], which is out of the scope of this work, but can be broadly understood as a measure of the oscillation of h on the unit cube $[0,1]^d$.

Proposition 5.4 (Koksma-Hlawka inequality) **:** Let $h : [0,1]^d \to \mathbb{R}$ have bounded variation V_h on $[0,1]^d$ in the sense of Hardy-Krause [8]. Then for $\{u_1, \dots, u_M\}$ a point set in $[0,1]^d$, we have

$$\left| \frac{1}{M} \sum_{k=1}^{M} h(u_k) - \int_S h(x) d\lambda^{\otimes d}(x) \right| \leq V_h D_M^*(u_1, \dots, u_M). \tag{5.12}$$

The proof of this inequality and basic results on discrepancy theory can be found in [103] and [54]. Eq. 5.12 shows that the absolute error made by the Monte Carlo approximation is upper bounded by a term depending only on h and the star discrepancy. Compared to the Central Limit Theoreom, this inequality is not probabilistic and not asymptotic, the bound being valid for every $M \in \mathbb{N}^*$. An important limitation is the term V_h, which is impractical to compute directly. When $d = 1$, this term is exactly the total variation of h, but in general, it is only upper bounded by the total variation. In the case of our function f involved in the estimation of SW, $V_f < +\infty$ holds since f is Lipschitz continuous. Another limitation of the previous bound is that the rate of convergence of the star discrepancy D_M^* of a sequence is most of the time not explicit and difficult to compute [136].

Nevertheless, this proposition ensures that if the rate of convergence of the star discrepancy of a sequence is better than $\mathcal{O}(\frac{1}{\sqrt{M}})$, for M large enough the approximation of the quasi Monte Carlo approximation using this sequence will outperform the one of classical Monte Carlo.

In the following, we present two L.D.S. defined on the unit square $[0,1]^d$, and see how their star discrepancy decreases with M. We then focus on practical methods to map these sequences from the hypercube to the hypersphere $\mathbb{S}^{d-1}$.

5.3.2.1.1 Halton sequence

The Halton sequence $(u_i)_{i \in \mathbb{N}} \in (\mathbb{R}^d)^{\mathbb{N}}$ [83] is a generalization of the von der Corput sequence [188]. In the following, we write, for any integer i, $c_l(i)$ the coefficients from the expansion of i in base b, and we define the radical-inverse function in base b as

$$\phi_b(i) = \sum_{l=0}^{+\infty} c_l(i) b^{-l-1}, \forall i \in \mathbb{N}.$$

The Halton sequence in dimension d is then defined as

$$u_i = (\phi_{b_1}(i), \dots, \phi_{b_d}(i))^T,$$

where b_i is chosen as the i-th prime number.

5.3.2.1.2 Sobol sequence

This sequence uses the base $b = 2$. To generate the j-th coordinate of the i-th point u_i in a Sobol sequence [175], one needs a primitive polynomial of degree n_j in $\mathbb{Z}/2\mathbb{Z}[X]$,

$$X^{n_j} + a_{1,j}X^{n_j-1} + a_{2,j}X^{n_j-2} + \ldots + a_{n_j-1,j}X + 1.$$

This polynomial is used to define a sequence of positive integers $(m_{k,j})$ by recurrence, with $+_{\mathbb{Z}/2\mathbb{Z}}$ the inner law of $\mathbb{Z}/2\mathbb{Z}$:

$$m_{k,j} = 2a_{1,j}m_{k-1,j} +_{\mathbb{Z}/2\mathbb{Z}} 2^2a_{2,j}m_{k-2,j} +_{\mathbb{Z}/2\mathbb{Z}} \ldots +_{\mathbb{Z}/2\mathbb{Z}} 2^{n_j}m_{k-n_j,j} +_{\mathbb{Z}/2\mathbb{Z}} m_{k-n_j,j}.$$

The values $m_{k,j}$, for $1 \leq k \leq n_j$, can be chosen arbitrarily provided that each one is odd and less than 2^k. Then one generates what is called direction numbers:

$$v_{k,j} = \frac{m_{k,j}}{2^k}.$$

The j-th coordinate of u_i is then obtained as

$$u_{i,j} = \sum_{k=1}^{+\infty} c_k(i)v_{k,j}.$$

5.3.2.1.3 Convergence rate of Halton and Sobol sequences Both sequences (Halton and Sobol) have a star discrepancy which converges to 0 (which means that they are indeed L.D.S.). The convergence rate is given by the following property [130] [137].

Proposition 5.5 : Let $(u_m)_{m\in\mathbb{N}^*}$ be either the Halton sequence or Sobol sequence in $[0,1]^d$. Then for $M \geq 1$, we have

$$D^*_M(u_1, \ldots, u_M) \leq c_d \frac{log(M)^d}{M}$$

where c_d is a constant that depends only on the dimension.

Thanks to Eq. 5.12, for any function h such that $V_h < +\infty$ (which is the case for our function f), this implies a convergence rate of the Monte Carlo estimator using these sequences in $\mathcal{O}(\frac{log(M)^d}{M})$, which means $\mathcal{O}(M^{-1+\epsilon})$ for every $\epsilon > 0$. This convergence rate is better than the one of classical Monte Carlo with i.i.d. sequences, even if the rate of convergence slows down when the dimension increases, because of the term $\log(M)^d$.

Remark 5.2 : Note that L.D.S. are designed to mimic the behavior of a random uniform sampling in $[0,1]^d$ while being completely deterministic. This deterministic behavior leads to patterns in the sampling; because of those patterns, the higher the dimension, the harder it is for those to fill the "gaps" in $[0,1]^d$. Moreover, the term $\log(M)^d$ implies that one needs M to be very large (exponential) to get the same level of space coverage in high dimension than in low dimension.

Remark 5.3 : Observe that both for Sobol and Halton sequences, generating M values has a complexity in $\mathcal{O}\big(Mlog_b^2(M)\big)$, where b is the base (or smallest basis for Halton) chosen.

5.3.2.1.4 L.D.S. on the sphere

To our knowledge, there is no true L.D.S. on the unit sphere $\mathbb{S}^{d-1}$ for $d \geq 3$, this question remaining an active research area. Practitioners typically map L.D.S. from the hypercube to the hypersphere, using one of the methods described below:

- **Equal area mapping** [7]: this method is only defined for mapping points in the unit square to $\mathbb{S}^2$. Denoting $(z_1, z_2) \in [0,1[^2$, one gets a point $u = \Phi(2\pi z_1, 1-2z_2)$ on $\mathbb{S}^2$ with:

$$\Phi(\eta, \beta) = \left(\sqrt{1-\beta^2}\cos(\eta), \sqrt{1-\beta^2}\sin(\eta), \beta\right), \ \eta, \beta \in [0,1[. \tag{5.13}$$

- **Spherical coordinates** [11]: This method maps the points from an L.D.S. in $[0,1]^{d-1}$ to $\mathbb{S}^{d-1}$ by using the spherical coordinates. Unfortunately, we found that the resulting sampling is usually not competitive compared to other sampling methods.

- **Normalization onto the sphere** [18]: An L.D.S. is generated in the d-hypercube $[0,1]^d$ and mapped to $\mathbb{R}^d$ using the inverse cumulative distribution function of the standard normal distribution (separately on each dimension). Then each point in the resulting sequence is normalized by its norm to map it onto $\mathbb{S}^{d-1}$.

Specific case of $\mathbb{S}^2$.
In the specific case of $\mathbb{S}^2$, it has been shown by [7] that if u is an L.D.S in $[0,1]^2$ and Φ the equal area mapping defined in Eq. 5.13, the spherical cap discrepancy $D_{\mathbb{L}_2,M}\big(\Phi(P)\big)$ (see definition Def. 5.3 in the next section) of the mapped sequence is in $\mathcal{O}\left(\frac{1}{M^{1/2}}\right)$. However, their experiments showed that the correct order seems rather to be $\mathcal{O}\left(\frac{log^c(M)}{M^{3/4}}\right)$ for $1/2 \leq c \leq 1$.

5.3.2.2 Deterministic point sets on $\mathbb{S}^{d-1}$

This section details different methods to design well distributed point sets on $\mathbb{S}^{d-1}$. Contrary to the L.D.S. defined above, these point sets are defined directly on the sphere, in order to be approximately uniformly distributed on $\mathbb{S}^{d-1}$. To measure this uniformity, we can rely on the notion of spherical cap on the sphere: a spherical cap of center $c \in \mathbb{S}^{d-1}$ and $t \in [-1,1]$ is defined as

$$C(c,t) = \{x \in \mathbb{S}^{d-1} \mid \langle x, c\rangle > t\}. \tag{5.14}$$

In other words, a spherical cap is the intersection of a portion of the sphere and a half-space (see Fig. 5.3 for an illustration).

To the best of our knowledge, there is no equivalent to the Koksma-Hlawka inequality for the sphere in full generality [31]. A sequence of points $\{u_n\}$ on $\mathbb{S}^{d-1}$ is said asymptotically uniformly distributed on $\mathbb{S}^{d-1}$ if for every spherical cap C, the proportion of points inside the cap, converges to the measure of the cap $s_{d-1}(C)$. It can be shown that this assumption is equivalent to assume that for every continuous function h, the Monte Carlo approximation $\frac{1}{M}\sum_{k=1}^{M} h(u_k)$ converges to $\mathbb{E}_{\theta\sim s_{d-1}}[h(\theta)]$.

In order to get a non asymptotic notion of the uniformity of a point set on $\mathbb{S}^{d-1}$, we can rely on different notions of spherical cap discrepancy on the sphere, defined as follows.

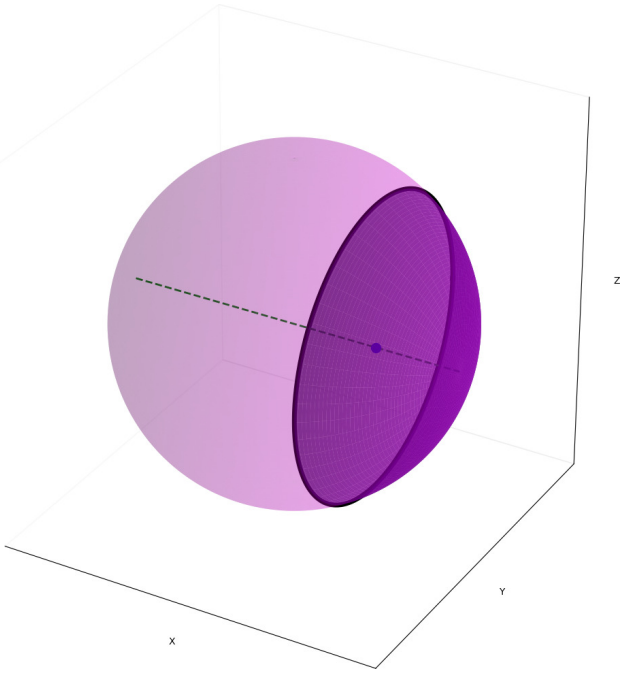


Figure 5.3: Illustration of a spherical cap on $\mathbb{S}^2$. The circle represents the intersection of the plane $\langle x, c\rangle = t$ with the sphere, and the purple colored area is the cap $C(c,t)$ as noted in Eq. 5.14.

Definition 5.3 : The spherical cap max-discrepancy of a point set P_M of size M is defined as [119]:

$$D_{max}(P_M) = \sup_{c\in\mathbb{S}^{d-1}, t\in[-1,1]} \left\{ \left| \frac{|P_M \cap C(c,t)|}{M} - s_{d-1}\big(C(c,t)\big) \right| \right\}.$$

The spherical cap $\mathbb{L}_2$-discrepancy of a point set P_M of size M is defined as [31]:

$$D^2_{\mathbb{L}_2}(P_M) = \left\{ \int_{-1}^{1} \int_{\mathbb{S}^{d-1}} \left| \frac{|P_M \cap C(c,t)|}{M} - s_{d-1}\big(C(c,t)\big) \right|^2 ds_{d-1}(c)dt \right\},$$

where $C(c,t)$ is a spherical cap of center c and height t.

Again, the idea is to compare the proportion of points in P_M that fall inside a spherical cap with the measure of the cap. This comparison is done for all possible caps on the sphere, and D_{max} represents the worst error over all possible caps, while $D^2_{\mathbb{L}_2}$ represents the average squared error over all possible caps.

When using Q.M.C. on the hypersphere to approximate the integral of functions h, another notion often used in the literature is the worst-case (integration) error (W.C.E.) on a Banach space of functions, which is the largest possible error made by the method on the space. For instance, on $H^\alpha(\mathbb{S}^{d-1})$.

Definition 5.4 : For $P_M = \{u_1, \dots, u_M\}$, for $\alpha \in \mathbb{N}$

$$WCE\Big(P_M, H^\alpha(\mathbb{S}^{d-1})\Big) = \sup_{h\in H^\alpha(\mathbb{S}^{d-1})} \left| \frac{1}{M}\sum_{m=1}^{M} h(u_m) - \frac{1}{s_{d-1}(\mathbb{S}^{d-1})} \int_{\mathbb{S}^{d-1}} h(w) ds_{d-1}(w) \right|.$$

Under some regularity condition, a sufficient and necessary one being $\alpha \geq \frac{1}{2} + \frac{d-1}{2}$ for $H^\alpha(\mathbb{S}^{d-1})$, [32] shows that optimizing the spherical cap $\mathbb{L}_2$-discrepancy is equivalent to optimizing the W.C.E. thanks to the Stolarsky's invariant principle [178]. In the case of our function f, we have seen that f is regular enough in the specific case of $\mathbb{S}^1$, since $f \in H^\alpha(\mathbb{S}^1)$ with $\alpha = 1 = \frac{1}{2} + \frac{1}{2}$. However in dimension larger than 3, this result does not hold anymore since f does not belong to any Sobolev space $H^\alpha(\mathbb{S}^d)$ with $\alpha > 1$.

5.3.2.2.1 Fibonacci point set on $\mathbb{S}^2$

Denoting φ the polar angle and χ the azimuthal angle forming the geographical coordinates (φ, χ), we retrieve the Cartesian coordinates (x, y, z) using the spherical coordinates (see Fig. 5.4 for an illustration). Noting $\phi = \frac{1+\sqrt{5}}{2}$ the golden ratio, the m-th point $u_m = (\varphi_m, \chi_m)$ of the Fibonacci point set is given by

$$\begin{aligned} \varphi_m &= \arccos\left(\frac{2m}{2M+1}\right), \\ \chi_m &= 2m\pi\phi^{-2}. \end{aligned}$$

It is a simple and efficient way, convergence rate wise, to generate points on $\mathbb{S}^2$ for the quasi-Monte Carlo method but it is only defined on $\mathbb{S}^2$. The complexity of the generation is linear in M, and according to [118], the corresponding convergence rate for the W.C.E. and the $\mathbb{L}_2$-spherical cap discrepancy is in $\mathcal{O}(\frac{1}{M^{3/4}})$. For an extensive list of other popular point configurations on $\mathbb{S}^2$, see [85].

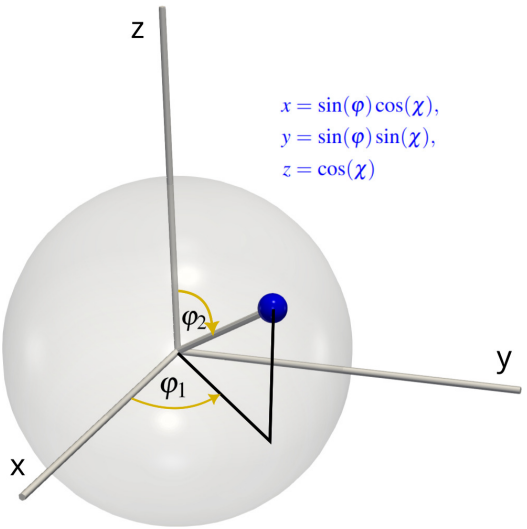


Figure 5.4: Illustration of the spherical coordinates in $\mathbb{R}^3$ for points on the sphere $\mathbb{S}^2$.

5.3.2.2.2 Equi-distributed points generated by the discrete s-Riesz energy

Another classical way to define equi-distributed point sets on the hypersphere is to rely on optimization. In such methods, the point set P_M is defined as the minimizer of a certain energy functional E_s,

$$P_M^* := \underset{u_1,\ldots,u_M \in \mathbb{S}^{d-1}}{\arg\min} \; E_s(u_1, \ldots, u_M).$$

The most common energy functional is the s-Riesz energy, which is defined as follows.

Definition 5.5 : For $s \geq 0$ and $P_M = \{u_1, \dots, u_M\}$ a set of points on $\mathbb{S}^{d-1}$, the s-Riesz energy of P is defined as

$$E_s(P_M) = \begin{cases} \sum_{i \neq j} \frac{1}{\|u_i - u_j\|^s} & \text{if } s > 0, \\ \sum_{i \neq j} \log \frac{1}{\|u_i - u_j\|} & \text{if } s = 0. \end{cases}$$

The resulting point set is called a minimal s-energy configuration. The s-Riesz energy can also be defined for $s < 0$, in this case the point set P_M is obtained as the maximizer of $E_s = \sum_{i \neq j} \|u_i - u_j\|^s$ [31]. Minimising E_s is non trivial, the functional being not convex, and the problem becomes more complex when the dimension increases. Minimal energy configuration points for E_s are called Fekete points and it is known that for $0 \leq s < d$, these sets are asymptotically uniformly distributed with respect to the normalized surface measure s_{d-1}, which means that Monte Carlo estimates using the Fekete points converge to the integral against s_{d-1} [119].

The spherical cap $\mathbb{L}_2$-discrepancy of a point configuration is minimal if and only if the sum of distances in the configuration is maximal. This would correspond to maximizing a s-Riesz energy for $s = -1$ [31]. However, the link between the configurations of minimal s-Riesz energy and the max or $\mathbb{L}_2$ discrepancies of these configurations is in general not straightforward, see [31], [119], [82]. For $0 \leq s < d$, and P_M a minimizer of size M of the Riesz s-energy on $\mathbb{S}^{d-1}$, the authors of [119] show that

$$D_{max}(P_M) \lesssim \mathcal{O}\left(\max\left(M^{-\frac{2}{d(d-s+1)}}, M^{-\frac{2(d-s)}{d(d-s+4)}}\right)\right).$$

This implies that $D_{max}(P_M) \xrightarrow[M \to +\infty]{} 0$, but the speed of convergence degrades with the dimension d, which means that the uniformity of these configurations is likely to suffer from the curse of dimensionality. Fig. 5.5 shows an example of s-Riesz points and Fibonacci points on $\mathbb{S}^2$ with 500 points.

Remark 5.4 : Since computing Riesz point configurations involves optimization (with a non linear complexity), the time needed to generate those points can be impractical. Note that generally the generation of the s-Riesz configuration points has a runtime complexity of $\mathcal{O}(TM^2)$, where T is the number of iterations of the optimization loop.

In the specific case of S^1, the Fekete points are unique up to a rotation, and are the M-th unit roots (see [82] and see Fig. 5.6 for an illustration):

$$\left\{ e^{\frac{2ik\pi}{M}} \mid k = 0, \dots, M-1 \right\}.$$

This explains why for 2D discrete measures, a uniform grid on $\mathbb{S}^1$ gives better results than any other sampling method for computing SW_2^2, as we will see in Sec. 5.4.

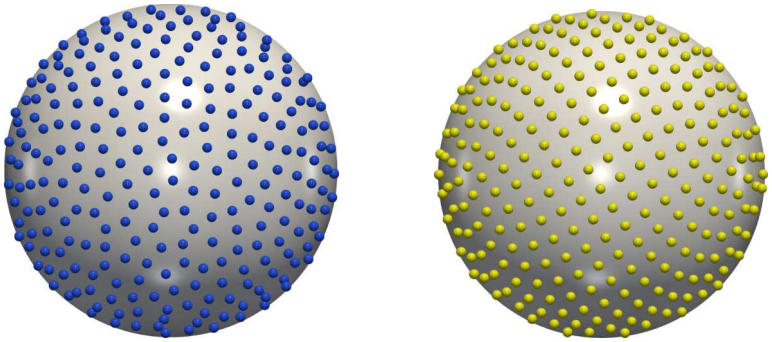


Figure 5.5: Illustration of s-Riesz points (on the left) and Fibonacci points (on the right) on S^2, with 500 points for both configurations.

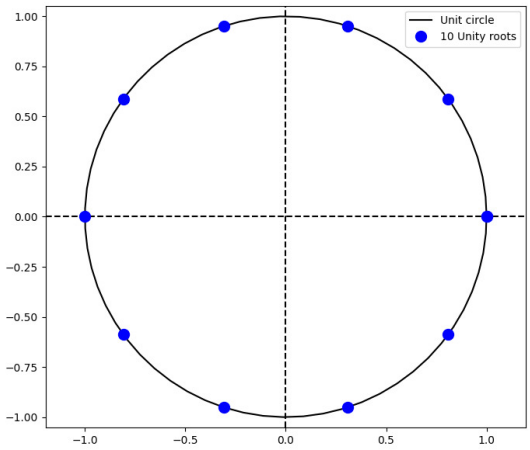


Figure 5.6: Plot of the 10-th unity roots, i.e solutions to the equation $z^{10} = 1$.

5.3.2.3 Random Quasi Monte-Carlo

The principle of Randomized Quasi-Monte Carlo (R.Q.M.C.) methods is to reintroduce stochasticty in Q.M.C. sequences. Indeed, Q.M.C. methods such as the ones described in Sec. 5.3.2.1 and Sec. 5.3.2.2 are deterministic. For a given M, the estimator given by one of these methods is always the same. As such, we cannot easily estimate the error or the variance of the Monte Carlo approximation. Besides, while results such as the Koksma-Hlawka inequality ensures that they converge at a certain rate, the different quantities involved in the inequality are much more complex to estimate than the one involved in the Central Limit Theorem. Random Quasi-Monte Carlo methods were especially designed to recover this ability to estimate the error easily. These sequences are usually defined on $[0, 1]^d$.

Definition 5.6 ([137]) **:** Let $\{\hat{u}_i\}_{i\geq 1}$ be a sequence of points in $[0, 1]^d$. It is said to be suitable for R.Q.M.C. if $\forall i$, $\hat{u}_i \sim \mathcal{U}([0, 1]^d)$ and if there exist a finite $c > 0$ and $K > 0$

such that for all $M \geq K$,

$$\mathbb{P}\left[D_M^*(\hat{P}_M) < c\frac{log^d(M)}{M}\right] = 1, \quad \text{where} \quad \hat{P}_M = \{\hat{u}_1, \dots, \hat{u}_M\}.$$

Denoting $X_M = \frac{1}{M}\sum_{i=1}^{M} h(\hat{u}_i)$ the empirical estimator of $\mathbb{E}_{\theta\sim s_{d-1}}[h(\theta)]$, the assumption $\hat{u}_i \sim \mathcal{U}([0,1]^d)$ implies that X_M is unbiased. Besides, the previous inequality implies that if $\{\hat{u}_i\}_{i\geq 1}$ is suitable for R.Q.M.C., then the variance of X_M is bounded by $c^2V_h^2\frac{log^{2d}(M)}{M^2}$. For functions h such that $V_h < \infty$, this yields a convergence rate in $\mathcal{O}\big(\log^d(M)/M\big)$, similar to the one of low discrepancy sequences.

Once a randomization method is chosen (such that it provides suitable R.Q.M.C. sequences), the process can be repeated several times to obtain K independent random estimators $X_M^1, \dots X_M^K$ of $\mathbb{E}_{\theta\sim s_{d-1}}[h(\theta)]$. The agregated estimate $X_{M,K} = \frac{1}{K}\sum_{k=1}^{K} X_M^K$ has a variance decreasing in $\mathcal{O}\big(\log^d(M)/(MK^{-1/2})\big)$. One of the key advantages of this approach is that this variance (or confidence intervals) can be estimated by the empirical variance of the K independent estimators.
There are several ways to generate sequences from low discrepancy sequences on $[0,1]^d$ in order to make them suitable for R.Q.M.C.. One of the most simple methods consists in applying the same random shift U to all points in the sequence, and taking the result modulo 1 componentwise [110]. More involved methods, such as Digital shift or Scrambling, are described in [110] and [137].

However, to the best of our knowledge, there is no proper R.L.D.S. on the sphere, as stated by [127]. In practice, R.L.D.S. on the unit cube are mapped onto the sphere by the methods described in paragraph 5.3.2.1.4. Another possibility, as done in [127], is to generate a random rotation matrix and apply it directly on point configurations on $\mathbb{S}^{d-1}$, such as the ones described in Sec. 5.3.2.2.

5.3.3 Spherical Sliced Wasserstein

A sampling method based on a Sliced-Wasserstein type discrepancy on the sphere $\mathbb{S}^{d-1}$ was developped by [25] for $d \geq 3$. We denote $\mathbb{C}_{d,2}$ the set of great circles of $\mathbb{S}^{d-1}$, a great circle being the intersection between a plane of dimension 2 and $\mathbb{S}^{d-1}$ [94]. The authors of [25] define a pseudo distance, called Spherical Sliced Wasserstein distance, between two probability measures Θ, Ξ defined on $\mathbb{S}^{d-1}$:

$$SSW_2^2(\Theta, \Xi) = \int_{\mathbb{C}_{d,2}} W_2^2(\pi_C \# \Theta, \pi_C \# \Xi) d\zeta(C), \tag{5.15}$$

where for all $x \in \mathbb{S}^{d-1}$, $\pi_C(x) = \arg\min_{y\in C} d_{\mathbb{S}^{d-1}}(x,y)$ with $d_{\mathbb{S}^{d-1}}(x,y) = \arccos(\langle x, y\rangle)$ [70] and ζ is the uniform distribution over $\mathbb{C}_{d,2}$.
As shown in [25], this distance can be used to sample points on $\mathbb{S}^{d-1}$ by minimizing SSW_2 between a discrete measure $\Theta = \frac{1}{M}\sum_{i=1}^{M}\delta_{\theta_i}$ and the uniform measure $\Xi = s_{d-1}$ on $\mathbb{S}^{d-1}$. To this aim, for $C_1, \dots, C_L$ L independent great circles, they approximate $SSW_2^2(\Theta, \Xi)$ by its

Monte Carlo approximation $Z_L(\Theta, \Xi) = \frac{1}{L}\sum_{l=1}^{L} W_2^2(\pi_{C_l}\#\Theta, \pi_{C_l}\#\Xi)$. Then, they note that $\pi_{C_l}\#s_{d-1} = s_1$ [93] for each l, and derive a closed form for $W_2^2(\pi_{C_l}\#\Theta, s_1)$ based on [52]. The final distance $SSW_2^2(\Theta, \Xi)$ can then be optimized with respect to the point positions θ_i with a projected gradient descent.

Remark 5.5 : There are cases in which SSW is a metric:

- Based on [157], SSW is a metric between any two probability measures on $\mathbb{S}^2$.
- A result from [114] also shows that SSW is a metric between any two absolutely continuous probability measures with continuous density functions on $\mathbb{S}^{d-1}$ for $d \geq 3$.

Remark 5.6 : Noting T the number of iterations for the gradient descent algorithm, and L as above, then the time complexity of this method is in $\mathcal{O}(TLM\log(M))$.

Remark 5.7 : Notice that SSW's form is similar to the $\mathbb{L}_2$-spherical cap discrepancy, where instead of averaging the "error" made by the sampling on a spherical cap, it averages the "error" made by the sampling on a great circle.

5.3.4 Variance reduction

All methods described so far are based on the idea of generating points on the sphere in such a way that these points are sufficiently well distributed to be used for Monte Carlo integration, and ideally yield faster convergence than M.C. with i.i.d. sequences. These point sequences or point sets are defined independently of the function to be integrated.

More involved approaches, such as importance sampling or control variates, use the knowledge of the function to be integrated to improve Monte Carlo estimators by decreasing their variance. Recently, two control variates based methods have been developped to estimate the Sliced Wasserstein distance. A control variate is a centered random vector $Y \in \mathbb{R}^p$, easy to sample, with finite second moments. Assume we want to estimate $\mathbb{E}_{\theta\sim s_{d-1}}[f(\theta)]$. Writing $\theta_1, \dots, \theta_M$ i.i.d. samples of $\theta \sim s_{d-1}$ and $Y_1, \dots, Y_M$ M independent copies of the random centered vector Y, we consider the following estimator

$$\frac{1}{M}\sum_{i=1}^{M}[f(\theta_i) - \beta^T Y_i],$$

where $\beta \in \mathbb{R}^p$ is a constant vector to be determined. The variance of this estimator is proportional to $\mathrm{Var}(f(\theta) - \beta^T Y)$. It follows that if we write β^* the parameter minimizing this variance, then the pair $(\mathbb{E}(f(\theta)), \beta^*)$ is solution of the least square problem

$$\min_{(\zeta,\beta)\in\mathbb{R}\times\mathbb{R}^p} \mathbb{E}[(f(\theta) - \zeta - \beta^T Y)^2].$$

An empirical version of this quadratic problem on a sample $(\theta_1, \dots, \theta_M)$ writes

$$(\widehat{\mathbb{E}(f(\theta))}_M, \beta_M) = \underset{\zeta,\beta\in\mathbb{R}\times\mathbb{R}^p}{\arg\min} \|\mathbf{F} - \zeta\mathbb{1}_M - \mathbf{Y}\beta\|_2^2 \tag{5.16}$$

where $\mathbf{F} = \left(f(\theta_i)\right)_{i=1,\dots,M}^T$, $\mathbb{1}_M = (1, \dots, 1)^T \in \mathbb{R}^M$, and $\mathbf{Y} = \left(Y_i^T\right)_{i=1,\dots,M} \in \mathbb{R}^{M\times p}$.

Recently, [129] introduced a Sliced Wasserstein distance estimation using Gaussian control variates and [109] developped a method using spherical harmonics control variates. We focus only on [109] here, since their method yields much better experimental results. In their work, [109] chose Spherical Harmonics [122] as control variates. Spherical harmonics are functions which form an orthonormal basis (ϕ_i) of the Hilbert space $L^2(\mathbb{S}^{d-1}, s_{d-1})$. In this setting, the random variable Y is thus chosen as $Y = (\phi_i(\theta))_{i=1,\dots,p}$, with $\theta \sim s_{d-1}$. In practice, the number p is chosen as $p = L_{n,d} = \sum_{l=1}^{n} N(d, 2l)$, the number of spherical harmonics of even degree up to $2n$, with $N(d,n) = (2n+d-2)\frac{(n+d-3)!}{(d-2)!n!}$ the number of spherical harmonics of degree n in dimension d.

[109] then computes the solution $(SHCV^2_{M,n}, \beta_M)$ of (5.16) on a sample $(\theta_1, \dots, \theta_M)$ and uses the control variates estimator $SHCV^2_{M,n}$ as estimator of the (squared) Sliced Wasserstein distance. They prove the following convergence property.

Proposition 5.6 : Let μ, ν be two discrete measures in $\mathbb{R}^d$ with finite moments of order 2 and let $d \geq 2$. For any sequence of degrees $n = (n_M)_M$ such that $n_M = o\left(M^{1/\left(2(d-1)\right)}\right)$ as $M \longrightarrow +\infty$, we have

$$\left|SHCV^2_{M,n}(\mu,\nu) - SW_2^2(\mu,\nu)\right| = \mathcal{O}_{\mathbb{P}}\left(\frac{1}{nM^{1/2}}\right), \tag{5.17}$$

where the notation $X_n = \mathcal{O}_{\mathbb{P}}(a_n)$ means that the sequence $\frac{X_n}{a_n}$ is stochastically bounded [3].

Notice that since $n_M = o\left(M^{1/\left(2(d-1)\right)}\right)$, in high dimensions d the global convergence rate is similar to that of the classical Monte Carlo method described in Sec. 5.3.1.1.

5.4 Experimental results

This section presents experimental results from all the different sampling strategies presented in Sec. 5.3, on a variety of datasets. To provide representative results, we select datasets spanning a range of dimensions going from 2 to 28×28. Those include a toy dataset and three "real-life" ones. We first present results on Gaussian mixtures in the following dimensions $\{2, 3, 5, 10, 20, 50\}$, the six ground truths (true distances) are estimated using 10^8 angles θ. Secondly, we show some dimensionality reduction results on 12 different datasets of persistence diagrams (for the case of 2 dimensional discrete measures). Then we show some convergence results in the specific case of measures in 3 dimensions. Specifically, we compare different estimations of the Sliced Wasserstein distance between 3D point clouds taken from the ShapeNetCore dataset, see [43]. Finally we compare different dimensionality reduction results on the MNIST dataset [107]. For the experiments on the Gaussian mixtures we compare the listed strategies with the following sampling numbers $\{100, 300, 500, 700, 900, 1100, 2100, 3100, 4100, 5100, 6100, 7100, 8100, 9100, 10100\}$. Otherwise, we use the following sampling numbers $\{100, 1100, 2100, 3100, 4100, 5100,$

[3] The notation $X_n = \mathcal{O}_{\mathbb{P}}(a_n)$ means that for all $\epsilon > 0$, there exist finite $K > 0$ and $N > 0$ such that $\mathbb{P}[|X_n| > Ka_n] < \epsilon$ for all $n > N$.

6100, 7100, 8100, 9100, 10100}. Tab. 5.3 displays the acronyms of all the sampling methods compared in the following experiments. For each sampling method from Tab. 5.3, there are two variants finishing with the term "Area Mapped" and two variants finishing with the term "Normalized Mapped". The first one means that we applied the equal area mapping detailed in paragraph 5.3.2.1.4. The second one means we normalize each point generated by those methods, this normalization method is also detailed in paragraph 5.3.2.1.4.

Name	Legends	Dimensions
Riesz Randomized	R.R.	2,3,5,10,20,50
Uniform Sampling	U.S.	2,3,5,10,20,50
Othornormal Sampling	O.S.	2,3,5,10,20,50
Halton Area Mapped	H.A.M.	2,3
Halton Randomized Area Mapped	H.R.A.M.	3
Halton Normalized Mapped	H.N.M.	5,10,20,50
Halton Randomized Normalized Mapped	H.R.N.M	5,10,20,50
Fibonacci Point Set	F.P.S.	3
Fibonacci Randomized Point Set	F.R.P.S.	3
Sobol Area Mapped	S.A.M.	3
Sobol Randomized Area Mapped	S.R.A.M.	3
Sobol Normalized Mapped	S.N.M.	5,10,20
Sobol Randomized Normalized Mapped	S.R.A.M.	5,10,20
Spherical Harmonics Control Variates	S.H.C.V.	3,5,10,20
Spherical Sliced Wasserstein Randomized	S.S.W.R.	3,5,10,20,50

Table 5.3: For each method used in this experimental part, associated acronym, and list of dimensions where this method is used.

5.4.1 Implementation of the sampling methods

This section provides details on the implementations used for the sampling methods, and specifies how the parameters are set. The implementations used are grouped and are available here `https://anonymous.4open.science/r/SW-Sampling-Guide-C157/README.md`.

- **Classical M.C. methods:** For both methods we used python included functions to sample a Gaussian variable and to sample orthogonal matrices in d dimension. For sampling orthogonal matrices we use the following python library scipy.stats.ortho_group `https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ortho_group.html`.

- **Halton & Sobol sequences:** In dimension 3 and less, we use python implementations from the library scipy.stats.qmc (`https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.qmc.Halton.html` & `https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.qmc.Sobol.html`. As for the parameters we set "scramble" to True to get the randomized version. For high dimensions, we use [109]'s implementation available here `https://github.com/RemiLELUC/SHCV`.

  Remark 5.8 : For the Sobol sequence, we noticed that the implementation provided by [109] cannot be used in dimension higher than 20.

- **Riesz point configuration:** We use a code provided by François Clement (`https://sites.math.washington.edu/~fclement/`), implementing a projected gradient descent method, where we choose the number of iterations as $T = 10$, the gradient step as 1 and $s = 0.1$. The function can be found in the riesz_noblur.py script in the repository `https://anonymous.4open.science/r/SW-Sampling-Guide-C157/README.md`.

- **Spherical Sliced Wasserstein:** We used the following implementation from [25] that can be found in POT library (Python Optimal Transport) `https://pythonot.github.io/auto_examples/backends/plot_ssw_unif_torch.html`. For the hyper-parameters we set the number of iteration $T = 250$, the learning rate $\epsilon = 150$ and the number of great circles $L = 500$. For the initialization, we generate $\theta_1, \dots, \theta_M \sim s_{d-1}$ following the method described in Sec. 5.3.1.1.

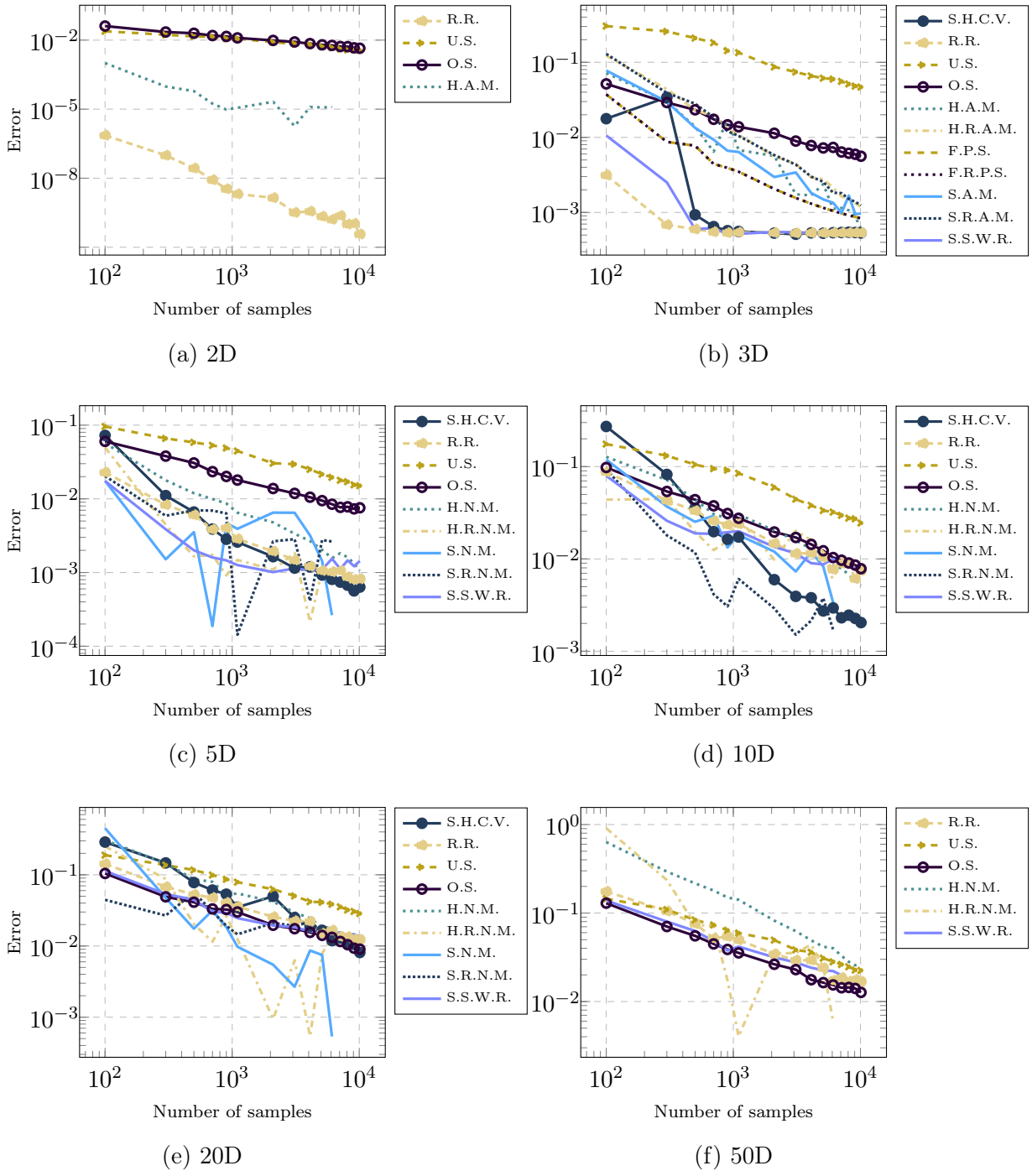
- **Spherical Harmonics Control Variates:** We use the implementation provided by [109], available in `https://github.com/RemiLELUC/SHCV`. They provide two possible functions **SHCV** and **SW_CV**, both functions return a value of a SW estimate. These functions differ in the way they implement the optimization of Eq. 5.16. Depending on the number of control variates, one of the functions is much more efficient than the other. For this reason, in our experiments, we use both functions and always keep only the minimal error among the two.

5.4.2 Gaussian data

This part details the experiments on a toy dataset chosen because it is simple to replicate and simple to understand. We compare different estimates of $SW_2^2(\mu_d, \nu_d)$ for $d \in \{2, 3, 5, 10, 20, 50\}$. We pick up [109]'s setting, using $\mu_d = \frac{1}{K}\sum_{i=1}^{K} \delta_{x_i}$ and $\nu_d = \frac{1}{K}\sum_{i=1}^{K} \delta_{y_i}$ with $x_1, \dots, x_K \sim \mathcal{N}(x, \mathbf{X})$, $y_1, \dots, y_K \sim \mathcal{N}(y, \mathbf{Y})$, where $K = 1000$. The means are drawn as $x, y \sim \mathcal{N}(\mathbb{1}_d, I_d)$ and the covariances are $X, Y = \Sigma_x \Sigma_x^T, \Sigma_y \Sigma_y^T$ where all entries of the matrices are drawn using the standard normal distribution. In Fig. 5.7, we show convergence curves generated by all the different sampling strategies in all the dimensions listed above. For random samplings, those curves are obtained after averaging the absolute error on a 100 runs. For deterministic sampling, those curves represent the absolute error of the approximation compared to the ground truth. Fig. 5.8 reports the distance estimation error as a function of computation time (in seconds). In both figures, both axes are log scaled. We can see in Fig. 5.7 that up to dimension 5, Q.M.C. methods are preferable convergence wise, then the orthonormal sampling is preferable in dimension 20 and 50. In contrast, we can see in Fig. 5.8 that for dimensions less than 10, the S.H.C.V. method has a better error, with similar running time. For higher dimensions, however, the orthonormal sampling is much faster, for a given error target.

Remark 5.9 : One may notice in Fig. 5.7b that both the S.H.C.V. method and the Q.M.C. method with the s-Riesz points (R.R.) reach a plateau at around 10^3 projections. Our hypothesis is that both methods have a better estimation of SW_2^2 than the simple random sampling with 10^8 projections that we use as a ground truth. We test this hypothesis in a simple case where $SW_2^2(\mu, \nu)$ can be computed explicitly. We define

$\mu = \frac{1}{2}[\delta_{x_1} + \delta_{x_2}]$ and $\nu = \frac{1}{2}[\delta_{y_1} + \delta_{y_2}]$, with $x_1, x_2 = (1,0,0)^T, (0,-1,0)^T$ and $y_1, y_2 = (0,0,1)^T, (0,0,-1)^T$. Simple computation yields $SW_2^2(\mu,\nu) = \frac{2(\pi-\sqrt{2})}{3\pi}$ (see Appendix C). Knowing the true value of $SW_2^2(\mu,\nu)$, we find that with 10^4 points, the Q.M.C. method with the s-Riesz points configuration and the S.H.C.V. methods already have errors one order smaller that ones made by uniform sampling with 10^8 points.

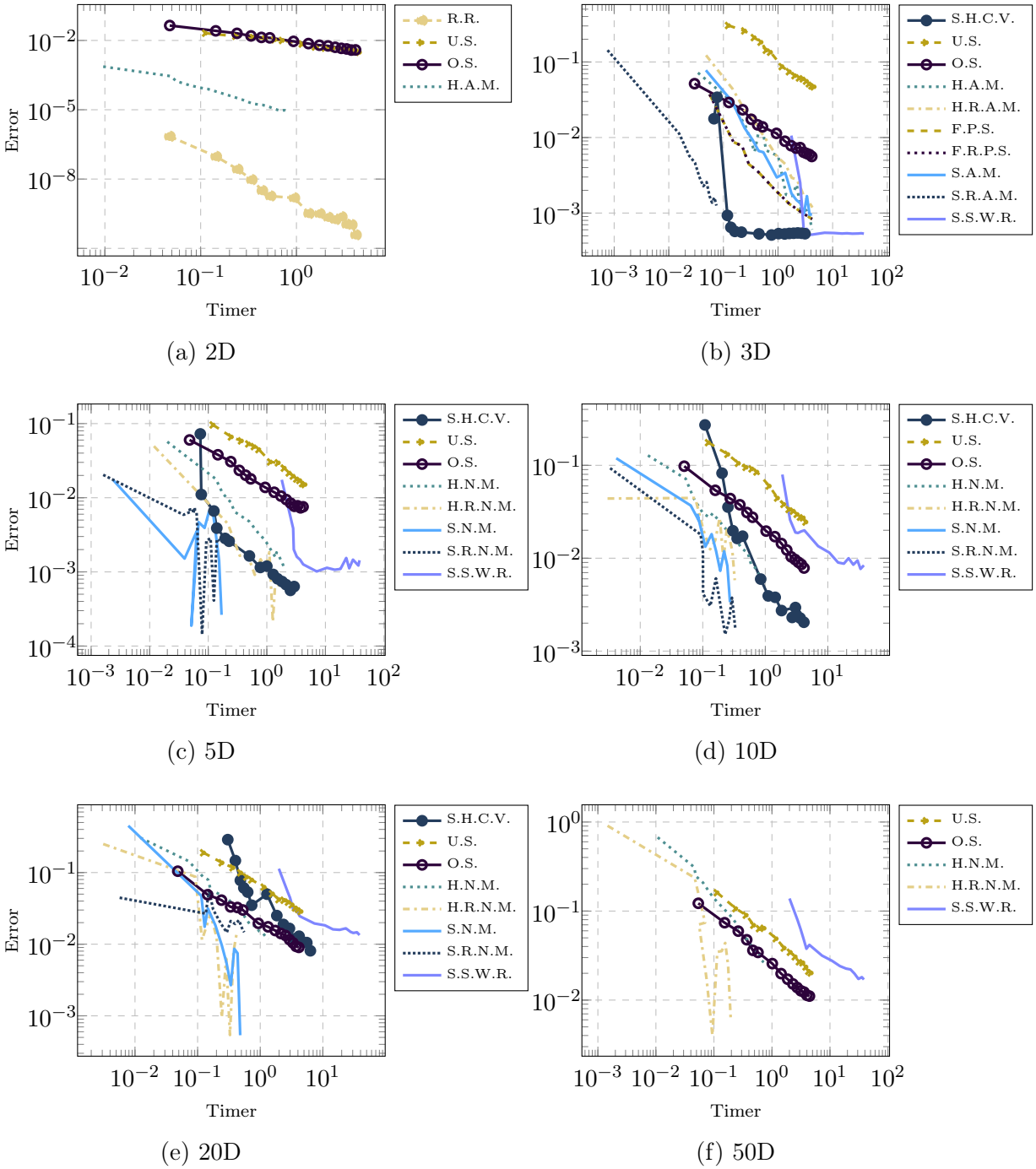


(a) 2D (b) 3D

(c) 5D (d) 10D

(e) 20D (f) 50D

Figure 5.7: Comparison of convergence rate results for the studied sampling methods (Gaussian data, Sec. 5.4.2).



(a) 2D (b) 3D

(c) 5D (d) 10D

(e) 20D (f) 50D

Figure 5.8: Distance estimation error as a function of computation time (seconds). Computation times include the point generation as well as the SW_2^2 distance approximation.

Remark 5.10 : Note that for the running time curves, we do not include the s-Riesz points configuration starting from the dimension 3 because it takes around 10^2 seconds to generate 10^3 points and 9×10^3 seconds to generate 10^4 points. However, observe that those points, once generated, can be stored once for all to compute other SW_2^2 distances or any other Monte Carlo estimation problems for functions defined on the unit sphere. This means that these configurations should not be discarded by default. For practical

applications where the number of SW_2^2 distances to compute is large, the computing time for these configurations can be factorized by the number of distances to compute and hence could become a negligible factor when the sampling number is moderate.

Remark 5.11 : Recalling the running time complexity $\mathcal{O}(TM^2)$ in paragraph 5.3.2.2.2 and the running time results above, this shows that one needs to spend 9×10^7 seconds to generate 10^6 points. This demonstrates the limitation of this sampling method in terms of scalability, in other words when one needs a very large sampling number.

5.4.3 Persistence diagrams reduction dimension score

The goal of this section is to evaluate the relevance of the sampling methods studied in Sec. 5.3, in the context of a concrete use case, involving two-dimensional real-life datasets. For that, we focus in this section on *persistence diagrams*, we refer again to Sec. 2.3 for a reminder of persistence diagrams (Fig. 5.9) .

Recall that two persistence diagrams can have a different number of points, so to make it a balanced transport problem one has to augment them. Formally, denoting $d_1 = \frac{1}{K_1}\sum_{k=1}^{K_1} \delta_{x_k}$, $d_2 = \frac{1}{K_2}\sum_{k=1}^{K_2} \delta_{y_k}$ the diagrams, and noting $\Delta_{d_1} = \frac{1}{K_1}\sum_{k=1}^{K_1} \delta_{\pi_\Delta(x_k)}$, $\Delta_{d_2} = \frac{1}{K_2}\sum_{k=1}^{K_1} \delta_{\pi_\Delta(y_k)}$ their projections on the diagonal Δ, one considers $\mu = \frac{1}{K}[K_1 d_1 + K_2 \Delta_{d_2}]$ and $\nu = \frac{1}{K}[K_2 d_2 + K_1 \Delta_{d_1}]$ as input measures with $K = K_1 + K_2$. Then the Sliced Wasserstein distance can be used to compare persistence diagrams as detailed by [41]. Also, the Wasserstein distance between persistence diagrams is equivalent to the Sliced Wasserstein distance using Prop. 2.14 and Bonnotte's result [29] (Theorem 5.1.5). Denoting $W_2^{\mathcal{D}}$ and W_2 the 2-Wasserstein distance for diagrams squared and the usual 2-Wasserstein distance between probability measures, recall the equivalence result we showed in Prop. 2.14:

$$W_2^{\mathcal{D},2}(D_1, D_2) \leq W_2^2(D_1, D_2) \leq 2W_2^{\mathcal{D},2}(D_1, D_2),$$

where D_1, D_2 are augmented diagrams. Bonnotte's result [29] states the following:

Proposition 5.7 : There exist a constant $C_{d,2} > 0$ such that for all probability measures μ, ν on a compact ball of radius R of $\mathbb{R}^d$,

$$SW_2^2(\mu, \nu) \leq W_2^2(\mu, \nu) \leq C_{d,2} R^{2-1/(d+1)} SW_2(\mu, \nu)^{1/(d+1)}.$$

Aggregating the inequalities, for the case of augmented persistence diagrams (by considering them as discrete measures on a bounded ball of radius R), we have:

$$\frac{1}{2} SW_2^2(\mu, \nu) \leq W_2^{\mathcal{D},2}(D_1, D_2) \leq C_{2,2} R^{5/3} SW_2(\mu, \nu)^{1/3}.$$

We present dimensionality reduction results on 12 ensembles of persistence diagrams [148] described in [149], which original scalar fields include simulated and acquired 2D and 3D ensembles from SciVis constests [133]. The dimensionality reduction techniques used are MDS [102] and t-SNE [189] applied on distance matrices obtained by the SW estimations between the persistence diagrams. For a given technique, one quantifies its ability to preserve the cluster structure of an ensemble by running the k-means algorithm in the

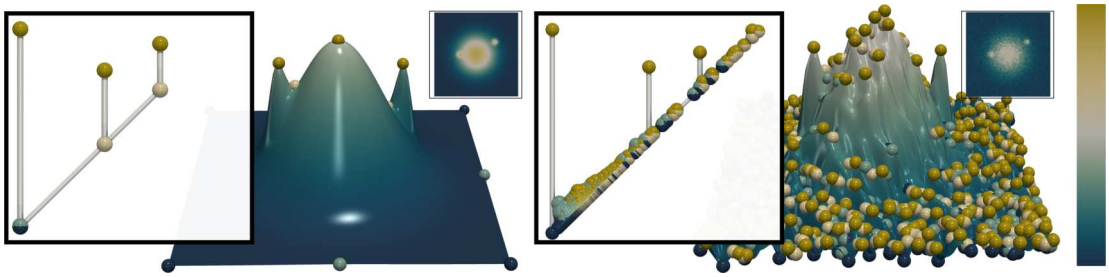


Figure 5.9: A simple example of a persistence diagram issued from a gaussian mixture (left). On the right one can see that the persistence diagram is stable to the addition of noise.

resulting 2D-layouts. Then one evaluates the quality of the clustering with the normalized mutual information (NMI) and adjusted rand index (ARI), which should both be equal to 1 for a clustering that is identical to the classification ground-truth. Tab. 5.4 shows the average clustering scores of both MDS [102] and t-SNE [189]. First we take the average from distance matrices made by each SW_2^2 estimates on all sampling number $\{100, 1100, 2100, 3100, 4100, 5100, 6100, 7100, 8100, 9100, 10100\}$. Then we average again over all the 12 different ensembles of persistence diagrams. One can see that all the methods are quite similar. But overall the s-Riesz points configuration, which are just the M-th unity roots up to a rotation, is slightly better.

Table 5.4: Average NMI and ARI scores for over all 12 ensembles of persistence diagrams.

Method	MDS NMI	t-SNE NMI
Riesz	0.74	0.65
Uniform	0.74	0.59
Orthonormal	0.75	0.63
Halton	0.74	0.58

Method	MDS ARI	t-SNE ARI
Riesz	0.64	0.51
Uniform	0.64	0.44
Orthonormal	0.64	0.48
Halton	0.63	0.41

5.4.4 3D Shapenet 55Core Data

This part details convergence results on a 3D dataset commonly used as a benchmark when studying shape comparison techniques. So as in [127] and [109], we took three 3D point clouds issued from the ShapenetCore dataset introduced by [43]. Among the different shapes in the dataset, we took one lamp, one plane and one bed; with all three of them having $K = 2048$ points. Fig. 5.10 displays the three datasets considered for this experiment.

Fig. 5.11 shows different convergence curves of Sliced Wasserstein estimates between the three point clouds. As in Sec. 5.4.2, the methods dominating are the Q.M.C., R.Q.M.C., S.S.W. and S.H.C.V. methods, especially the s-Riesz points configuration and the Spherical Sliced Wasserstein sampling.

5.4.5 MNIST reduction dimension score

The goal of this section is twofold. First, it evaluates the practical convergence of the studied sampling methods on real-life high-dimensional datasets. Second, it describes an

Figure 5.10: The three point clouds taken from the ShapenetCore dataset (a plane, a lamp and a bed).

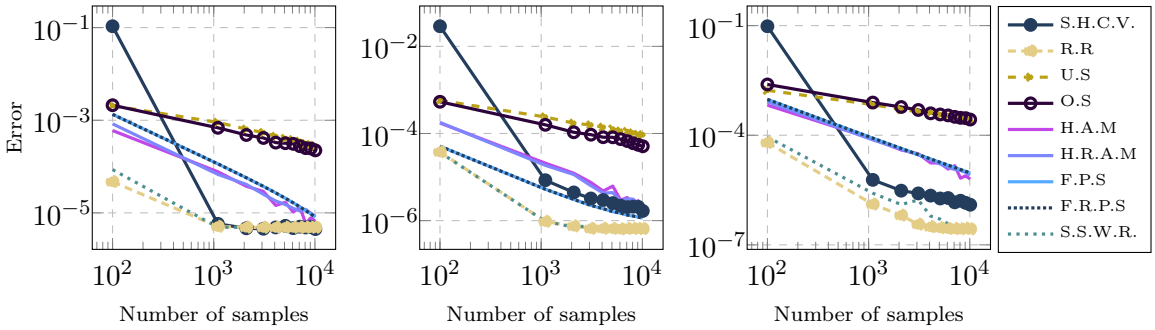


Figure 5.11: Comparison of convergence rate results from the different sampling methods. The first plot shows errors made with respect to the SW_2^2 distance between a lamp and a plane. The second one is between a plane and a bed. The last one corresponds to SW_2^2 between a plane and a bed.

application of the SW distance for high-dimensional data, namely, dimensionality reduction. For this, we select the classical MNIST dataset [107]. To construct our dataset, we represent each digit image as a point in $\mathbb{R}^{28\times 28}$. For each class $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, we select randomly 600 digit images and divide them into groups of 200. This results in 30 point clouds of 200 points each, in $\mathbb{R}^{28\times 28}$, with 10 ground-truth classes. Fig. 5.12 illustrates the 30×30 matrix of SW distances between all point clouds in the dataset. We use MDS and t-SNE to produce 2D layouts from the distance matrices generated by the various sampling methods with different sample sizes. We then apply a clustering algorithm to these 2D layouts and average the clustering scores (NMI and ARI, see Sec. 5.4.3) on all sampling numbers for all the studied sampling strategies. Results are provided in Tab. 5.5. In such high dimension ($d = 784$), we see that the performance of L.D.S. collapse, the three sampling methods standing out being the s-Riesz points configuration, the uniform sampling and the orthonormal sampling.

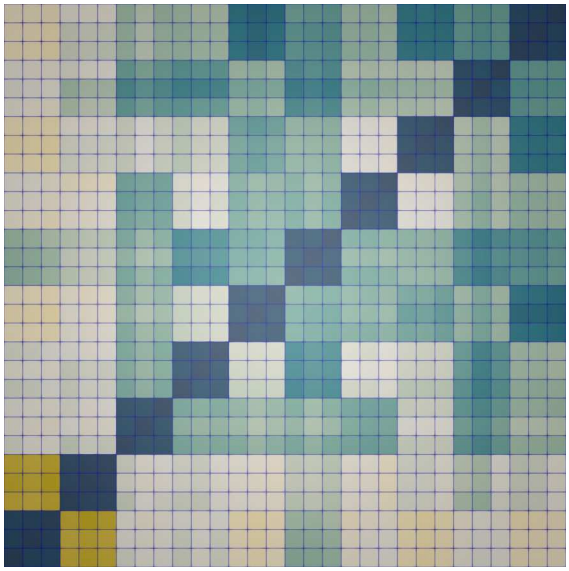


Figure 5.12: Sliced Wasserstein distance matrix of our dataset using 10^6 projections. All 10 classes, $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, of 3 members each are well represented in the matrix.

Table 5.5: Average NMI and ARI scores with standard deviation. Higher scores correspond to better clustering.

Method	MDS NMI	t-SNE NMI
Riesz	$1 \pm 0.$	0.98 ± 2e-2
Uniform	$1 \pm 0.$	0.97 ± 4e-2
Orthonormal	$1 \pm 0.$	0.98 ± 3e-2
Halton	0.91 ± 1e-1	0.91 ± 9e-2
S.S.W.	$1 \pm 0.$	0.98 ± 4e-2

Method	MDS ARI	t-SNE ARI
Riesz	$1 \pm 0.$	0.95 ± 7e-2
Uniform	$1 \pm 0.$	0.91 ± 1e-1
Orthonormal	$1 \pm 0.$	0.94 ± 8e-2
Halton	0.75 ± 2e-1	0.76 ± 2e-1
S.S.W.	$1 \pm 0.$	0.94 ± 1e-1

5.5 Application to a dictionary encoding method of persistence diagrams

In Chapt. 3, we introduced our Wasserstein dictionary method, and the main computational time bottleneck lies on the computations of the Wasserstein distances and the Wasserstein barycenters. One way to tackle this time bottleneck would be using the Sliced Wasserstein distance instead of the Wasserstein distance.

This part details how the Sliced Wasserstein distance can be used for a Wasserstein dictionary framework, or in this case a Sliced Wasserstein dictionary framework for persistence diagrams. The core of our dictionary method is the Wasserstein barycenter, thus we need a Sliced Wasserstein barycenter [28]. In this part, we reprise the notation introduced in Chapt. 3, where we enumerate a persistence diagram X as $X = \{x^1, \dots, x^K\}$.

Definition 5.7 : Given $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m)$ barycentric coefficients and denoting $\mathcal{D} = \{a_1, \dots, a_m\}$ augmented diagrams persistence diagrams, the Sliced Wasserstein barycenter is defined as

$$Y^* \in \arg\min E(Y) = \sum_{i=1}^{m} \lambda_i SW_2^2(Y, a_i). \tag{5.18}$$

The energy minimized in Eq. 5.18 is smooth, i.e it is $\mathcal{C}^1$ and its gradient is Lipschitzian

with:

$$\nabla E(Y) = \sum_{i=1}^{m} \lambda_i \int_{\mathbb{S}^1} \left(\begin{bmatrix} \langle y^1, \theta \rangle \\ \vdots \\ \langle y^K, \theta \rangle \end{bmatrix} - \begin{bmatrix} \langle a_i^{\tau_\theta \circ \sigma_{\theta,i}(1)}, \theta \rangle \\ \vdots \\ \langle a_i^{\tau_\theta \circ \sigma_{\theta,i}(K)}, \theta \rangle \end{bmatrix} \right) \theta^T ds_1(\theta), \tag{5.19}$$

Y being a persistence diagram, and τ_θ and $\sigma_{\theta,i}$ permutations of $[\![1, K]\!]$ which respectively orders Y and a_i on the line of angle θ.

Still the energy in Eq. 5.18 is non-convex [28], thus to compute a local minimum, one has to use a gradient descent scheme. However, in order to implement numerically this descent, one has to approximate the integral in Eq. 5.19 by a Monte-Carlo approximation (or quasi Monte-Carlo). More precisely, denoting $\theta_1, \dots, \theta_M$ angles sampled on the unit circle $\mathbb{S}^1$ Eq. 5.18 becomes

$$E_{\theta_1,\dots,\theta_M}(Y) = \sum_{i=1}^{m} \frac{\lambda_i}{M} \sum_{j=1}^{M} W_2^2(\pi_{\theta_j} \# Y, \pi_{\theta_j} \# a_i). \tag{5.20}$$

Following Eq. 5.20, we can compute a gradient analogous to Eq. 5.19 and implement a simple gradient descent scheme with gradient step 2 [28]. As for sampling the directions $\theta_1, \dots, \theta_M$, according to our previous discussion Sec. 5.6 and the experimental results Sec. 5.4, just taking the $M = 50$ unity roots is sufficient for this task. Now that we have a computable sliced Wasserstein barycenter, we can extend our Wasserstein dictionary framework to a sliced Wasserstein dictionary framework. Take $X_1, \dots, X_N$ input persistence diagrams, we want to optimize $\mathcal{D} = \{a_1, \dots, a_m\}$ a dictionary of m persistence diagrams and $\Lambda = \boldsymbol{\lambda}_1, \dots, \boldsymbol{\lambda}_N$ N vectors of barycentric coefficients of size m minimizing:

$$E(\Lambda, \mathcal{D}) = \sum_{n=1}^{N} SW_2^2\big(Y_{SW}(\boldsymbol{\lambda}_n, \mathcal{D}), X_n\big), \tag{5.21}$$

with $Y_{SW}(\boldsymbol{\lambda}_n, \mathcal{D})$ denoting the sliced Wasserstein barycenter of $\boldsymbol{\lambda}_n$ and $\mathcal{D}$. Unfortunately, unlike Chapt. 3, the sliced Wasserstein distance does not yield any optimal transport plans, nor the points in the barycenter $Y_{SW}(\boldsymbol{\lambda}_n, \mathcal{D})$ have an analytic formula (like in Chapt. 4). Thus, in practice to optimize E in Eq. 5.21, as in Chapt. 4, we have to use the automatic differentiation implemented in Pytorch, with Adam [98] as the optimizer.

Also, in Chapt. 4, we motivated the utilization of an alternate framework computing a robust Wasserstein barycenter based on a Wasserstein distance using a generic cost, more precisely using W_q for $q \in]1, 2[$ instead of W_2. However, the Sliced Wasserstein barycenter does not benefit this change. Indeed, recall that the $1D$ Wasserstein distance is purely obtained by sorting the elements on $\mathbb{R}$ in the discrete case. In particular, this means that the transport plan are exactly the same for all W_q, $q \in [1, +\infty)$. Thus the notion of a robust barycenter does not apply for the Sliced Wasserstein distance.

5.6 Recommendation & summary

In this chapter, we have studied several sampling strategies on the sphere for computing an approximation of the Sliced Wasserstein distance.

Regarding theoretical guarantees, this study highlighted the following limitations. The classical i.i.d. sampling benefits from theoretical guarantees with a convergence rate in $\mathcal{O}(1/\sqrt{M})$ and a time complexity linear in the number M of projections. Orthonormal sampling and L.D.S. such as Halton or Sobol lack convergence rate guarantees on the sphere (these guarantees being only obtained for sequences on hypercubes for L.D.S). As for deterministic point generation methods (like Riesz), the Sliced Wasserstein integrand also lacks sufficient regularity to guarantee results in dimensions higher than 2.

While lacking theoretical guarantees in terms of convergence, the experimental study suggests that Q.M.C methods (L.D.S. or s-Riesz points) provide competitive results in small to intermediate dimension, while having a similar convergence rate to classical random sampling methods in intermediate to higher (for Riesz) dimensions. These results seem to indicate that, while f is not regular enough for the convergence guarantees detailed in this chapter, there may be some non-proven convergence results requiring weaker regularity conditions that would be applicable to SW.

Now, considering computation times, as shown by Fig. 5.8 and Tab. 5.2, classical i.i.d. sampling remains the slowest method in all our experiments. While orthonormal sampling lacks theoretical guarantees, it seems to be one of the most efficient methods whatever the dimension, and is particularly competitive in high dimensions, with a very reasonable increase of computation time. L.D.S. methods also remain competitive in practice for small dimensions. s-Riesz points, while competitive in terms of convergence rate, have a prohibitive time complexity in $\mathcal{O}(M^2)$, which makes them completely unsuitable for a large number of projections.

The experiments also suggest that the S.H.C.V. method is very competitive in intermediate dimensions, while becoming less efficient when d increases.

Based on the different experimental results provided in this chapter, we make the following recommendations:

- For small dimensions (less than 3), Q.M.C. methods such as s-Riesz points or L.D.S. mapped onto the sphere can be privileged with respect to uniform sampling.
- For high dimensions (greater than 20), the orthonormal sampling method emerges as the most suitable choice. It is also one of the simplest methods to implement, which makes it particularly attractive in practice.
- For intermediate dimensions (between 5 and 10), choosing an appropriate method should depend on the experimental requirements. Spherical harmonics are an excellent option if computational resources are limited and if the number of SW distances to be computed is low. However, it is worth noting that some Q.M.C. strategies, being independent of the input measures, have the advantage of allowing the generated points to be reused and of allowing an independent computation in M (except the Riesz points). This should be particularly beneficial when a high number of projections is required and a large number of SW distances must be computed. In such cases, we suggest to store the samples to factorize the computing time across experiments.

Chapter 6

Conclusion and perspectives

In this thesis, we developed and studied an analysis and lossy compression method for ensembles of topological representations, based on barycenters. The goal of this method is to facilitate the analysis of ensembles of scalar fields using their topological representations, while simultaneously addressing the main challenges posed by modern data: increasing size and geometrical complexity. These topological representations already address these challenges by encapsulating the topological information concisely. This work extends this approach by finding a compact representation of topological representations while preserving the analysis results of scalar field ensembles. These results are presented visually to end users, providing clear, simple, and concise insights about a data ensemble.

6.1 Summary of contributions

The contributions of this thesis propose solutions for finding concise representations of an ensemble of topological representations, ranging from the conception of the method, to providing an extension for stability reasons, and finally offering an alternative to address computational time considerations. All contributions presented in this thesis are implemented either in the open-source library Topology ToolKit [184] or in Python [69].

6.1.1 Wasserstein Dictionaries of Persistence Diagrams

A Wasserstein barycenter is a well-known *representative* of an ensemble of persistence diagrams. However, while it provides global information on the main structures in an ensemble of persistence diagrams, it cannot describe the variability within the ensemble. We addressed this problem in Chapt. 3 by proposing a Wasserstein dictionary encoding method, which consists of finding a smaller set of *representative* diagrams and barycentric coefficients to generate barycenters that approximate each member of an ensemble of persistence diagrams. Each barycenter should share the same topological pattern profile as a member of the input ensemble of persistence diagrams. We first proposed a simple optimization algorithm, based on alternating gradient descent with gradient steps updated automatically. However, the optimization problem is neither convex nor continuous, leading to instabilities without further improvements.

We overcame these obstacles by introducing a persistence-based multi-scaling approach. This approach allows our method to focus heavily on the high-persistence pairs,

thereby biasing the gradient descent toward a convex region. It also accelerates the initial algorithm and generally produces better results.

We demonstrated the utility of this method with applications such as data reduction and dimensionality reduction.

6.1.2 Robust Barycenters of Persistence Diagrams

While widely used for its descriptive power and its ability to summarize the topological profile of an ensemble of persistence diagrams, a Wasserstein barycenter can be sensitive to the presence of outliers in the ensemble. Indeed, a Wasserstein barycenter—also called a Fréchet mean—behaves like an arithmetic mean and, as such, inherits the drawback of being sensitive to outliers. Consequently, this sensitivity can introduce optimization difficulties in our Wasserstein dictionary method. To address this problem, we proposed using an alternative framework to compute a Wasserstein barycenter with general function costs. This new framework operates as a fixed-point algorithm and produces a barycenter that is robust to the presence of outliers in an ensemble of persistence diagrams. We demonstrated the robustness of these barycenters through applications to classical clustering problems of persistence diagrams and by incorporating this robust barycenter into our Wasserstein dictionary encoding method.

6.1.3 A User's Guide to Sampling Strategies for Sliced Optimal Transport

Another downside of a Wasserstein barycenter is its computational time bottleneck. This arises from the fact that the Wasserstein distance has a runtime complexity of $\mathcal{O}\big(N^3 \log(N)\big)$, where N is the size of the two persistence diagrams. To address this, we considered an alternative called the Sliced Wasserstein distance, defined as the average of 1D Wasserstein distances of measures projected onto lines sampled from the hypersphere. In practice, this distance is usually computed using a Monte Carlo method and has a runtime complexity of $\mathcal{O}\big(MN \log(N)\big)$, where M is the number of lines sampled from the hypersphere. We provided a general user guide to different sampling strategies for Monte Carlo and quasi-Monte Carlo methods to compute the Sliced Wasserstein distance, including comparisons of theoretical and experimental results. In particular, we demonstrated that computing the Sliced Wasserstein distance between persistence diagrams—and, consequently, computing a Wasserstein barycenter—is both simple and fast. We also detailed how this Sliced Wasserstein barycenter can be applied to the dictionary encoding problem for an ensemble of persistence diagrams.

6.2 Discussion

Limitations regarding the different contributions of this thesis were already presented in the dedicated chapters. However, we would like to further detail some limitations of our work and possible solutions to address them.

A crucial limitation of our first work is the non-convexity and non-differentiability of the functional optimized for the dictionary encoding of persistence diagrams (Chapt. 3). These characteristics of the functional led us to introduce a multi-scaling approach to our

original algorithm. Another possible way to address this issue would be to use a regularized version of the optimization problem, as in [167]. This regularization, called the entropic optimal transport problem, is usually solved using the Sinkhorn algorithm [48,104]. However, this regularization can only be applied to histograms, meaning that the persistence diagrams must be vectorized [4,104]. While this regularization method can simplify the optimization scheme and reduce computational time, it depends on parameters, and vectorized persistence diagrams cannot be visually interpreted or analyzed.

Another approach would be to fix the optimal transport plans during the optimization scheme. At fixed transport plans, the energy is locally convex and differentiable (even $\mathcal{C}^\infty$) (Appendix A and Appendix B), ensuring a steady descent toward a local minimum. However, this method can lead to a *"bad"* local minimum, as fixing the transport plans prevents the optimization algorithm from exploring other locally convex regions. A further alternative would be to use a stochastic gradient descent method, typically employed when the functional is not convex. Such a method allows the optimizer to explore other regions and potentially find a better local minimum than the one reached initially. Unfortunately, in our case, the computation of new barycenters between each gradient step already changes the optimal transport plan, effectively simulating the same behavior as stochastic gradient descent.

Another limitation concerns the time and memory bottlenecks of computing the robust barycenter (Chapt. 4). Indeed, the computation of this barycenter still relies on Wasserstein distances with runtime complexities of $\mathcal{O}\big(N^3 \log(N)\big)$. Moreover, our implementation in PyTorch requires a GPU with sufficient dedicated VRAM, meaning that computing robust barycenters of persistence diagrams with a large number of points ($>$10,000) can be challenging. One way to alleviate these bottlenecks would be to introduce a persistence-driven progressive approach, as in [173, 191]. A persistence-based progressive approach focuses solely on the high-persistence pairs of a persistence diagram. These high-persistence pairs, which generally represent the important topological features, are typically present in small numbers ($<$50). Thus, computation could focus on the high-persistence pairs while providing a rough approximation of the barycenter for the low-persistence pairs. However, as we observed in Chapt. 3, the main structures of interest can sometimes be encoded in the small-persistence pairs, while noise in the original data can appear in high-persistence pairs. Although such situations are uncommon, they have been reported in the literature [4, 56].

6.3 Perspectives

6.3.1 Generalization to other topological representations

A natural direction for future work is the extension of the methods explored in this thesis to other topological representations such as merge trees, contour trees, Reeb graphs, or Morse-Smale complexes. These topological representations capture information that is not represented by persistence diagrams, so generalizing the methods explored here could lead to improved results.

First, a generalization of the robust barycenter framework to merge trees seems feasible. Indeed, [149] already defined a Wasserstein distance between merge trees, and consequently introduced a notion of Wasserstein barycenters for merge trees. This Wasserstein

distance is analogous to the 2-Wasserstein distance between persistence diagrams. Therefore, using a Wasserstein distance with a generic transportation cost should be feasible in theory and in practice, and defining a robust barycenter for merge trees could be a natural future result. Naturally, since Wasserstein barycenters of merge trees are already defined, an extension of our Wasserstein dictionary framework to merge trees could also be possible. As methods for finding concise representations of ensembles of merge trees are already established [112, 150], we believe that adapting our Wasserstein dictionary encoding framework to merge trees should be achievable.

However, the same cannot be said for the other topological descriptors, as defining informative and computable metrics remains an open research problem. One possibility for contour trees would be to extend the edit distance [99]. Yet, defining such a metric does not guarantee the existence of computable transport plans or barycenters, which are necessary for our Wasserstein dictionary framework.

6.3.2 Theoretical studies on the convex areas of the functional in the Wasserstein dictionary framework

As discussed earlier, one of the main limitations we encountered was the non-differentiability and non-convexity of the functional optimized in Chapt. 3. Unfortunately, we could not fully characterize the regions of differentiability of this functional, as this is not a central topic in the context of this thesis. However, based on several experiments run on toy datasets, we observed certain structures in the birth/death space when the number of atoms is equal to three. We know that the lack of differentiability is caused by changes in the optimal transport plans between the barycenters and the atoms in the dictionary. In other words, the triangle from which each point of the barycenter originates changes.

From our observations, we hypothesize that the Delaunay triangulation [51] provides the structure subdividing the birth/death space, delimiting regions where the optimal transport plans remain constant. Based on this assumption, one idea would be to compute the dual Voronoi diagram [193], color each triangle according to the corresponding optimal transport plan, then select the color yielding the best energy after each gradient step before updating the barycenter. An obvious drawback of this approach is its runtime complexity, since computing a Delaunay triangulation and a Voronoi diagram is computationally expensive. Thus, for large persistence diagrams this method is not feasible, but for small persistence diagrams it could be a potential solution. Of course, this is based solely on the assumption that the underlying structure is indeed a Delaunay triangulation.

6.3.3 Further applications of the Wasserstein dictionary framework

State-of-the-art methods for classical supervised dictionary encoding have several applications, such as inpainting and denoising. In other words, dictionary-based representations can be used for image or video restoration. The Wasserstein dictionary framework, originally established on histograms by [167] and inspired by these methods, could similarly be used to restore persistence diagrams that are missing in a time series. Recent methods based on dictionary encoding have also been developed for this purpose, such as deep dictionary learning [169]. The principle of this method is to combine the concept of dictionary

representation with deep networks, by using a dictionary as a neuron within a multi-layer network. Such an approach could potentially be adapted to persistence diagrams. For instance, classical auto-encoders have already been extended to merge trees (and persistence diagrams) by [147], by generalizing their principal geodesic analysis framework [150]. Thus, we assume that a similar generalization of the Wasserstein dictionary framework to persistence diagrams is feasible. However, one obvious drawback of such a method would be its computational cost, as noted in [147].

Appendices

Appendix A: Dimensionality reduction

Fig. .1 extends Fig. 3.7 to all our test ensembles and it confirms visually the conclusions of the table of quality scores (Table 3).

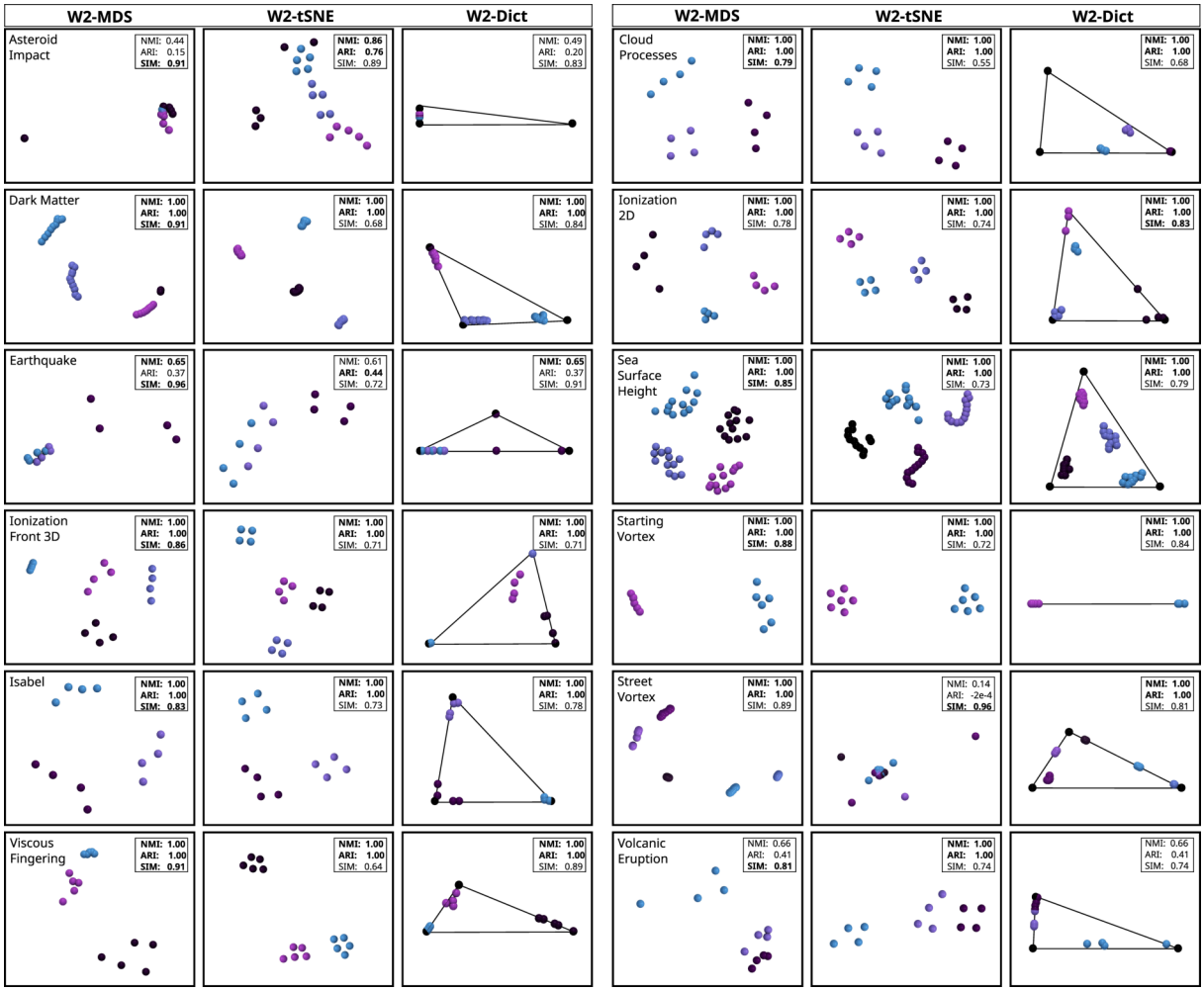
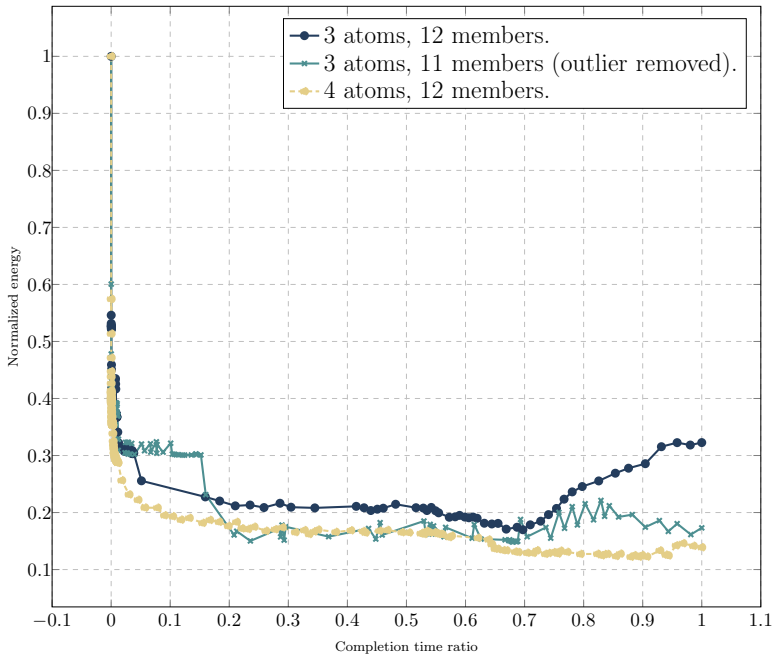


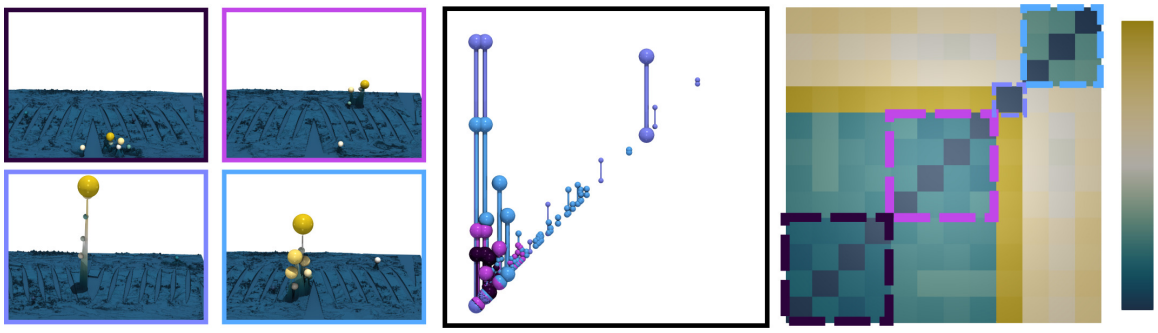
Figure .1: Comparison of the planar layouts for typical dimensionality reduction techniques on all our test ensembles. The color encodes the classification ground-truth [149]. For each quality score, the best value appears bold. For the *Sea Surface Height* ensemble, the naive optimization procedure has been used (cf. Sec. 6.3).

Appendix B: Volcanic eruption ensemble

This appendix discusses the special case of the *Volcanic eruption* ensemble (12 members), for which a consistent energy increase can be observed in the Figure 13 of the main manuscript (normalized energy of our multi-scale optimization as a function of computation time), beyond 70% of the completion time (the optimization reaches the stopping conditions at 100%).



(a)



(b)

Figure .2: Evolution of the (normalized) energy E_D along the optimization (top curves), with our multi-scale strategy, for the *Volcanic eruption ensemble*, for distinct initializations. The ground-truth classification of this ensemble contains 3 classes [149], including one outlier (light purple entry in the bottom views, from left to right: terrain view of the data, aggregated birth/death space, distance matrix). A clear energy increase can be observed when considering the entire ensemble (black curve), while a more characteristic oscillating behavior occurs when discarding the outlier (green curve). When initializing the optimization with 4 atoms (1 per class, plus 1 for the outlier), the optimization results in few oscillations and a consistent energy decrease (yellow curve).

The ground-truth classification of this ensemble contains 3 classes [149]. However, one of these classes contains a clear outlier (light purple entry in the bottom views of Fig. .2), corresponding to a peak of activity in the eruption (see the terrain views of 4 members, bottom left of Fig. .2, including the outlier, light purple frame). The corresponding persistence diagram (light purple diagram in the aggregated birth/death space, bottom middle of Fig. .2) contains features which are significantly more persistent than the other diagrams (taken from distinct ground-truth classes, one color per class). Then, this outlier exhibits an excessively high distance to the rest of the ensemble, as illustrated in the Wassertein distance matrix (bottom right of Fig. .2, light purple entry).

The presence of this outlier challenges our optimization when using a number of atoms equal to the number of ground-truth classes (which is the default strategy documented in the main manuscript). As shown in the energy plots (Fig. .2, top), a consistent energy increase can be observed when using only 3 atoms (1 per ground-truth class, black curve). When removing the outlier, the energy evolution exhibits a more characteristic oscillating behavior (green curve). Finally, when initializing the optimization with 4 atoms (1 per class, plus 1 for the outlier), the optimization results in few oscillations and a consistent energy decrease (yellow curve). This indicates that the outlier member (light purple) should be interpreted as a singleton class and that the best dictionary encoding will consequently be obtained with 4 atoms.

Appendix C: Explicit computation of $SW(\mu, \nu)$ in a simple case

We define $\mu = \frac{1}{2}[\delta_{x_1} + \delta_{x_2}]$ and $\nu = \frac{1}{2}[\delta_{y_1} + \delta_{y_2}]$, with $x_1, x_2 = (1,0,0)^T, (0,-1,0)^T$ and $y_1, y_2 = (0,0,1)^T, (0,0,-1)^T$. Now let us find the four sections (as in Fig. 5.2), for which the ordering of $\langle\theta, x_1\rangle$ with $\langle\theta, x_2\rangle$ and $\langle\theta, y_1\rangle$ with $\langle\theta, y_2\rangle$ is fixed.

Let us write $\theta = \begin{bmatrix} \sin(\varphi)\cos(\chi) \\ \sin(\varphi)\sin(\chi) \\ \cos(\varphi) \end{bmatrix}$ in spherical coordinates with $\varphi \in [0, \pi]$ and $\chi \in [0, 2\pi]$.

Let $\theta \in \mathbb{S}^{d-1}$, we have

$$\begin{aligned} \langle\theta, x_1\rangle \geq \langle\theta, x_2\rangle &\iff \sin(\varphi)\big(\cos(\chi) + \sin(\chi)\big) \geq 0 \iff \sin(\varphi) \geq 0 \text{ and } \cos(\chi) + \sin(\chi) \geq 0 \\ &\iff \chi \in [0, 3\pi/4] \cup [7\pi/4, 2\pi] = A, \end{aligned}$$

and $\langle\theta, y_1\rangle \geq \langle\theta, y_2\rangle \iff \varphi \in [0, \pi/2] = B$. Denoting A^c the complement of A in $[0, 2\pi]$ and B^c the complement of B in $[0, \pi]$, the Sliced Wasserstein distance simply writes:

$$\begin{aligned} SW_2^2(\mu, \nu) = &\frac{1}{8\pi} \int_{A^c} \int_{B^c} \big(\langle\theta, x_1 - y_1\rangle^2 + \langle\theta, x_2 - y_2\rangle^2\big) \sin(\varphi) d\varphi d\chi \\ &+ \frac{1}{8\pi} \int_{A^c} \int_{B} \big(\langle\theta, x_1 - y_2\rangle^2 + \langle\theta, x_2 - y_1\rangle^2\big) \sin(\varphi) d\varphi d\chi \\ &+ \frac{1}{8\pi} \int_{A} \int_{B^c} \big(\langle\theta, x_2 - y_1\rangle^2 + \langle\theta, x_1 - y_2\rangle^2\big) \sin(\varphi) d\varphi d\chi \end{aligned}$$

$$+ \frac{1}{8\pi} \int_A \int_B \left(\langle \theta, x_2 - y_2 \rangle^2 + \langle \theta, x_1 - y_1 \rangle^2 \right) \sin(\varphi) d\varphi d\chi.$$

From there, simple integral computations yield $SW_2^2(\mu, \nu) = \frac{2(\pi - \sqrt{2})}{3\pi}$.

Bibliography

[1] ISO/IEC Guide 98-3:2008 uncertainty of measurement - part 3: Guide to the expression of uncertainty in measurement (GUM). 2008.

[2] A. Acharya and V. Natarajan. A parallel and memory efficient algorithm for constructing the contour tree. In *IEEE PacificViz*, 2015.

[3] H. Adams, S. Chepushtanova, T. Emerson, E. Hanson, M. Kirby, F. Motta, R. Neville, C. Peterson, P. Shipman, and L. Ziegelmeier. Persistence images: A stable vector representation of persistent homology, 2016.

[4] H. Adams, S. Chepushtanova, T. Emerson, E. Hanson, M. Kirby, F. Motta, R. Neville, C. Peterson, P. Shipman, and L. Ziegelmeier. Persistence Images: A Stable Vector Representation of Persistent Homology. *Journal of Machine Learning Research*, 18(8):1–35, 2017.

[5] M. Agueh and G. Carlier. Barycenters in the wasserstein space. *SIAM Journal on Mathematical Analysis*, 43(2):904–924, 2011.

[6] J. Ahrens, B. Geveci, and C. Law. Paraview: An end-user tool for large-data visualization. *The Visualization Handbook*, pages 717–731, 2005.

[7] C. Aistleitner, J. S. Brauchart, and J. Dick. Point sets on the sphere $\mathbb{S}^2$ with small spherical cap discrepancy. *Discrete and Computational Geometry*, Aug. 2012.

[8] C. Aistleitner, F. Pausinger, A. M. Svane, and R. F. Tichy. On functions of bounded variation, 2016.

[9] K. Anderson, J. Anderson, S. Palande, and B. Wang. Topological data analysis of functional MRI connectivity in time and space domains. In *MICCAI Workshop on Connectomics in NeuroImaging*, 2018.

[10] R. Anirudh, V. Venkataraman, K. N. Ramamurthy, and P. K. Turaga. A Riemannian Framework for Statistical Analysis of Topological Persistence Diagrams. In *IEEE CVPR Workshops*, 2016.

[11] G. B. Arfken, H. J. Weber, and F. E. Harris. *Mathematical Methods for Physicists: A Comprehensive Guide*. Academic Press, 2011.

[12] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[13] S. Asmussen and P. Glynn. *Stochastic simulation: Algorithms and analysis*. Springer Verlag, 2007.

[14] T. Athawale and A. Entezari. Uncertainty quantification in linear interpolation for isosurface extraction. *IEEE TVCG*, 2013.

[15] T. Athawale, E. Sakhaee, and A. Entezari. Isosurface visualization of data with nonparametric models for uncertainty. *IEEE TVCG*, 2016.

[16] T. M. Athawale, D. Maljovec, C. R. Johnson, V. Pascucci, and B. Wang. Uncertainty Visualization of 2D Morse Complex Ensembles Using Statistical Summary Maps. *CoRR*, abs/1912.06341, 2019.

[17] U. Ayachit, A. C. Bauer, B. Geveci, P. O'Leary, K. Moreland, N. Fabian, and J. Mauldin. ParaView Catalyst: Enabling In Situ Data Analysis and Visualization. In *ISAV*, 2015.

[18] K. Basu. *Quasi-Monte Carlo Methods in Non-Cubical Spaces*. Phd thesis, Stanford University, 2016.

[19] D. P. Bertsekas. A new algorithm for the assignment problem. *Mathematical Programming*, 21(1):152–171, 1981.

[20] H. Bhatia, A. G. Gyulassy, V. Lordi, J. E. Pask, V. Pascucci, and P.-T. Bremer. Topoms: Comprehensive topological exploration for molecular and condensed-matter systems. *J. of Comp. Chem.*, 2018.

[21] H. Bhatia, S. Jadhav, P. Bremer, G. Chen, J. A. Levine, L. G. Nonato, and V. Pascucci. Flow visualization with quantified spatial and temporal errors using edge maps. *IEEE TVCG*, 2012.

[22] S. Biasotti, D. Giorgio, M. Spagnuolo, and B. Falcidieno. Reeb graphs for shape analysis and applications. *TCS*, 2008.

[23] T. Bin Masood, J. Budin, M. Falk, G. Favelier, C. Garth, C. Gueunet, P. Guillou, L. Hofmann, P. Hristov, A. Kamakshidasan, C. Kappe, P. Klacansky, P. Laurin, J. Levine, J. Lukasczyk, D. Sakurai, M. Soler, P. Steneteg, J. Tierny, W. Usher, J. Vidal, and M. Wozniak. An Overview of the Topology ToolKit. In *TopoInVis*, 2019.

[24] A. Bock, H. Doraiswamy, A. Summers, and C. T. Silva. TopoAngler: Interactive Topology-Based Extraction of Fishes. *IEEE TVCG*, 2018.

[25] C. Bonet, P. Berg, N. Courty, F. Septier, L. Drumetz, and M.-T. Pham. Spherical sliced-wasserstein, 2023.

[26] C. Bonet, L. Drumetz, and N. Courty. Sliced-wasserstein distances and flows on cartan-hadamard manifolds. *arXiv preprint arXiv:2403.06560*, 2024.

[27] G. Bonneau, H. Hege, C. Johnson, M. Oliveira, K. Potter, P. Rheingans, and T. Schultz". Overview and state-of-the-art of uncertainty visualization. *Mathematics and Visualization*, 37:3–27, 2014.

[28] N. Bonneel, J. Rabin, G. Peyré, and H. Pfister. Sliced and Radon Wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45, 2015.

[29] N. Bonnotte. Unidimensional and evolution methods for optimal transportation. PhD thesis, 2013.

[30] L. Brandolini, L. Colzani, G. Gigante, and G. Travaglini. On the koksma–hlawka inequality. *Journal of Complexity*, 29(2):158–172, 2013.

[31] J. S. Brauchart. Optimal discrete riesz energy and discrepancy, 2011.

[32] J. S. Brauchart and J. Dick. A characterization of sobolev spaces on the sphere and an extension of stolarsky's invariance principle to arbitrary smoothness. *Constructive Approximation*, 38(3):397–445, Oct. 2013.

[33] P. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A Multi-Resolution Data Structure for 2-Dimensional Morse Functions. In *Proc. of IEEE VIS*, 2003.

[34] P. Bremer, G. Weber, J. Tierny, V. Pascucci, M. Day, and J. Bell. Interactive exploration and analysis of large scale simulations using topology-based data segmentation. *IEEE TVCG*, 2011.

[35] M. R. Bridson and A. Haefliger. *Metric spaces of non-positive curvature / Martin R. Bridson, André Haefliger*. Grundlehren der mathematischen Wissenschaften a series of comprehensive studies in mathematics. Springer, Berlin [etc, 1999.

[36] N. Brown, R. Nash, P. Poletti, G. Guzzetta, M. Manica, A. Zardini, M. Flatken, J. Vidal, C. Gueunet, E. Belikov, J. Tierny, A. Podobas, W. D. Chien, S. Markidis, and A. Gerndt. Utilising urgent computing to tackle the spread of mosquito-borne diseases. In *UrgentHPC@SC*, 2021.

[37] P. Bubenik. Statistical topological data analysis using persistence landscapes. *J. Mach. Learn. Res.*, 16:77–102, 2015.

[38] H. Carr, J. Snoeyink, and U. Axen. Computing contour trees in all dimensions. In *Symp. on Dis. Alg.*, 2000.

[39] H. Carr, G. Weber, C. Sewell, and J. Ahrens. Parallel peak pruning for scalable SMP contour tree computation. In *IEEE LDAV*, 2016.

[40] H. A. Carr, J. Snoeyink, and M. van de Panne. Simplifying Flexible Isosurfaces Using Local Geometric Measures. In *IEEE VIS*, 2004.

[41] M. Carrière, M. Cuturi, and S. Oudot. Sliced wasserstein kernel for persistence diagrams, 2017.

[42] M. E. Celebi, H. A. Kingravi, and P. A. Vela. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Syst. Appl.*, 2013.

[43] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu. Shapenet: An information-rich 3d model repository, 2015.

[44] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. In *SoCG*, 2005.

[45] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 2007.

[46] D. Cohen-Steiner, H. Edelsbrunner, J. Harer, and Y. Mileyko. Lipschitz Functions Have Lp-Stable Persistence. *FCM*, 2010.

[47] N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2016.

[48] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NIPS*. 2013.

[49] M. Cuturi and A. Doucet. Fast computation of wasserstein barycenters. In *ICML*, 2014.

[50] L. De Floriani, U. Fugacci, F. Iuricich, and P. Magillo. Morse complexes for shape segmentation and homological analysis: discrete models and algorithms. *CGF*, 2015.

[51] B. Delaunay. Sur la sphere vide. *Bulletin de l'Acadeemie des Sciences de l'URSS*, (6):793–800, 1934.

[52] J. Delon, J. Rabin, and Y. Gousseau. Transportation distances on the circle and applications, 2010.

[53] I. Deshpande, Z. Zhang, and A. G. Schwing. Generative modeling using the sliced wasserstein distance. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3483–3491, 2018.

[54] J. Dick and F. Pillichshammer. *Digital nets and sequences: discrepancy theory and quasi–Monte Carlo integration*. Cambridge University Press, 2010.

[55] P. Diggle, P. Heagerty, K.-Y. Liang, and S. Zeger. *The Analysis of Longitudinal Data*. Oxford University Press, 2002.

[56] V. Divol and W. Polonik. On the choice of weight functions for linear representations of persistence diagrams. *Journal of Applied and Computational Topology*, 3(3):249–283, Aug. 2019.

[57] H. Doraiswamy and V. Natarajan. Computing Reeb Graphs as a Union of Contour Trees. *IEEE TVCG*, 2013.

[58] H. Edelsbrunner and J. Harer. *Computational Topology: An Introduction*. American Mathematical Society, 2009.

[59] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Morse-Smale complexes for piecewise linear 3-manifolds. In *SoCG*, 2003.

[60] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical morse complexes for piecewise linear 2-manifolds. In *SoCG*, 2001.

[61] A. Elnekave and Y. Weiss. Generating natural images with direct patch distributions matching. 03 2022.

[62] G. Favelier, N. Faraj, B. Summa, and J. Tierny. Persistence Atlas for Critical Point Variability in Ensembles. *IEEE TVCG*, 2018.

[63] G. Favelier, C. Gueunet, and J. Tierny. Visualizing ensembles of viscous fingers. In *IEEE SciVis Contest*, 2016.

[64] R. Feng, F. Götze, and D. Yao. Determinantal point processes on spheres: multivariate linear statistics, 2023.

[65] F. Ferstl, K. Bürger, and R. Westermann. Streamline variability plots for characterizing the uncertainty in vector field ensembles. *IEEE TVCG*, 2016.

[66] F. Ferstl, M. Kanzler, M. Rautenhaus, and R. Westermann. Visual analysis of spatial variability and global correlations in ensembles of iso-contours. *CGF*, 2016.

[67] J. Feydy, B. Charlier, F.-X. Vialard, and G. Peyré. Optimal transport for diffeomorphic registration. In *Medical Image Computing and Computer Assisted Intervention-MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part I 20*, pages 291–299. Springer, 2017.

[68] H. Fischer. *A History of the Central Limit Theorem: From Classical to Modern Probability Theory*. Sources and Studies in the History of Mathematics and Physical Sciences. Springer New York, 2010.

[69] R. Flamary, N. Courty, A. Gramfort, M. Z. Alaya, A. Boisbunon, S. Chambon, L. Chapel, A. Corenflos, K. Fatras, N. Fournier, L. Gautheron, N. T. Gayraud, H. Janati, A. Rakotomamonjy, I. Redko, A. Rolet, A. Schutz, V. Seguy, D. J. Sutherland, R. Tavenard, A. Tong, and T. Vayer. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021.

[70] P. Fletcher, L. Conglin, S. Pizer, and J. Sarang. Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE Transactions on Medical Imaging*, 23(8):995–1005, 2004.

[71] R. Forman. A User's Guide to Discrete Morse Theory. *AM*, 1998.

[72] D. Guenther, R. Alvarez-Boto, J. Contreras-Garcia, J.-P. Piquemal, and J. Tierny. Characterizing Molecular Interactions in Chemical Systems. *IEEE TVCG*, 2014.

[73] C. Gueunet, P. Fortin, J. Jomier, and J. Tierny. Task-Based Augmented Contour Trees with Fibonacci Heaps. *IEEE TPDS*, 2019.

[74] C. Gueunet, P. Fortin, J. Jomier, and J. Tierny. Task-based Augmented Reeb Graphs with Dynamic ST-Trees. In *EGPGV*, 2019.

[75] P. Guillou, J. Vidal, and J. Tierny. Discrete morse sandwich: Fast computation of persistence diagrams for scalar data–an algorithm and a benchmark. *IEEE TVCG*, 2023.

[76] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017.

[77] D. Günther, J. Salmon, and J. Tierny. Mandatory critical points of 2D uncertain scalar fields. *CGF*, 2014.

[78] A. Gyulassy, P. Bremer, R. Grout, H. Kolla, J. Chen, and V. Pascucci. Stability of dissipation elements: A case study in combustion. *CGF*, 2014.

[79] A. Gyulassy, P. Bremer, and V. Pascucci. Shared-Memory Parallel Computation of Morse-Smale Complexes with Improved Accuracy. *IEEE TVCG*, 2019.

[80] A. Gyulassy, M. A. Duchaineau, V. Natarajan, V. Pascucci, E. Bringa, A. Higginbotham, and B. Hamann. Topologically Clean Distance Fields. *IEEE TVCG*, 2007.

[81] A. Gyulassy, A. Knoll, K. Lau, B. Wang, P. Bremer, M. Papka, L. A. Curtiss, and V. Pascucci. Interstitial and Interlayer Ion Diffusion Geometry Extraction in Graphitic Nanosphere Battery Materials. *IEEE TVCG*, 2016.

[82] M. Götz. On the riesz energy of measures. *Journal of Approximation Theory*, 122(1):62–78, 2003.

[83] J. H. Halton. Algorithm 247: Radical-inverse quasi-random point sequence. *Communications of the ACM*, 7(12):701–702, 1964.

[84] J. Hammersley and D. Handscomb. *Monte Carlo Methods*. Methuen's monographs on applied probability and statistics. Methuen, 1964.

[85] D. P. Hardin, T. J. Michaels, and E. B. Saff. A comparison of popular point configurations on $\mathbb{S}^2$, 2016.

[86] A. Hatcher. *Algebraic topology*. Cambridge University Press, 2002.

[87] E. Hebey. *Sobolev Spaces on Riemaniann Manifolds*. Springer, 1996.

[88] C. Heine, H. Leitte, M. Hlawitschka, F. Iuricich, L. De Floriani, G. Scheuermann, H. Hagen, and C. Garth. A survey of topology-based methods in visualization. *CGF*, 2016.

[89] E. Heitz, K. Vanhoey, T. Chambon, and L. Belcour. A sliced wasserstein loss for neural texture synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9412–9420, 2021.

[90] J. Hertrich, A. Houdard, and C. Redenbach. Wasserstein patch prior for image superresolution. *IEEE Transactions on Computational Imaging*, 8:693–704, 2022.

[91] M. Hummel, H. Obermaier, C. Garth, and K. I. Joy. Comparative visual analysis of lagrangian transport in CFD ensembles. *IEEE TVCG*, 2013.

[92] C. R. Johnson and A. R. Sanderson. A next step: Visualizing errors and uncertainty. *IEEE CGA*, 2003.

[93] S. Jung. Geodesic projection of the von mises–fisher distribution for projection pursuit of directional data. *Electronic Journal of Statistics*, 15, 01 2021.

[94] S. Jung, I. L. Dryden, and J. S. Marron. Analysis of principal nested spheres. *Biometrika*, 99(3):551–568, 07 2012.

[95] L. Kantorovich. On the translocation of masses. *AS USSR*, 1942.

[96] J. Kasten, J. Reininghaus, I. Hotz, and H. Hege. Two-dimensional time-dependent vortex regions based on the acceleration magnitude. *IEEE TVCG*, 2011.

[97] M. Kerber, D. Morozov, and A. Nigmetov. Geometry helps to compare persistence diagrams. *ACM J. of Experimental Algorithmics*, 22, 2017.

[98] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017.

[99] P. N. Klein. *Computing the Edit-Distance Between Unrooted Ordered Trees*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[100] S. Kolouri, P. E. Pope, C. E. Martin, and G. K. Rohde. Sliced wasserstein auto-encoders. In *International Conference on Learning Representations*, 2018.

[101] M. Kraus. Visualization of uncertain contour trees. In *IVTA*, 2010.

[102] J. B. Kruskal and M. Wish. Multidimensional Scaling. In *SUPS*, 1978.

[103] L. Kuipers and H. Niederreiter. *Uniform distribution of sequences*. Courier Corporation, 2012.

[104] T. Lacombe, M. Cuturi, and S. Oudot. Large Scale computation of Means and Clusters for Persistence Diagrams using Optimal Transport. In *NIPS*, 2018.

[105] D. E. Laney, P. Bremer, A. Mascarenhas, P. Miller, and V. Pascucci. Understanding the structure of the turbulent mixing layer in hydrodynamic instabilities. *IEEE TVCG*, 2006.

[106] T. Le, K. Nguyen, S. Sun, N. Ho, and X. Xie. Integrating efficient optimal transport and functional maps for unsupervised shape correspondence learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23188–23198, 2024.

[107] Y. LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

[108] C.-Y. Lee, T. Batra, M. H. Baig, and D. Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10285–10295, 2019.

[109] R. Leluc, A. Dieuleveut, F. Portier, J. Segers, and A. Zhuman. Sliced-wasserstein estimation with spherical harmonics as control variates, 2024.

[110] C. Lemieux. Monte carlo and quasi-monte carlo sampling. 2009.

[111] C. Leruste. *Topologie algébrique : une introduction, et au delá*. Mathématiques en devenir. Calvage & Mounet, Paris, 2017.

[112] M. Li, S. Palande, L. Yan, and B. Wang. Sketching merge trees for scientific visualization. In *IEEE TopoInVis*, 2023.

[113] T. Liebmann and G. Scheuermann. Critical Points of Gaussian-Distributed Scalar Fields on Simplicial Grids. *CGF*, 2016.

[114] X. Liu, Y. Bai, R. D. Martín, K. Shi, A. Shahbazi, B. A. Landman, C. Chang, and S. Kolouri. Linear spherical sliced optimal transport: A fast metric for comparing spherical data, 2024.

[115] S. Maadasamy, H. Doraiswamy, and V. Natarajan. A hybrid parallel algorithm for computing and tracking level set topology. In *HiPC*, 2012.

[116] A. Maceachren, A.Robinson, S. Hopper, S. Gardner, R. Murray, M. Gahegan, and E. Hetzler. Visualizing geospatial information uncertainty: What we know and what we need to know. *CGIS*, 2005.

[117] D. Maljovec, B. Wang, P. Rosen, A. Alfonsi, G. Pastore, C. Rabiti, and V. Pascucci. Topology-inspired partition-based sensitivity analysis and visualization of nuclear simulations. In *IEEE PacificViz*, 2016.

[118] R. Marques. *Bayesian and Quasi-Monte Carlo spherical integration for global illumination*. Phd thesis, Université de Rennes, 2013.

[119] J. Marzo and A. Mas. Discrepancy of minimal riesz energy points. *Constructive Approximation*, 54(3):473–506, 2021.

[120] M. Mirzargar, R. Whitaker, and R. Kirby. Curve boxplot: Generalization of boxplot for ensembles of curves. *IEEE TVCG*, 20(12):2654–2663, 2014.

[121] G. Monge. Mémoire sur la théorie des déblais et des remblais. *Académie Royale des Sciences de Paris*, 1781.

[122] C. Müller. *Introduction*. Springer New York, New York, NY, 1998.

[123] J. Munkres. Algorithms for the assignment and transportation problems. *J. of SIAM*, 1957.

[124] J. R. Munkres. *Elements of Algebraic Topology*. Addison-Wesley, 1984.

[125] K. Nadjahi. *Sliced-Wasserstein distance for large-scale machine learning: theory, methodology and extensions*. PhD thesis, Institut polytechnique de Paris, 2021.

[126] F. Nauleau, F. Vivodtzev, T. Bridel-Bertomeu, H. Beaugendre, and J. Tierny. Topological Analysis of Ensembles of Hydrodynamic Turbulent Flows – An Experimental Study. In *IEEE LDAV*, 2022.

[127] K. Nguyen, N. Bariletto, and N. Ho. Quasi-monte carlo for 3d sliced wasserstein, 2024.

[128] K. Nguyen and N. Ho. Control variate sliced wasserstein estimators. 05 2023.

[129] K. Nguyen and N. Ho. Sliced wasserstein estimation with control variates, 2024.

[130] H. Niederreiter. Low-discrepancy and low-dispersion sequences. *Journal of number theory*, 30(1):51–70, 1988.

[131] M. Olejniczak, A. S. P. Gomes, and J. Tierny. A Topological Data Analysis Perspective on Non-Covalent Interactions in Relativistic Calculations. *International Journal of Quantum Chemistry*, 2019.

[132] M. Olejniczak and J. Tierny. Topological Data Analysis of Vortices in the Magnetically-Induced Current Density in LiH Molecule. *Physical Chemistry Chemical Physics*, 2023.

[133] Organizers. The IEEE SciVis Contest. `http://sciviscontest.ieeevis.org/`, 2004.

[134] M. Otto, T. Germer, H.-C. Hege, and H. Theisel. Uncertain 2D vector Field Topology. *CGF*, 2010.

[135] M. Otto, T. Germer, and H. Theisel. Uncertain topology of 3D vector fields. *IEEE PacificViz*, 2011.

[136] A. B. Owen. Multidimensional variation for quasi-monte carlo. In *Contemporary Multivariate Analysis And Design Of Experiments: In Celebration of Professor Kai-Tai Fang's 65th Birthday*, pages 49–74. World Scientific, 2005.

[137] A. B. Owen. Monte carlo book: the quasi-monte carlo parts, 2019.

[138] A. T. Pang, C. M. Wittenbrink, and S. K. Lodha. Approaches to uncertainty visualization. *The Visual Computer*, 1997.

[139] S. Parsa. A deterministic *o(m log m)* time algorithm for the reeb graph. In *SoCG*, 2012.

[140] V. Pascucci, G. Scorzelli, P. T. Bremer, and A. Mascarenhas. Robust on-line computation of Reeb graphs: simplicity and speed. *ACM ToG*, 2007.

[141] C. Petz, K. Pöthkow, and H.-C. Hege. Probabilistic local features in uncertain vector fields with spatial correlation. *CGF*, 2012.

[142] G. Peyré, M. Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[143] T. Pfaffelmoser, M. Mihai, and R. Westermann. Visualizing the variability of gradients in uncertain 2D scalar fields. *IEEE TVCG*, 2013.

[144] T. Pfaffelmoser, M. Reitinger, and R. Westermann. Visualizing the positional and geometrical variability of isosurfaces in uncertain scalar fields. *CGF*, 2011.

[145] T. Pfaffelmoser and R. Westermann. Visualization of global correlation structures in uncertain 2D scalar fields. *CGF*, 2012.

[146] H. Poincaré. Analysis situs. *Journal de l'École polytechnique*, 1:1–121, 1895.

[147] M. Pont and J. Tierny. Wasserstein auto-encoders of merge trees (and persistence diagrams), 2023.

[148] M. Pont, J. Vidal, J. Delon, and J. Tierny. Wasserstein Distances, Geodesics and Barycenters of Merge Trees – Ensemble Benchmark. `https://github.com/MatPont/WassersteinMergeTreesData`, 2021.

[149] M. Pont, J. Vidal, J. Delon, and J. Tierny. Wasserstein Distances, Geodesics and Barycenters of Merge Trees. *IEEE TVCG*, 2022.

[150] M. Pont, J. Vidal, and J. Tierny. Principal geodesic analysis of merge trees (and persistence diagrams). *IEEE TVCG*, 2023.

[151] K. Pöthkow and H.-C. Hege. Positional Uncertainty of Isocontours: Condition Analysis and Probabilistic Measures. *IEEE TVCG*, 2011.

[152] K. Pöthkow and H.-C. Hege. Nonparametric models for uncertainty visualization. *CGF*, 2013.

[153] K. Pöthkow, B. Weber, and H.-C. Hege. Probabilistic Marching Cubes. *CGF*, 2011.

[154] K. Potter, S. Gerber, and E. W. Anderson. Visualization of uncertainty without a mean. *IEEE Computer Graphics and Applications*, 2013.

[155] K. Potter, P. Rosen, and C. R. Johnson. From quantification to visualization: A taxonomy of uncertainty visualization approaches. *IFIP AICT*, 2012.

[156] K. Potter, A. Wilson, P. Bremer, D. Williams, C. Doutriaux, V. Pascucci, and C. R. Johnson. Ensemble-vis: A framework for the statistical visualization of ensemble data. In *2009 IEEE ICDM*, 2009.

[157] M. Quellmalz, R. Beinert, and G. Steidl. Sliced optimal transport on the sphere. *Inverse Problems*, 39(10):105005, Aug. 2023.

[158] J. Rabin, J. Delon, and Y. Gousseau. A statistical approach to the matching of local features. *SIAM Journal on Imaging Sciences*, 2(3):931–958, 2009.

[159] J. Rabin, G. Peyré, J. Delon, and M. Bernot. Wasserstein barycenter and its application to texture mixing. In *Scale Space and Variational Methods in Computer Vision: Third International Conference, SSVM 2011, Ein-Gedi, Israel, May 29–June 2, 2011, Revised Selected Papers 3*, pages 435–446. Springer, 2012.

[160] V. Robins and K. Turner. Principal Component Analysis of Persistent Homology Rank Functions with case studies of Spatial Point Patterns, Sphere Packing and Colloids. *Physica D: Nonlinear Phenomena*, 334:99–117, 2016.

[161] V. Robins, P. J. Wood, and A. P. Sheppard. Theory and Algorithms for Constructing Discrete Morse Complexes from Grayscale Digital Images. *IEEE PAMI*, 2011.

[162] M. Rowland, J. Hron, Y. Tang, K. Choromanski, T. Sarlos, and A. Weller. Orthogonal estimation of wasserstein distances, 2019.

[163] T. Salimans, H. Zhang, A. Radford, and D. Metaxas. Improving gans using optimal transport. In *International Conference on Learning Representations*, 2018.

[164] F. Santambrogio. Optimal transport for applied mathematicians. *Birkäuser, NY*, 55(58-63):94, 2015.

[165] J. Sanyal, S. Zhang, J. Dyer, A. Mercer, P. Amburn, and R. Moorhead. Noodles: A tool for visualization of numerical weather model ensemble uncertainty. *IEEE TVCG*, 2010.

[166] S. Schlegel, N. Korn, and G. Scheuermann. On the interpolation of data with normally distributed uncertainty for visualization. *IEEE TVCG*, 18(12):2305–2314, 2012.

[167] M. A. Schmitz, M. Heitz, N. Bonneel, F. Ngole, D. Coeurjolly, M. Cuturi, G. Peyré, and J.-L. Starck. Wasserstein dictionary learning: Optimal transport-based unsupervised nonlinear dictionary learning. *SIAM Journal on Imaging Sciences*, 2018.

[168] A. Schrijver. *Combinatorial Optimization - Polyhedra and Efficiency*. Springer, 2003.

[169] K. Seemakurthy, A. Majumdar, J. Gubbi, N. K. Sandeep, A. Varghese, S. Deshpande, M. Girish Chandra, and P. Balamurali. Deep dictionary learning for inpainting. In R. V. Babu, M. Prasanna, and V. P. Namboodiri, editors, *Computer Vision, Pattern Recognition, Image Processing, and Graphics*, pages 79–88, Singapore, 2020. Springer Singapore.

[170] N. Shivashankar and V. Natarajan. Parallel Computation of 3D Morse-Smale Complexes. *CGF*, 2012.

[171] N. Shivashankar, P. Pranav, V. Natarajan, R. van de Weygaert, E. P. Bos, and S. Rieder. Felix: A topology based framework for visual exploration of cosmic filaments. *IEEE TVCG*, 2016.

[172] R. Sinkhorn. Diagonal equivalence to matrices with prescribed row and column sums. *American Mathematical Monthly*, 1967.

[173] K. Sisouk, J. Delon, and J. Tierny. Wasserstein Dictionaries of Persistence Diagrams . *IEEE Transactions on Visualization & Computer Graphics*, 30(02):1638–1651, Feb. 2024.

[174] K. Sisouk, J. Delon, and J. Tierny. A User's Guide to Sampling Strategies for Sliced Optimal Transport. working paper or preprint, Feb. 2025.

[175] I. Sobol. The distribution of points in a cube and the accurate evaluation of integrals (in russian) zh. *Vychisl. Mat. i Mater. Phys*, 7:784–802, 1967.

[176] M. Soler, M. Petitfrere, G. Darche, M. Plainchault, B. Conche, and J. Tierny. Ranking Viscous Finger Simulations to an Acquired Ground Truth with Topology-Aware Matchings. In *IEEE LDAV*, 2019.

[177] T. Sousbie. The Persistent Cosmic Web and its Filamentary Structure: Theory and Implementations. *Royal Astronomical Society*, 2011.

[178] K. B. Stolarsky. Sums of distances between points on a sphere. ii. *Proceedings of the American Mathematical Society*, 41(2):575–582, 1973.

[179] A. Szymczak. Hierarchy of stable Morse decompositions. *IEEE TVCG*, 2013.

[180] E. Tanguy, J. Delon, and N. Gozlan. Computing barycentres of measures for generic transport costs. *arXiv preprint arXiv:2501.04016*, Dec. 2024.

[181] E. Tanguy, J. Delon, and N. Gozlan. Computing barycentres of measures for generic transport costs, 2025.

[182] E. Tanguy, R. Flamary, and J. Delon. Reconstructing discrete measures from projections. consequences on the empirical sliced wasserstein distance. *Comptes Rendus Mathématique*, 2023.

[183] S. Tarasov and M. Vyali. Construction of contour trees in 3D in O(n log n) steps. In *SoCG*, 1998.

[184] J. Tierny, G. Favelier, J. A. Levine, C. Gueunet, and M. Michaux. The Topology ToolKit. *IEEE TVCG*, 2017. `https://topology-tool-kit.github.io/`.

[185] J. Tierny, A. Gyulassy, E. Simon, and V. Pascucci. Loop surgery for volumetric meshes: Reeb graphs reduced to contour trees. *IEEE TVCG*, 2009.

[186] K. Turner. Medians of populations of persistence diagrams, 2019.

[187] K. Turner, Y. Mileyko, S. Mukherjee, and J. Harer. Fréchet Means for Distributions of Persistence Diagrams. *DCG*, 2014.

[188] J. G. van der Corput. Verteilungsfunktionen. I. *Proc. Akad. Wet. Amsterdam*, 38:813–821, 1935.

[189] L. P. van der Maaten and G. Hinton. Visualizing Data Using t-SNE. *JMLR*, 2008.

[190] J. Vidal, J. Budin, and J. Tierny. Progressive Wasserstein Barycenters of Persistence Diagrams. *IEEE TVCG*, 2020.

[191] J. Vidal, J. Budin, and J. Tierny. Progressive wasserstein barycenters of persistence diagrams. *IEEE TVCG*, 26(1):151–161, 2020.

[192] C. Villani. *Optimal transport: old and new*. Springer Verlag, 2008.

[193] G. Voronoi. Nouvelles applications des parametres continus à la théorie des formes quadratiques. deuxième mémoire. recherches sur les parallélloèdres primitifs. *Journal für die reine und angewandte Mathematik*, 134:198–287, 1908.

[194] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. Contour boxplots: A method for characterizing uncertainty in feature sets from simulation ensembles. *IEEE TVCG*, 2013.

[195] D. P. Woodruff. *Sketching as a Tool for Numerical Linear Algebra*. Now Publishers, 2014.

[196] J. Wu, Z. Huang, D. Acharya, W. Li, J. Thoma, D. P. Paudel, and L. V. Gool. Sliced wasserstein generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3713–3722, 2019.

[197] K. Wu and S. Zhang. A contour tree based visualization for exploring data with uncertainty. *IJUQ*, 2013.

[198] X. Xianliang and H. Zhongyi. Central limit theorem for the sliced 1-wasserstein distance and the max-sliced 1-wasserstein distance, 2022.

[199] L. Yan, Y. Wang, E. Munch, E. Gasparovic, and B. Wang. A structural average of labeled merge trees for uncertainty visualization. *IEEE TVCG*, 2019.

Abstract

Topological Data Analysis (TDA) is a family of techniques developed to efficiently and robustly highlight implicit structural patterns in complex datasets. These techniques involve computing a topological descriptor for each element of a dataset by encoding its main topological features in a concise manner. A prominent example is the persistence diagram. However, even though they are concise representations, persistence diagrams can still require significant storage space and may be too complex to be analyzed easily. In this thesis, our goal is to develop an encoding method for ensembles of persistence diagrams while maintaining the same descriptive power. First, we develop a non-linear dictionary encoding for persistence diagrams. Then, we strengthen our approach by making it more robust to outliers within an ensemble of persistence diagrams by using robust barycenters. This dictionary-based approach involves computing Wasserstein distances, which are known to be computationally expensive depending on the size of the input diagrams. One way to address this problem is through Sliced Optimal Transport, more specifically the Sliced Wasserstein distance. We present applications of this work in data reduction to further compress an ensemble of persistence diagrams; in dimensionality reduction by creating a planar view that provides insight into the arrangement of the data; and in robustness to outliers in the context of a clustering problem.

Résumé

L'analyse topologique des données est un ensemble de techniques développées pour mettre en évidence, de manière efficace et robuste, des structures implicites dans des ensembles de données complexes. Ces techniques consistent à calculer un descripteur topologique pour chaque élément d'un jeu de données, en encodant ses principales caractéristiques topologiques de manière concise. Un exemple couramment utilisé est le diagramme de persistance. Cependant, bien qu'il s'agisse de représentations concises, les diagrammes de persistance peuvent nécessiter un espace de stockage important et être parfois trop complexes pour être analysés simplement. Dans cette thèse, notre objectif est de développer une méthode d'encodage pour des ensembles de diagrammes de persistance tout en conservant leur pouvoir descriptif. Nous commençons par développer un encodage non linéaire par dictionnaire pour les diagrammes de persistance. Nous renforçons ensuite notre approche en la rendant plus robuste aux valeurs aberrantes au sein d'un ensemble de diagrammes de persistance, en utilisant des barycentres robustes. Cette approche par dictionnaire implique le calcul de distances de Wasserstein, connues pour être coûteuses en temps de calcul en fonction de la taille des diagrammes en entrée. Une façon de contourner ce problème consiste à utiliser le transport optimal par tranches, plus précisément la distance de Wasserstein tranchée (Sliced Wasserstein). Nous présentons des applications de ce travail à la réduction de données pour compresser davantage un ensemble de diagrammes de persistance ; à la réduction de dimension en créant une vue planaire donnant un aperçu de la disposition des données ; et à la robustesse aux valeurs aberrantes dans le cadre d'un problème de regroupement non supervisé.