Persistence-Based Clustering in Riemannian Manifolds

FRÉDÉRIC CHAZAL, INRIA Saclay – Île-de-France
LEONIDAS J. GUIBAS, Stanford University
STEVE Y. OUDOT, INRIA Saclay – Île-de-France
PRIMOZ SKRABA, INRIA Saclay – Île-de-France

We present a clustering scheme that combines a mode-seeking phase with a cluster merging phase in the corresponding density map. While mode detection is done by a standard graph-based hill-climbing scheme, the novelty of our approach resides in its use of *topological persistence* to guide the merging of clusters. Our algorithm provides additional feedback in the form of a set of points in the plane, called a *persistence diagram* (PD), which provably reflects the prominences of the modes of the density. In practice, this feedback enables the user to choose relevant parameter values, so that under mild sampling conditions the algorithm will output the *correct* number of clusters, a notion that can be made formally sound within persistence theory. In addition, the output clusters have the property that their spatial locations are bound to the ones of the basins of attraction of the peaks of the density.

The algorithm only requires rough estimates of the density at the data points, and knowledge of (approximate) pairwise distances between them. It is therefore applicable in any metric space. Meanwhile, its complexity remains practical: although the size of the input distance matrix may be up to quadratic in the number of data points, a careful implementation only uses a linear amount of memory and takes barely more time to run than to read through the input.

Categories and Subject Descriptors: F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems — Geometrical problems and computations, Computations on discrete structures

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Unsupervised Learning, Clustering, Mode Seeking, Topological Persistence, Morse Theory, Computational Topology

ACM Reference Format:
Chazal, F., Guibas, L. J., Oudot, S. Y., and Skraba, P. 2011. Persistence-Based Clustering in Riemannian Manifolds. *J. ACM* 60, 6, Article A (January 2013), 38 pages.
DOI:http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

Unsupervised learning or clustering is an important tool for understanding and interpreting data in a variety of fields. Obtaining the most natural clustering is an ill-posed problem in general, and it is particularly difficult with massive and high-dimensional

This work was carried out within the associated teams TGDA and CoMeT. It was suported by ANR grant GIGA, NSF grants FODAVA 0808515, DMS 0900700, and CCF 1011228, by a Stanford-France grant, and by a gift from Google, Inc. This work is supported by the National Science Foundation, under grant CNS-0435060, grant CCR-0325197 and grant EN-CS-0329609.

Author's addresses: F. Chazal, S. Y. Oudot and P. Skraba, INRIA Saclay – Île-de-France, 4 rue Jacques Monod. 91893 Orsay Cedex, France; Leonidas J. Guibas, Computer Science Department, Gates Building, Stanford University, Stanford, CA 94305, USA.

DOI:http://dx.doi.org/10.1145/0000000.0000000

data sets where visualization techniques fail. The breadth of the existing work on clustering [Hartigan 1975] shows the high interest this topic has aroused among the scientific community. Here we recount a few classical methods to show where our approach stands with respect to the literature:

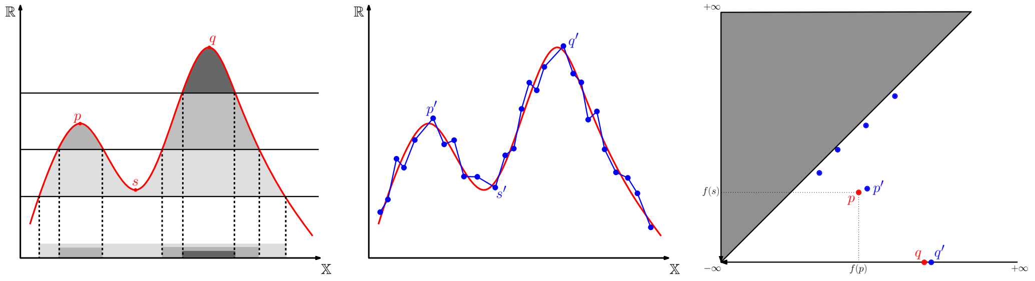
K-means [Lloyd 1982] is perhaps the most commonly used approach. Given a fixed number k of clusters, it tries to place cluster centers and define cluster boundaries so as to minimize the sum of the squared distances to the center within each cluster. This minimization problem is known to be NP-hard, so k-means resorts to an iterative expectation-maximization procedure that is guaranteed to converge at least to some local minimum. This minimum is not guaranteed to be global, however. Another issue with k-means and its variants is that they produce bad results on highly non-convex clusters.

Spectral clustering [von Luxburg 2007] was designed specifically to work on non-convex data. It first computes an embedding of the data set endowed with a diffusion distance between the points, given by a Laplacian of some neighborhood graph. Then, it applies the standard k-means method in the new ambient space. Computing the embedding requires an eigendecomposition of the Laplacian, which may have numerical issues as the size of the data grows. The presence of a gap in the spectrum of the Laplacian gives an indication of the correct number k of clusters. However, problems arise when there are more than a small number of outliers in the data, in which case no such gap may exist.

Density-based techniques make the assumption that the data points are drawn from some unknown density function f. Clustering becomes then a problem of understanding the structure of f, as estimated from the samples. A popular approach consists in thresholding the density at some fixed level α, then treating the connected components of the superlevel-set $F^\alpha = f^{-1}([\alpha, +\infty))$ as clusters and the rest of the data as noise. In practice, the density f is unknown so its superlevel F^α needs to be approximated from the data, which algorithms like DBSCAN [Ester et al. 1996; Sander et al. 1998] do by various graph-based heuristics. Unfortunately, due to the use of a fixed density threshold α, these techniques do not respond well to hierarchical data sets, in which subtle multi-scale clustering phenomena may occur.

Another popular approach, called *mode-seeking*, consists in detecting the local peaks of f in order to use them as cluster centers and to partition the data according to their *basins of attraction*. The precise notion of the basin of attraction B_p of a peak p varies between references, yet the bottom line remains that B_p corresponds to the subset of the data points that eventually reach p by some greedy hill-climbing procedure. This line of work started with the algorithm of [Koontz et al. 1976] and was followed by numerous variants and extensions, including Mean-Shift [Comaniciu and Meer 2002] and its successors [Sheikh et al. 2007; Vedaldi and Soatto 2008]. A common issue faced by these techniques is that the gradient and extremal points of a density function are notoriously unstable, so their approximation from a density estimator can lead to unpredictable results. This is why methods such as Mean-Shift adopt a proactive strategy that consists in smoothing the estimator before launching the hill-climbing procedure, which in turn raises the difficult question of how much smoothing is needed to remove the noise without affecting the signal, and to obtain the correct number of clusters.

Enter topological persistence. In this paper, we adopt a more reactive strategy that consists in using *topological persistence* [Edelsbrunner et al. 2002; Zomorodian and Carlsson 2005] to detect and merge unstable clusters after their computation, thus regaining some stability. Although our method belongs to the same family as Mean-Shift, the use of persistence makes it possible to link explicitly the input parameter values



(a) (b) (c)

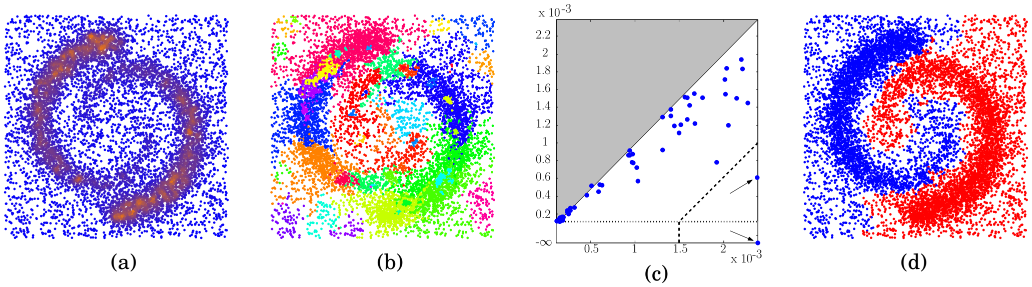
Fig. 1. Sketch of topological persistence: (a) a new connected component is born in the superlevel-set F^α when $\alpha = f(p)$, and it dies when $\alpha = f(s)$; its lifespan is represented as a point in the PD of f; (b) a piecewise-linear approximation $\tilde{f}$ of f; (c) superimposition of the PDs of f (red) and $\tilde{f}$ (blue), showing the one-to-one correspondence between the prominent peaks of f and $\tilde{f}$.

to the output number of clusters. It also provides a sound theoretical framework for characterizing the correct number of clusters, in the same spirit as spectral clustering.

Topological persistence estimates the *prominence* (also called *persistence*) of the density peaks and builds a hierarchy of the peaks based on it. The prominence of a peak is defined as the difference between its height and the level at which its basin of attraction meets the one of a higher peak (its parent in the hierarchy). More precisely, focusing on the 1-parameter family of superlevel-sets $F^\alpha = f^{-1}([\alpha, +\infty))$ of the density function f, persistence studies the evolution of the connectivity (and more generally, of the topology) of F^α as α ranges from $+\infty$ to $-\infty$. A new connected component C is born in F^α when α reaches the height of a peak p of f, and dies when it gets connected in F^α to the component of a higher peak (see Figure 1(a)). As mentioned above, the prominence of p is simply the height difference between birth and death values of C. The lifespan of each connected component C can be represented as a point in the plane, with the x-coordinate giving the birth time of C and the y-coordinate giving its death time. The collection of such points is called the (0-dimensional) *persistence diagram* (PD) of f, illustrated in Figure 1(c). The key insight of this planar data representation is that the PD reveals part of the topological structure of the density function f. More precisely, each peak of f is uniquely represented by one point in the PD, and its prominence is given by the vertical distance of this point to the diagonal $y = x$.

Originally defined in *Morse theory*, prominence is known to be more stable than other measures of significance such as absolute height. For example, a small bump occurring at a high density will have large absolute height but small prominence. The same kind of stability holds for PDs. For instance, f and its noisy approximation $\tilde{f}$ (see Figure 1(b)) have similar PDs, in the sense that there is a one-to-one mapping of small amplitude from the prominent peaks of $\tilde{f}$ to the ones of f, the rest of the peaks being treated as topological noise and mapped to the diagonal in the PD (see Figure 1(c)). Thanks to this fundamental stability property, with only limited knowledge of the underlying space and a finite estimate of the density f it is possible to *provably* and *efficiently* approximate the PD of f. The combination of such guarantees with computational practicality is at the heart of topological data analysis [Carlsson et al. 2004; Carlsson 2009; Carlsson et al. 2008; Ghrist 2007], which includes this work.

It is worth noting that PDs are similar in spirit to the *dendrograms* provided by agglomerative clustering schemes, whose principle is to build the clusters in a bottom-up fashion, starting with each point being its own cluster and merging at each step the most similar clusters together. The output dendrogram describes the sequence of



(a) (b) (c) (d)

Fig. 2. Our approach in a nutshell: (a) estimation of the underlying density function f at the data points; (b) result of the basic graph-based hill-climbing step; (c) approximate PD showing 2 points far off the diagonal corresponding to the 2 prominent peaks of f; (d) final result obtained after merging the clusters of non-prominent peaks.

merges that have occurred during the process, thus encoding the hierarchical structure of the obtained family of clusterings. While these techniques bear some connections with ours, they are actually based on a different clustering paradigm that suffers from its own limitations — see e.g. Section 14.3.12 in [Hastie et al. 2009].

Our method. Our clustering scheme, called ToMATo (*Topological Mode Analysis Tool*), combines the original graph-based hill-climbing algorithm of [Koontz et al. 1976] with a cluster merging step guided by persistence. As illustrated in Figure 2(b), hill-climbing is very sensitive to perturbations of the density function f that arise from a density estimator $\tilde{f}$. Computing the PD of $\tilde{f}$ enables us to quantify the prominences of its peaks and, in favorable cases, to distinguish those that correspond to peaks of the true density f from those that are inconsequential. In Figure 2(c) for instance, we can see 2 points (pointed to by arrows) that are further from the diagonal than the other points: these correspond to the 2 prominent peaks of $\tilde{f}$ (one of them is at $y = -\infty$, since the highest peak never dies). To obtain the final clustering, we merge every cluster of prominence less than a given thresholding parameter τ into its parent cluster in the persistence hierarchy. As shown in Figures 2(c) and 2(d), the PD gives us a precise understanding of the relationship between the choice of τ and the number of obtained clusters.

In practice we run ToMATo twice: in the first run we set $\tau = +\infty$ to merge all clusters and thus compute the PD; then, using the PD we choose a value for τ (which amounts to selecting the number of clusters) and re-run the algorithm to obtain the final result. The feedback provided by the PD proves invaluable in interpreting the clustering results in many cases. Indeed, the PD gives a clear indication of whether or not there is a natural number of clusters, and because it is a planar point cloud we can understand its structure visually, regardless of the dimensionality of the input data.

ToMATo is highly generic and agnostic to the choice of distance, underlying graph, and density estimator. Our theoretical guarantees make use of graphs that do not require the geographic coordinates of the data points at hand (only pairwise distances are used) nor estimates of the density at extra points. This makes the algorithm applicable in very general settings. ToMATo is also highly efficient: in the worst case it has an almost-linear running time in the size of the underlying graph, and only a linear memory usage in the number of data points. Most often we use Euclidean distances, however other metrics such as diffusion distances can be used. Indeed, the choice of metric and density estimator define the space we study, while our algorithm gives the structure of this space. Finally, ToMATo comes with a solid mathematical formulation. We show that, given a finite sampling of an *unknown* space with pointwise estimates of an *unknown* density function f, our algorithm computes a faithful approximation of the PD of f. Under conditions of a sufficient *signal-to-noise* ratio in this PD, we can determine the correct number of clusters and show that significant clusters always have stable regions. In some applications, the number of clusters is not obvious and we see this in the corresponding PDs. However, in these cases the relationship between the choice of parameters and the number of obtained clusters is transparent.

Obtaining guarantees in such general settings using only simple tools like neighborhood graphs is made possible by recent advances on the stability of persistence diagrams [Chazal et al. 2009; Chazal et al. 2012]. Previous stability results [Cohen-Steiner et al. 2007] required the use of piecewise-linear approximations of the density functions, as in Figure 1(b) for instance. The construction of such approximations becomes quickly intractable when the dimensionality of the data grows. This fact might explain why topological persistence was never really exploited in mode analysis before, except in some restricted or low-dimensional settings [Paris and Durand 2007].

Paper layout. In the first part of the paper (Sections 2 through 5) we emphasize the experimental aspects of our work, describing the approach, giving an intuitive overview of its theoretical guarantees, discussing the choice of its parameters in practice, and demonstrating its potential in terms of applications through a series of experimental results obtained on synthetic and real-life data sets. The precise statements and proofs of our theoretical claims are detailed in the second part of the paper (Sections 6 through 11).

PART I: *APPROACH, GUARANTEES, AND RESULTS*

2. THE ALGORITHM

We first provide an intuitive insight into our approach by considering the continuous setting underlying our input. We then give the details of the algorithm in the discrete setting.

The continuous setting. Consider an m-dimensional Riemannian manifold $\mathbb{X}$ and a Morse function $f : \mathbb{X} \to \mathbb{R}$, *i.e.* a C^∞-continuous function with non-degenerate critical points such that all the critical values are distinct. Assume that f has a finite number of critical points. The *ascending region* of a critical point m, noted $A(m)$, is the subset of the points of $\mathbb{X}$ that eventually reach m by moving along the flow induced by the gradient vector field of f. For all $x \in A(m)$, we call m the *root* of x. Ascending regions of the peaks of f are known to form pairwise-disjoint open cells homeomorphic to $\mathbb{R}^m$. Furthermore, assuming $\mathbb{X}$ to have no boundary and f to be bounded from above and proper[1], the ascending regions of the peaks of f cover $\mathbb{X}$ up to a subset of Hausdorff measure zero. It is then natural to use them to partition (almost all) the space $\mathbb{X}$ into regions of influence.

For any $\alpha \in \mathbb{R}$, let F^α denote the closed superlevel-set $f^{-1}([\alpha, +\infty))$. Consider the nested family of spaces $\{F^\alpha\}_{\alpha\in\mathbb{R}}$ obtained by letting parameter α decrease from $+\infty$ to $-\infty$. This family is called the *superlevel-sets filtration* of f. For any $\alpha \in \mathbb{R}$ and $x \in \mathbb{X}$, let $C(x, \alpha) \subseteq F^\alpha$ denote the path-connected component of F^α that contains x. Morse theory tells us that when a local maximum m_p of f enters the superlevel-sets filtration, at time $\alpha = f(m_p)$, a new path-connected component $C(m_p, \alpha)$ appears in

[1] Meaning that for any bounded closed interval $[a, b] \subset \mathbb{R}$, the pre-image $f^{-1}([a, b])$ is a compact subset of $\mathbb{X}$.

the superlevel-set F^α. In homological terms, the peak m_p is called the *generator* of the component born at time $f(m_p)$. This component ceases to be independent in F^α when it gets connected to another component generated by a higher peak m_q. At that particular time, noted $\alpha = d(m_p)$, persistence theory tells us that $C(m_p, \alpha)$ gets *merged* into $C(m_q, \alpha)$. While m_q remains the generator of the component $C(m_q, \alpha)$, m_p ceases to be a generator, and by analogy we call m_q its root, noted $m_q = r(m_p)$. In the (0-dimensional) persistence diagram $\mathrm{D}_0 f$, the lifespan of m_p as a generator is encoded by the point p of coordinates $p_x = f(m_p)$ and $p_y = d(m_p) \leq p_x$. The difference $p_x - p_y \geq 0$ between birth and death times is called the *prominence* of the peak m_p. Equivalently, we say that m_p is $(p_x - p_y)$-*prominent*. As for the peak m_q, if it remains the generator of $C(m_q, \alpha)$ for all values $\alpha \leq f(m_q)$, then persistence theory sets its death-time $d(m_q)$ to $-\infty$, so its lifespan is represented in $\mathrm{D}_0 f$ by the point $(f(m_q),\ -\infty)$ and its prominence is infinite.

Given a thresholding parameter $\tau \geq 0$, we restrict our focus to the peaks m_p of f of prominence at least τ. Intuitively, the points of $\mathbb{X}$ that are *attracted* by m_p are the ones belonging to ascending regions that are eventually merged by persistence into the connected component of m_p before being merged into the component of any other peak of prominence at least τ. Formally, for every peak m_q of f (of arbitrary prominence), let us iterate the *root map* $m_q \mapsto r(m_q)$ until some peak of prominence at least τ is reached[2]. We call r^*_τ the thus iterated root map, and we note that every peak of prominence at least τ is a fixed point of r^*_τ. The union of the ascending regions of the peaks mapped to m_p through r^*_τ is referred to as *the basin of attraction of* m_p (of parameter τ) in the paper, noted $B_\tau(m_p)$:

$$\forall m_p \text{ s.t. } p_x - p_y \geq \tau,\ B_\tau(m_p) = \bigcup_{r^*_\tau(m_q) = m_p} A(m_q). \tag{1}$$

Note that $B_\tau(m_p)$ contains $A(m_p)$ since m_p is a fixed point of r^*_τ. More precisely, we have $A(m_p) = B_0(m_p) \subseteq B_\tau(m_p)$. In addition, since the iterated root map $m_q \mapsto r^*_\tau(m_q)$ is uniquely defined, the basins of attraction form a partition of the union of all ascending regions. These basins are our target clusters.

The discrete setting. ToMATo takes as input an unweighted simple graph G, whose vertex set represents the data points and whose edges connect the points according to some user-defined proximity rule. Each vertex i of G must be assigned a non-negative value $\tilde{f}(i)$ corresponding to the estimated density at that point. In addition, ToMATo takes in a non-negative *merging* parameter τ, whose choice and use are elaborated below. In this discrete setting, the algorithm mimics the process described above in the continuous setting by running the following procedures in this order:

1. (Mode-seeking) To compute the initial clusters, ToMATo iterates over the vertices of G sorted by decreasing $\tilde{f}$-values : at each vertex i, it simulates the effect of the gradient of the underlying density function by connecting i to its neighbor in G with highest $\tilde{f}$-value, if that value is higher than $\tilde{f}(i)$. Otherwise, all neighbors of i have lower values, so i is declared a peak of $\tilde{f}$. The resulting collection of pseudo-gradient edges forms a spanning forest of the graph, and each tree in this forest can be viewed as the analog within G of the ascending region of a peak of the true density function in the underlying continuous domain.

[2] Such a prominent peak is always reached eventually, since the function f has finitely many peaks and since the root map satisfies $f(m_q) < f(r(m_q))$, meaning that $r(m_q)$ is more prominent than m_q.

2. (Merging) To handle merges between trees, ToMATo iterates over the vertices of G again, in the same order, while maintaining a union-find data structure $\mathcal{U}$, where each entry corresponds to a union of trees of the spanning forest. We call *root* of an entry e, or $r(e)$ for short, the vertex contained in e whose $\tilde{f}$-value is highest. By definition, this vertex is the root of one of the trees contained in e, that is, a local peak of $\tilde{f}$ in G. During the iteration process, two different scenarios may occur when a vertex i is considered:
   (a) Vertex i is a peak of $\tilde{f}$ within G, *i.e.* the root of some tree T. Then, i creates a new entry e in $\mathcal{U}$, in which T is stored, and we let $r(e) = i$.
   (b) Vertex i is not a peak and therefore belongs to some tree stored in an existing entry e_i of $\mathcal{U}$ (of which i is not the root). Then, we compute the set $\mathcal{E}$ of the entries of $\mathcal{U}$ that contain neighbors of i in G. We iterate over this set in any order, and for each entry $e \in \mathcal{E}$ considered, we check whether $e \neq e_i$ and $\min\{\tilde{f}(r(e)),\ \tilde{f}(r(e_i))\} < \tilde{f}(i) + \tau$, that is, whether the two entries differ and at least one of them has a less than τ-prominent root. If so, then e and e_i are merged into a single entry $e \cup e_i$ in $\mathcal{U}$, and we let $r(e \cup e_i) = \operatorname{argmax}_{\{r(e),\ r(e_i)\}} \tilde{f}$, so in effect the entry with the lower root is merged into the one with the higher root.

Upon termination, the (merged) clusters stored in the entries of the union-find data structure $\mathcal{U}$ form a partition of the vertex set of G, and their roots are the peaks of $\tilde{f}$ of prominence at least τ within the graph. The output of ToMATo is then the subset of this collection of clusters that is stored in those entries e such that $\tilde{f}(r(e)) \geq \tau$. The rest of the data points is stored in entries with roots lower than τ, so it is treated as background noise and discarded from the data set[3].

In addition to the clustering, ToMATo outputs the lifespans of all the entries that have been created in the union-find data structure during the merging phase. By analogy with the continuous setting, an entry is born when it is created in $\mathcal{U}$ with a single tree attached to it as described in scenario (a) above, and it dies when it gets merged into another entry with higher root as described in scenario (b). For ease of visualization, the lifespan is represented as a point (x, y) in the plane, where x is the birth time and y the death time of the entry ($y = -\infty$ if the entry never gets merged into another one). It is easy to see that the thus obtained planar diagram of points coincides with the persistence diagram of the scalar field $\tilde{f}$ when parameter τ is set to $+\infty$, as the condition $\min\{\tilde{f}(r(e)),\ \tilde{f}(r(e_i))\} < \tilde{f}(i) + \tau$ in scenario (b) becomes always trivially satisfied and the merging rule is the one prescribed by persistence theory. When $\tau < +\infty$, the entries whose roots are at least τ-prominent never get merged into other entries, so their corresponding points in the output diagram are projected down vertically onto the horizontal line $y = -\infty$.

Implementation details and complexity. In practice the mode-seeking and merging procedures can be run simultaneously during a single pass over the vertices of the graph G: for each considered vertex i, the approximate gradient at i is computed, then the possible merges in the union-find data structure $\mathcal{U}$ are performed—these involve only previously visited vertices. The corresponding pseudo-code is given in Algorithm 1.

[3] This extra filtering step departs from the approach described in the continuous setting. It stems from the observation that the data points may not be densely sampled over the entire manifold $\mathbb{X}$. Depending on the proximity rule used in the definition of the neighborhood graph G, the sparseness of the data in low-density regions may create independent connected components that give birth to spurious clusters with infinite prominence — see Figure 6 for an illustrative example.

The mode-seeking phase takes a linear time in the size of G once the vertices have been sorted. As for the merging phase, it makes $O(n)$ `union` and $O(m)$ `find` queries to the union-find data structure $\mathcal{U}$, where n and m are respectively the number of vertices and the number of edges of G. If an appropriate representation is used for $\mathcal{U}$ (e.g. a disjoint-set forest [Cormen et al. 2001]), and if the vertex gradients and the entry roots are stored in separate containers with constant-time access (e.g. arrays), then the worst-case time complexity of Algorithm 1 becomes $O(n \log n + m\alpha(n))$, where α stands for the inverse Ackermann function.

As for the space complexity, note that the graph G does not have to be stored entirely in main memory, since only the neighborhood of the current vertex i is involved at the i-th iteration of the `clustering` procedure. The main memory usage is thus reduced to $O(n)$, where n is the number of vertices of G. The total space complexity remains $O(n+m)$ though, as the graph needs to be stored somewhere (e.g. on the disk).

ALGORITHM 1: `Clustering`

```
Input: simple graph $G$ with $n$ vertices, $n$-dimensional vector $\tilde{f}$, real parameter $\tau \geq 0$.

Sort the vertex indices $\{1, 2, \cdots, n\}$ so that $\tilde{f}(1) \geq \tilde{f}(2) \geq \cdots \geq \tilde{f}(n)$;
Initialize a union-find data structure $\mathcal{U}$ and two vectors $g, r$ of size $n$;
for $i = 1$ to $n$ do
    Let $\mathcal{N}$ be the set of neighbors of $i$ in $G$ that have indices lower than $i$;
    if $\mathcal{N} = \emptyset$ then
        // vertex $i$ is a peak of $\tilde{f}$ within $G$
        Create a new entry $e$ in $\mathcal{U}$ and attach vertex $i$ to it;
        $r(e) \leftarrow i$;            // $r(e)$ stores the root vertex associated with the entry $e$
    else
        // vertex $i$ is not a peak of $\tilde{f}$ within $G$
        $g(i) \leftarrow \mathrm{argmax}_{j \in \mathcal{N}} \tilde{f}(j)$;      // $g(i)$ stores the approximate gradient at vertex $i$
        $e_i \leftarrow \mathcal{U}.\mathtt{find}(g(i))$;
        Attach vertex $i$ to the entry $e_i$;
        for $j \in \mathcal{N}$ do
            $e \leftarrow \mathcal{U}.\mathtt{find}(j)$;
            if $e \neq e_i$ and $\min\{\tilde{f}(r(e)),\ \tilde{f}(r(e_i))\} < \tilde{f}(i) + \tau$ then
                $\mathcal{U}.\mathtt{union}(e,\ e_i)$;
                $r(e \cup e_i) \leftarrow \mathrm{argmax}_{\{r(e),\ r(e_i)\}} \tilde{f}$;
                $e_i \leftarrow e \cup e_i$;
            end
        end
    end
end

Output: the collection of entries $e$ of $\mathcal{U}$ such that $\tilde{f}(r(e)) \geq \tau$.
```

3. PARAMETER SELECTION

ToMATo takes in three inputs: the neighborhood graph G, the density estimator $\tilde{f}$, and the merging parameter τ. Although the freedom left to the user in the choice of these inputs gives our approach a lot of flexibility, the latter must not come at the expense of a significant increase in the amount of effort needed to run the program. This is why this section provides some insights into the choice of parameters.

Neighborhood graph G. ToMATo relies heavily on the neighborhood information encoded in the input graph G. Choosing a relevant neighborhood graph (and thereby a relevant metric) is a problem faced by many clustering techniques. In our experiments we primarily used the *δ-Rips graph*, which connects two data points whenever they lie within distance δ of each other. This purely metric definition makes it possible to use these graphs in arbitrary metric spaces, and to interpret the structure of the obtained PDs thanks to a sound theoretical framework (see Section 4). The choice of a particular value for δ corresponds more or less to the choice of a scale at which to inspect the data. It can be tricky on some instances, where different choices of scale may reveal different structures. This is why we recommend running ToMATo at several scales, either sequentially or in parallel. This can be done even for large data sets thanks to the efficiency of the algorithm. For too large values of δ there will be no real structure in the PD, while too small values of δ will produce too many infinitely prominent peaks in the PD, corresponding to the connected components of the graph. By examining the PDs obtained at different scales, one can find an appropriate trade-off.

Another popular choice of neighborhood graph is the k-nearest neighbor (k-nn) graph. Its main advantage is that it remains sparse whatever the layout of the data. We tested the algorithm with this graph and generally found that it performed well, recovering the correct clusters under a suitable choice of parameter k. However, to the best of our knowledge there currently exists no theory that validates these empirical observations, and in practice we were left with the task of choosing k, which we accomplished by trial-and-error.

We also ran ToMATo using Delaunay graphs and some of their variants [Toussaint 1980]. These have the great advantage of being parameter-free, and the disadvantage of creating long edges connecting high-density areas that are far apart, thus leading to artificial merges between clusters. One way around this issue is to discretize the long edges and to estimate the density at the newly created nodes, in order to reveal additional valleys that separate the prominent peaks. This requires the ability to estimate the density outside the input point cloud, which is generally the case when a Delaunay graph is built.

Density estimator $\tilde{f}$. While the algorithm is agnostic to the choice of density estimator, we experimented with two of them: a truncated Gaussian kernel estimator, and the *distance to a measure* estimator[4] proposed in [Biau et al. 2011]. Each of these estimators uses one parameter, and we refer the reader to the appropriate references for some insights into the choice of these parameters.

Merging parameter τ. During the merging phase, ToMATo eventually merges all clusters of prominence less than τ into clusters of prominence at least τ. In other words, the choice of τ determines which peaks of $\tilde{f}$ are considered significant. To choose τ, we run ToMATo twice. In the first run, τ is set to $+\infty$, which makes ToMATo output the PD of the scalar field $\tilde{f}$ over the graph G, just as the 0-dimensional version of the standard persistence algorithm [Edelsbrunner et al. 2002] would do. This PD reveals the topological structure of $\tilde{f}$, providing the height and prominence of each peak of $\tilde{f}$. Hence it can be used to determine a suitable value for τ, to be assigned in a second run of ToMATo that computes the final clustering.

In cases where the PD of $\tilde{f}$ shows a large gap separating a small set of k highly prominent peaks from the rest of the structure, we infer that the number of clusters

[4] Given an integer parameter k, the distance of a point x to the empirical measure of support a finite set of points P is the square root of the average of the squared distances of x to its k nearest neighbors in P. The inverse of this quantity is used as density estimator.

is likely to be k, and so we set τ to be any value between the prominences of the k distinguished peaks and the prominences of the rest of the PD. Then the output of the second run of ToMATo contains exactly k clusters. Detecting a large gap automatically can be done by means of the following simple heuristic: we sort the points in the PD by decreasing prominence (possibly weighted by the corresponding peak heights, to avoid a squeezing effect due to the presence of extremely or even infinitely prominent peaks), and then we look for the largest drop in the sequence of (weighted) prominences. This is reminiscent of what is commonly done in spectral clustering for finding a gap in a Laplacian spectrum, and in fact our prominence gap and the spectral gap play very similar roles, even if in completely different settings.

In cases where the PD of $\tilde{f}$ does not show any well-separated structure, it still provides a clear relationship between the choice of parameter τ and the number of clusters obtained after re-running ToMATo. The choice of a particular value (or of a collection of values) for τ depends on the context, and in practice it requires to use additional application-specific information on the data. This is what we did for instance on the biological data set to distinguish between several possible choices of τ (see Section 5.2).

4. THEORETICAL GUARANTEES

In this section, we give an intuitive overview of the theoretical guarantees that come along with ToMATo and validate the above heuristics. Formal statements and proofs can be found in the second part of the paper (Sections 6 through 11).

Let $\mathbb{X}$ be an m-dimensional Riemannian manifold with positive convexity radius[5], and $f : \mathbb{X} \to \mathbb{R}$ a Lipschitz-continuous probability density function with respect to the m-dimensional Hausdorff measure. We assume that the input data set P has been sampled over $\mathbb{X}$ according to f in i.i.d. fashion, and that the values of f at the data points and the geodesic distances in $\mathbb{X}$ between the data points are known either exactly or within a small additive error. Finally, we assume the input graph G to be the δ-Rips graph built over P using the estimated geodesic distances, for some user-defined parameter δ.

Definition 4.1. Given two values $d_2 > d_1 \geq 0$, the persistence diagram $\mathrm{D}_0 f$ is called (d_1, d_2)-*separated* if every point of $\mathrm{D}_0 f$ lies either in the region D_1 above the diagonal line $y = x - d_1$, or in the region D_2 below the diagonal line $y = x - d_2$ and to the right of the vertical line $x = d_2$.

This condition formalizes the intuitive notion that the points of $\mathrm{D}_0 f$ can be separated between prominent peaks (region D_2) and topological noise (region D_1), as illustrated in Figure 3. In this respect, it acts very similarly to a signal-to-noise ratio condition: the larger the prominence gap $d_2 - d_1$, the more clearly the prominent peaks are separated from the noise. In the limit case where $d_1 = 0$, all peaks of f are at least d_2-prominent and none of them is viewed as noise. The additional condition that the points of D_2 must lie to the right of the vertical line $x = d_2$ follows the description of the extra filtering step performed by the algorithm after the merging phase, and it stems from the fact that only some superlevel-set of the density f can be densely sampled by the data points.

Our first result relates the number of clusters computed by the algorithm to the number of prominent peaks of f. Using the stability of persistence diagrams [Chazal et al. 2009; Chazal et al. 2012] to relate the diagram of f to the diagram output by

[5] Recall that the convexity radius of $\mathbb{X}$ is the infimum over the points $x \in \mathbb{X}$ of the supremum over the values $r \geq 0$ such that any geodesic ball of center x and radius $r' < r$ is geodesically convex, that is, any two points in that ball are joined by a unique geodesic of length less than $2r'$, and this geodesic is contained in the ball.

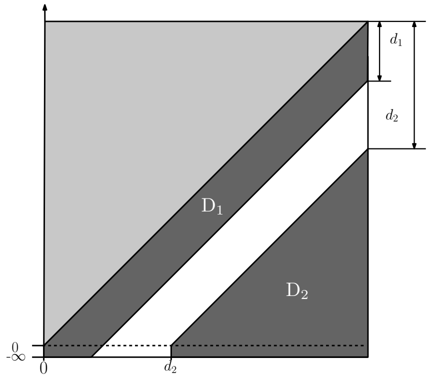


Fig. 3. The separation of the persistence diagram $D_0 f$ between prominent peaks (region D_2) and topological noise (region D_1).

step 2 of the algorithm, we can prove that the regions D_1 and D_2 remain disjoint under perturbations caused by our approximation, and can therefore be separated using any value within a certain range for the thresholding parameter τ. With such values of parameter τ as input, the algorithm computes the correct number of clusters with high probability:

RESULT 1 (THEOREM 9.2). *If $D_0 f$ is (d_1, d_2)-separated and if the Rips parameter $\delta > 0$ is smaller than a fraction of $d_2 - d_1$ and of the convexity radius of $\mathbb{X}$, then there is a range $[d_1 + O(\delta),\, d_2 - O(\delta)]$ of values of the thresholding parameter τ such that the number of clusters output by the algorithm is equal to the number of peaks of f of prominence at least τ with probability at least $1 - e^{-\Omega(n)}$, where n is the number of data points.*

Explicit bounds are given in Theorem 9.2. The big-O notations hide factors proportional to the Lipschitz constant c of f. The big-Ω notation hides a factor increasing monotonically with c and δ and depending on certain geometric quantities of the manifold $\mathbb{X}$. As can be seen fro the statement, the larger the prominence gap $d_2 - d_1$, the larger the range of admissible values for τ, and of course the more easily this range can be detected. In the meantime, the smaller δ, the larger the range, but also the smaller the probability of success[6].

Another question is how well the output of the algorithm approximates the basins of attraction of the prominent peaks over the point cloud, assuming that f is of Morse type. In full generality, this is a hopeless question since the basins of attaction are not stable even in the smooth case. There are indeed many examples of very close functions having very different basins of attraction, and clearly the algorithm cannot provably-well approximate the unstable parts of the basins. An illustrative example is given in Figures 4 and 5. Yet, we can ensure that the output of the algorithm approximates some stable parts of the basins:

RESULT 2 (THEOREM 10.1). *Under the same hypotheses as in Result 1, it holds with probability at least $1 - e^{-\Omega(n)}$ that for every point $p \in D_2$ the algorithm outputs a cluster C such that $C \cap F^\alpha = B_\tau(m_p) \cap P \cap F^\alpha$ for all values $\alpha \in [\alpha_\tau(m_p) + d_1 + O(\delta),\, f(m_p))$, where m_p is the peak of f corresponding to point p, where $B_\tau(m_p)$ denotes the basin of attraction of m_p in the underlying manifold $\mathbb{X}$, and where $\alpha_\tau(m_p)$ is the first*

[6] This follows the intuition that a minimum point density is required for the connectivity of the δ-Rips graph to reflect the one of some superlevel-set of the density f.

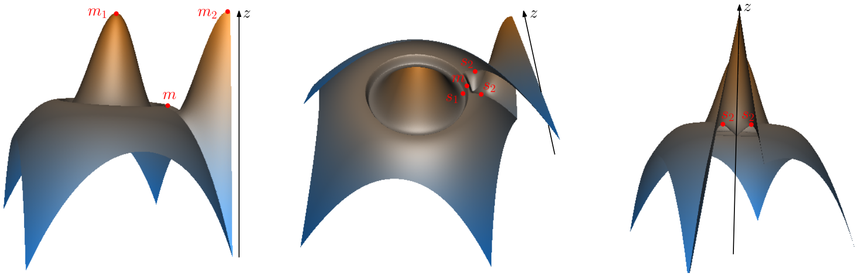


Fig. 4. A function $f : [0,1]^2 \to \mathbb{R}$ with unstable basins of attraction. The three peaks m, m_1, m_2 have respective prominences $f(m) - f(s_2)$, $f(m_1) - f(s_1)$, and $+\infty$. When $\tau > f(m) - f(s_2)$, the ascending region $A(m)$ is merged into the basin of attraction $B_\tau(m_2)$ at the value $\alpha = f(s_2)$. However, since $f(s_2) - f(s_1)$ can be made arbitrarily small compared to $f(m_1) - f(m)$, arbitrarily small perturbations of f compared to the prominence gap $f(m_1) - f(m) + f(s_2) - f(s_1)$ merge $A(m)$ into $B_\tau(m_1)$ instead, thus making $A(m)$ an unstable part of $B_\tau(m_2)$. In the discrete setting, where the square $[0,1]^2$ is replaced by a point cloud, different samplings of the square or different values of parameter δ lead to different merges of the cluster associated with m. This erratic behavior of the algorithm only stops when δ becomes small enough compared to the (arbitrarily small) quantity $f(s_2) - f(s_1)$.

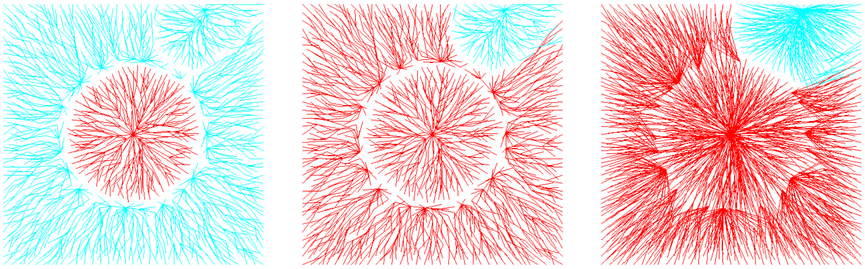
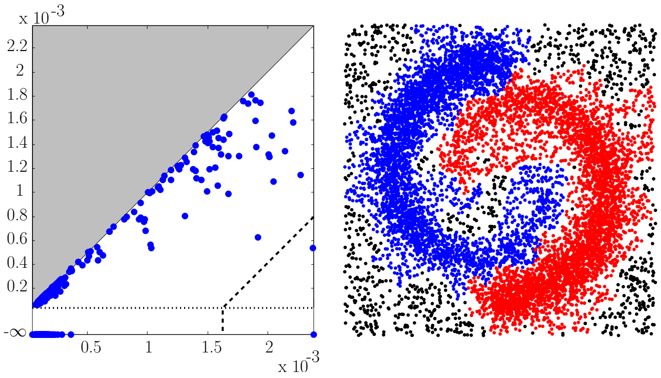


Fig. 5. Outputs of the algorithm obtained from a uniform ε-sample P of the unit square ($\varepsilon = 0.15$) endowed with the function f of Fig. 4. We chose a value of τ that gives two clusters, and we used three different values for the Rips parameter: $\delta = 0.27$ (left), $\delta = 0.28$ (center), $\delta = 0.6$ (right). Notice how some values of δ induce a correct merge of $A(m)$ into $B_\tau(m_2)$ whereas others induce an incorrect merge of $A(m)$ into $B_\tau(m_1)$. The limit value of ε below which no such failure of the algorithm occurs depends on the arbitrarily small quantity $f(s_2) - f(s_1)$.

value of α at which $B_\tau(m_p)$ gets connected to the basin of attraction of another peak of f of prominence at least τ in the superlevel-set F^α.

In plain words, cluster C is the *trace* of the basin of attraction $B_\tau(m_p)$ over the point cloud P, until (approximately) the value $\alpha_\tau(m_p)$ at which $B_\tau(m_p)$ meets the basin of another τ-prominent peak of f. Beyond that value, the cluster may start diverging from the basin, which itself may start being unstable, as illustrated in Figures 4 and 5. As will be shown in Section 10 (Eq. 6), we have $\alpha_\tau(m_p) \leq f(m_p) - d_2$, so the length of the interval of values of α for which C is the trace of $B_\tau(m_p)$ over P is at least $d_2 - d_1 - O(\delta)$.

Our proof of Result 2 also shows an important fact, namely: that each basin of attraction $B_\tau(m_p)$ is stable under small perturbations of the function f, at least between values $f(m_p)$ and $\alpha_\tau(m_p) + d_1 + O(\delta)$. This fact opens the door to a more statistical approach to clustering: since we know the top parts of the basins (and therefore of the clusters computed by the algorithm) are stable under small perturbations of the function, we can conduct multiple runs of the algorithm with random perturbations of the function, and then find correspondences between the outputs of different runs. Each point can then be assigned a quantitative measure of its classification stability over the runs.



(a) (b)

Fig. 6. The twin spirals data set from Figure 2, processed using a smaller Rips parameter: (a) the persistence diagram; (b) the final clustering with late appearing connected components filtered out (in black).

Note finally that the probabilistic nature of our theoretical results does not stem from the algorithm itself, which is deterministic, but from the fact that the input data set must form a dense sampling of some superlevel-set of f for the algorithm to produce a faithful approximation of $\mathrm{D}_0 f$. This event can only occur with some probability since the data points are sampled at random from f.

5. EXPERIMENTAL RESULTS

We focused on three types of inputs: (1.) structured synthetic data sets in $\mathbb{R}^2$ and $\mathbb{R}^3$, where direct data inspection allowed us to check our results visually; (2.) simulated alanine-dipeptide protein conformations in $\mathbb{R}^{21}$, where the knowledge of the intrinsic parameters of the simulation allowed us to check our results *a posteriori*; (3.) image pixels distributions in color space, where the quality of the clustering could be checked visually on the resulting image segmentation. In our experiments we used the two estimators mentioned in Section 3: truncated Gaussian kernel and *distance to a measure*. Our implementation was done in C++, and it was run on a PC with 8 CPU cores running at 2.4 GHz and 8 GB of RAM[7]. The code is publicly available at the following address: http://geometrica.saclay.inria.fr/data/ToMATo/.

5.1. Synthetic Data

Our first data set consists of 10k points sampled from two twin spirals in the unit square, shown in Figure 2 (a). Using a δ-Rips graph, with $\delta = 0.04$, and the *distance to a measure* density estimator, we obtain the PD in Figure 2(c). Choosing τ by the gap heuristic we obtain the clustering shown in Figure 2(d). A smaller Rips parameter, $\delta = 0.02$, gives many infinitely persistent components (Figure 6(a)), with all but one appearing late in the PD (near the lower-left corner). Components in this part of the PD are discarded by the extra filtering step performed by the algorithm after completion of the merging phase, which removes much of the background noise (Figure 6(b)).

We also experimented with the k-nn graph (taking $k = 35$) and the Delaunay graph. The obtained PDs are shown in Figure 7. Although not identical, they share the same overall structure with 2 prominent clusters, and the resulting clusterings are virtually identical to Figure 2(d).

[7] Each run used only one core and a fraction of the available memory.

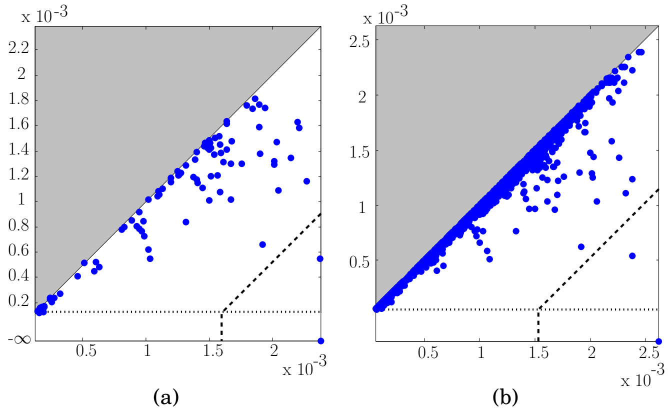


Fig. 7. PDs obtained on the twin spirals data set of Figure 2 using (a) the k-nn graph with $k = 35$ and (b) the Delaunay graph. The resulting clusterings are virtually the same as in Figure 2(d).

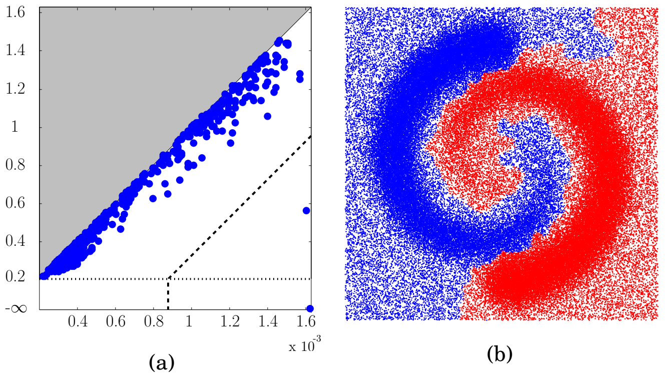


Fig. 8. The twin spirals data set with 100k points, processed using the Rips graph: (a) the persistence diagram; (b) the final clustering.

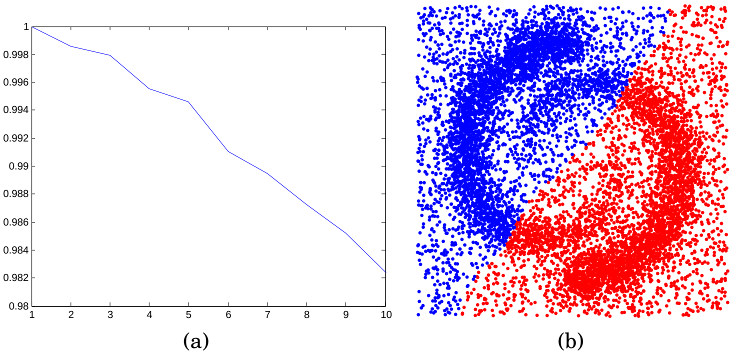
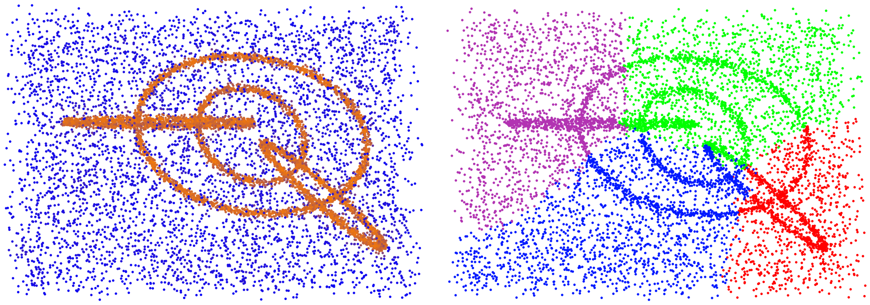


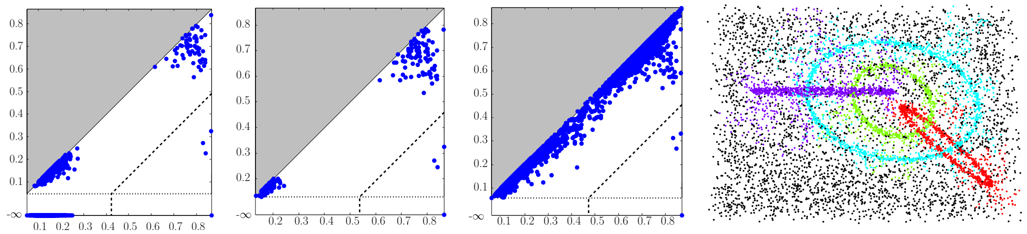
Fig. 9. Result of spectral clustering on the twin spirals data set with 10k samples: (a) a plot of the first 10 eigenvalues, and (b) the obtained clustering.

To illustrate the scalability of our approach, we generated a second data set with about 100k samples from the same probability distribution. It only took ToMATo a few seconds to cluster this data set using the Rips graph. The result is shown in Figure 8.



(a) (b)

Fig. 10. (a) The rings data set with the estimated density function. (b) The result obtained using spectral clustering.



(a) (b) (c) (d)

Fig. 11. Outputs of ToMATo on the rings data set: the obtained PDs with (a) δ-Rips graph, (b) k-nn graph, and (c) Delaunay graph. (d) Clustering obtained with the δ-Rips graph.

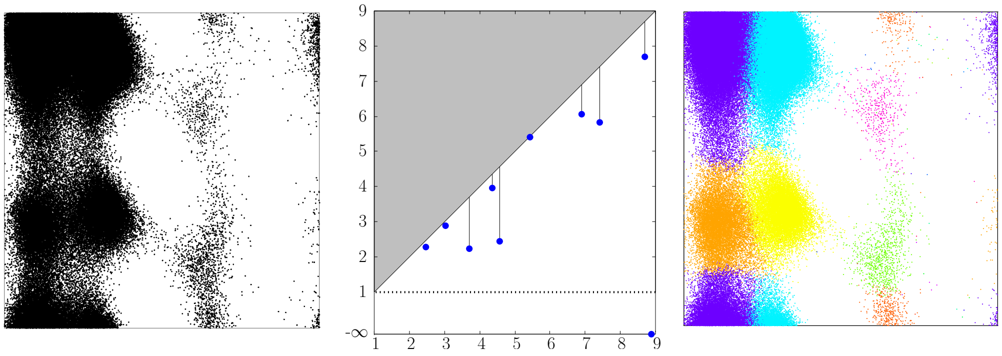
The PD is much better separated than previously because the approximation of the PD of the underlying density function provably improves as the number of samples increases, as stated in our theoretical results.

For comparison, we ran spectral clustering [Chen et al. 2008] on the twin spirals data set with 10k samples, using the k-nn graph. The result, shown in Figure 9, was consistent across choices of input parameters. It is explained by the effect of the background noise on the k-means procedure in eigenspace. We were unable to run the code on δ-Rips graphs or on the data set with 100k points because of numerical issues in the eigenvalues computation.

We also considered another synthetic example, made of four noisy interlocked rings in $\mathbb{R}^3$ with uniform background noise added (Figure 10(a)). Spectral clustering again failed on this data set (Figure 10(b)), for the same reason as before. It did obtain correct clusters with much of the background noise removed, but this required significant tweaking of the number of neighbors: too many resulted in bad clustering and too few resulted in numerical instability in the computation. For comparison, Figure 11 shows the outputs of ToMATo.

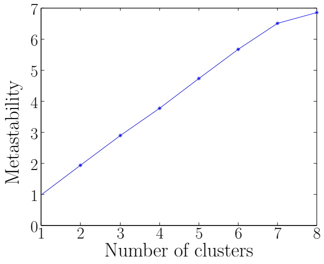
5.2. Alanine-dipeptide conformations

Next we cluster conformations of the alanine-dipeptide molecule. The data consist of short trajectories of conformations generated by atomistic simulations of this small protein [Chodera et al. 2006]. Accurate simulation by molecular dynamics must be done at the atomic scale, generally limiting the length of simulations to picoseconds because of the small time steps needed to integrate stiff bond length and angle potentials. Biologically interesting dynamics, however, often occur on the scale of mil-



(a) (b) (c)

Fig. 12. Biological data set: (a) input point cloud, projected down to the (ϕ, ψ) domain for visualization purposes; (b) output PD represented on a log-log scale; (c) output clustering with 7 clusters.



(a)

Rank	(Unnormalized) Prominence	Metastability
1	∞	0.99982
2	3827	1.91865
3	1334	2.8813
4	557	3.76217
5	85	4.73838
6	32	5.65553
7	26	6.50757
8	7.2	6.8193
9	3.0	–
10	2.2	–

(b)

Fig. 13. Quantitative evaluation of the quality of the output of ToMATo on the biological data set: (a) metastability of the obtained clustering versus the number of clusters; (b) corresponding intervals sorted by decreasing prominence.

liseconds. One solution to this issue is to generate a coarser model using *metastable* states [Huisinga and Schmidt 2005]. These are conformational clusters between which transitions are infrequent and independent. Such coarser representations are tractable using Markovian models [Chodera et al. 2007; Chodera et al. 2006; Chodera et al. 2007] while still allowing for useful simulations. A key problem is the discovery of these metastable states.

The alanine-dipeptide was chosen as example because its dynamics are relatively well-understood: it is known that there are only two relevant degrees of freedom, and these are known *a priori*. This makes it possible to visualize the clustering results by projecting the points onto these coordinates which are referred to as ϕ and ψ (Ramachandran plots). In previous work [Chodera et al. 2006], clustering was done manually into 6 clusters. Subsequent work [Chodera et al. 2007] tried to automatically recover these 6 clusters, as we did using our method.

Our input consisted of 960 trajectories, each one made of 200 protein conformations, each conformation being represented as a 21-dimensional vector with 3 coordinates per atom of the protein. For our experiments we took the trajectories and treated the conformations as 192,000 independent samples in $\mathbb{R}^{21}$. The metric used on this point cloud was root-mean-squared deviation (RMSD) after the best possible rigid matching computed using the method of [Theobald 2005]. The RMSD distance matrix was the only input to our clustering scheme. The output is shown in Figure 12.

It appears from the persistence diagram that there could be anywhere from 4 to 7 clusters. The first 4 clusters are much more prominent than the following 3 clusters. Since there is clearly a multiscale behavior, we plot the PD on a log-log scale. From this perspective, the first 4 clusters are still prominent but relative to their height the 5th and 6th clusters are prominent as well. While the 7th cluster is not as prominent, it is still more prominent than the following clusters, suggesting that 7 is also a reasonable number of clusters. To confirm this insight we came back to the original problem of finding clusters that maximize the *metastability* (as defined in [Huisinga and Schmidt 2005]): we computed the metastabilities of all our candidate clusterings, and we reported them in the table and plot of Figure 13. These results show that the metastability increases linearly with the number of clusters, up to 7 clusters, after which it starts leveling off. So, choosing 4, 5, 6 or 7 clusters should not affect the metastability significantly, thus confirming the observations made from the PD. This is an example of a scenario where the insights into the number of clusters provided by the PD can be validated by exploiting further application-specific information on the data.

Computing the input RMSD distance matrix took the most time: all pairwise distances between conformations were estimated, which took about a day of computation. In order to save space, for each conformation we only recorded the distances to its 15,000 closest conformations in the matrix. On this input, ToMATo only took a few minutes to run. Meanwhile, the amount of memory used remained approximately constant, which enabled us to make several runs in parallel to find a suitable Rips parameter δ.

5.3. Image Segmentation

Finally we use our approach to segment color images. Turning image segmentation into a clustering problem can be done by mapping the pixels in the image to points in some color space like Luv, where they are to be clustered according to the basins of attraction of the peaks of their underlying density function. The segments in the image are then the pre-images of the clusters through the mapping. This is the approach taken e.g. by Mean-Shift [Comaniciu and Meer 2002]. The reason why Luv is preferred over other color spaces like RGB is because the Euclidean distance in Luv space is known to capture the subjective notion of perceptual difference reasonably well.

Clustering in Luv space is oblivious to proximity relations between pixels in the image, allowing pixels that are far apart in the image to end up in a same cluster. Depending on the context, this property can be viewed either as a feature or as a drawback. Removing it requires to take spatial information into account during the clustering phase, which is usually done by appending the two pixel coordinates to the three color channels, thus yielding a 5-dimensional point cloud. The obvious drawback is that the contributions of color and spatial coordinates must be balanced properly in the computation of distances, because the scales of the color channels and spatial coordinates are unrelated. This is an issue in its own right.

In the context of our method, it is natural to consider the pixels in the image domain and in Luv space separately, building the neighborhood graph G in the image domain while estimating the density in Luv space. An advantage of this approach is that, due to the grid structure of the image, the number of neighbors of a pixel in the graph G is constant, and therefore the graph is sparse. However, applied naively, this approach does not work, since pixels belonging to well-separated high-density areas in Luv space can be neighbors in the image, thus leading to the premature merge of some of these areas by the algorithm. For instance, consider a black-and-white image with the same number of black and white pixels. Then, the data points in Luv space are gathered at two distinct hotspots: the black spot, and the white spot. Now, the density function is constant over the image domain, and since black and white regions are neighbors in the image, they all get merged together (resulting in a single cluster) whatever small positive value is assigned to the prominence threshold τ, and regardless of the actual black and white patterns in the image.

To overcome this defect, we modify the proximity rule used for building G as follows, so that it also takes color information into account: two pixels are connected in G if and only if they are close both in the image domain and in Luv space. In practice the spatial constraint is checked first, so that the neighborhoods of the data points have constant size from the beginning. Typically, in practice we used 5×5 windows in the image domain, and the graph construction and clustering phases took barely more than a second each on images with a few hundreds of thousands of pixels. Computing the truncated Gaussian estimator in Luv space was more expensive, however it only took 10 to 20 seconds on each image using the ANN library [Mount and Arya 2010] for proximity queries.

Since natural images have textures, the corresponding point clouds in Luv space contain lots of very small clusters independent from the rest of the data. As a result of our proximity rule, the outputs of ToMATo also contained a lot of very small clusters, which we simply discarded in a post-processing step—in practice, all clusters containing fewer than 100 points were discarded, and the corresponding pixels were marked in black in the segmented images.

The results obtained with this approach are shown in Figure 14. For each input image we show a histogram of the prominences of the peaks detected by the algorithm (ignoring the highest peak, whose prominence is infinite), as well as the segments obtained after choosing a suitable value for parameter τ (this value is indicated by the arrow in each histogram). The segments are shown in fake colors, so the segmentation structure is better highlighted: for instance, one can see that on the mandrill image the algorithm discriminated the left cheek from the right cheek and the left eye from the right eye, due to their separation in the image domain. Again, the black pixels in the segmentation results do not correspond to a single cluster, but rather to a myriad of clusters with fewer than 100 points each, which were discarded in a post-processing step. Focusing now on the histograms, observe that none of them exhibits a clear prominence gap. Instead, they exhibit a series of smaller gaps, which suggests that the correct number of clusters may not be readily identified, thus following the widely accepted idea that image segmentation is an ill-posed problem. Nevertheless, the histograms still provide a precise understanding of the relationship between the choice of parameter τ and the number of obtained segments on each image.

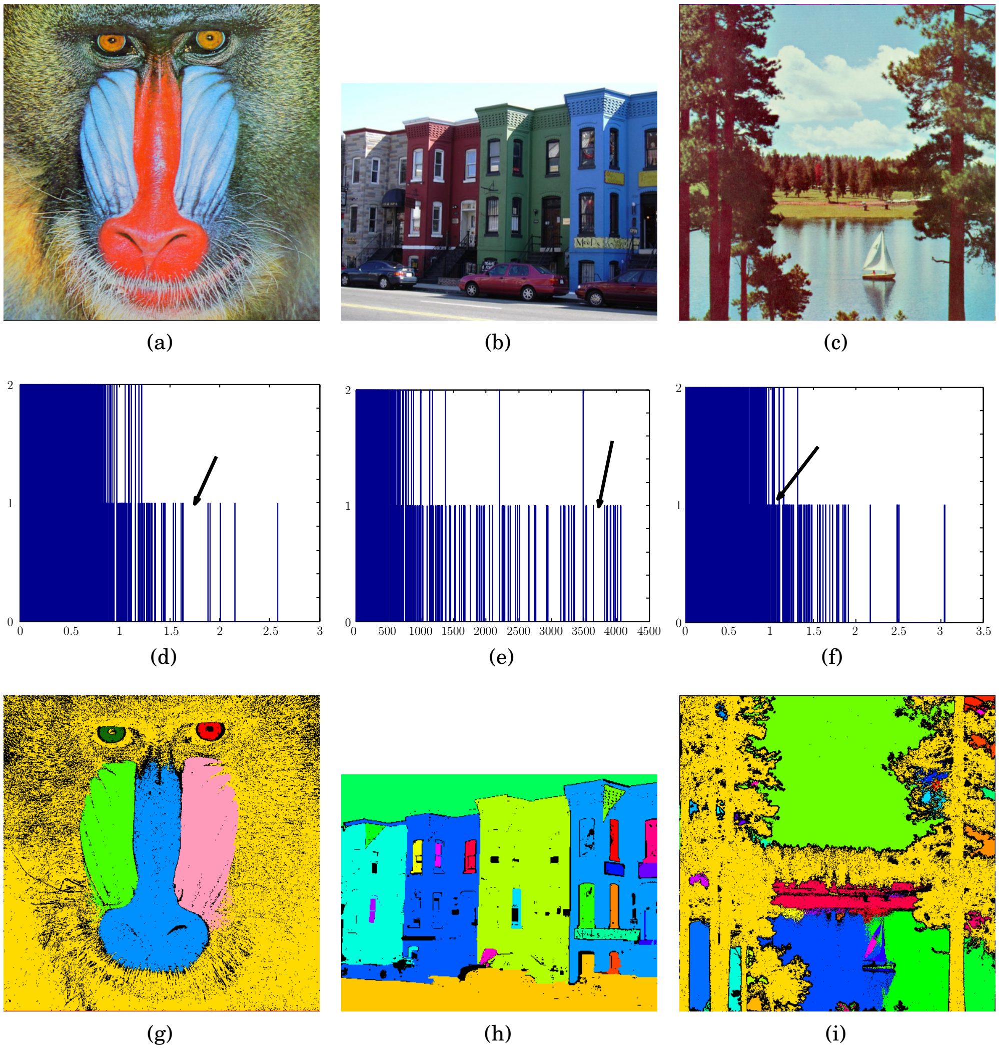


Fig. 14. Outputs of ToMATo in color image segmentation. Top row: input color images. Middle row: histograms of the prominences of the peaks of the estimated density in the neighborhood graph. Each arrow shows the choice of parameter τ made by the user. Bottom row: segmentation obtained after re-running the clustering algorithm with the chosen value of τ.

PART II: *THEORETICAL ANALYSIS*

Our analysis makes consistent use of topological persistence theory, as introduced in [Edelsbrunner et al. 2001] and later developed in [Edelsbrunner et al. 2002; Zomorodian and Carlsson 2005]. We therefore begin this part of the paper with a brief description of the theory (Section 6), referring to two recent surveys [Chazal and Cohen-Steiner 2007; Edelsbrunner and Harer 2007] for further details.

The analysis *per se* is then carried out in Sections 7 through 11, where $\mathbb{X}$, f and P denote the following mathematical objects:

- $\mathbb{X}$ is an m-dimensional Riemannian manifold with positive convexity radius $\varrho(\mathbb{X})$,
- $f : \mathbb{X} \to \mathbb{R}$ is a c-Lipschitz probability density function with respect to the m-dimensional Hausdorff measure on $\mathbb{X}$,
- P is a finite set of points sampled over $\mathbb{X}$ according to f in i.i.d. fashion.

In Sections 7 through 10 we consider a simplified model for our input, where the values of f at the points of P and the pairwise geodesic distances between these points are assumed to be known exactly. We also take the δ-Rips graph $R_\delta(P)$ as the neighborhood graph used by the algorithm. The analysis proceeds as follows:

1. we show that some superlevel-set of f is densely sampled by P with high probability (Section 7),
2. under this condition and a relevant choice of parameter δ, we show that the persistence diagram computed by the clustering algorithm approximates a large part of the persistence diagram of f (Section 8),
3. we deduce that the algorithm can recover the correct number of clusters under some sufficient signal-to-noise ratio condition on the persistence diagram of f (Section 9),
4. we show that under the same condition the clusters computed by the algorithm approximate the stable parts of the basins of attraction of the peaks of f (Section 10).

Then, in Section 11 we consider a more realistic model for our input, where density values and geodesic distances are known with some small uncertainty, and we study the stability of the output of the algorithm with respect to small perturbations of the input.

6. BACKGROUND ON TOPOLOGICAL PERSISTENCE

We use singular homology with coefficients in a commutative ring, assumed to be a field and omitted in our notations. We refer the reader to [Hatcher 2001] for a thorough introduction to homology theory.

A *persistence module* $\mathcal{X}$ is a finite directed system of finite-dimensional vector spaces connected by linear maps:

$$X^m \longrightarrow X^{m-1} \longrightarrow \cdots \longrightarrow X^1 \longrightarrow X^0.$$

The structure of this system is encoded as a planar point set, called the *persistence diagram* of $\mathcal{X}$ and noted $\mathrm{D}\mathcal{X}$. Formally, $\mathrm{D}\mathcal{X}$ is defined as a multi-set of points in the extended plane $\overline{\mathbb{R}}^2$, where $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$, contained in the union of the extended diagonal $\Delta = \{(x,x) : x \in \overline{\mathbb{R}}\}$ and of the extended grid $\{(i,j) : m \geq i > j \geq 0\} \cup \{(i,-\infty) : m \geq i \geq 0\}$. The multiplicities of the points of Δ are set to $+\infty$, while the multiplicities of the grid points are defined by finite alternating sums of ranks of composed homomorphisms $X^l \to X^k$, $l > k$ [Chazal et al. 2009; Cohen-Steiner et al. 2005]. Since all the spaces are finite-dimensional, these ranks are finite, and so the diagram $\mathrm{D}\mathcal{X}$ only contains finitely many points off the diagonal Δ. Intuitively, every such point (i, j) encodes the lifespan of some generator appearing at time i and dying at time $j < i$ in the sequence of vector spaces[8].

In the following we consider persistence modules defined by continuous sequences of vector spaces $\{X^\alpha\}_{\alpha\in\mathbb{R}}$, connected by linear maps $X^\alpha \to X^\beta$ for all $\alpha \geq \beta$, such that $X^\alpha \to X^\alpha$ is the identity map and $X^\alpha \to X^\beta \to X^\gamma$ commute with $X^\alpha \to X^\gamma$ for all $\alpha \geq \beta \geq \gamma$. The definition of persistence diagram can be extended to this continuous setting via a limit process [Chazal et al. 2009], under some *tameness* condition stating that the homomorphisms $X^\alpha \to X^\beta$ have finite ranks for all $\alpha > \beta$. Under this condition, the persistence diagram $\mathrm{D}\mathcal{X}$ may contain infinitely many points off the extended diagonal Δ, however all its accumulation points belong to Δ, so $\mathrm{D}\mathcal{X}$ is finite outside any offset of Δ.

A natural measure of proximity between persistence diagrams is the bottleneck distance [Cohen-Steiner et al. 2005]. Given two tame persistence modules $\mathcal{X}$ and $\mathcal{Y}$, a *multi-bijection* γ between $\mathrm{D}\mathcal{X}$ and $\mathrm{D}\mathcal{Y}$ is a bijection

$$\gamma : \bigcup_{p\in|\mathrm{D}\mathcal{X}|} \coprod_{i=1}^{\mu(p)} p \to \bigcup_{q\in|\mathrm{D}\mathcal{Y}|} \coprod_{i=1}^{\mu(q)} q,$$

where $|\mathrm{D}\mathcal{X}|$ denotes the *support* of $\mathrm{D}\mathcal{X}$, *i.e.* the set $\mathrm{D}\mathcal{X}$ considered as a subset of $\overline{\mathbb{R}}^2$ without any multiplicities, and where $\mu(p)$ denotes the multiplicity of point $p \in |\mathrm{D}\mathcal{X}|$ in $\mathrm{D}\mathcal{X}$. Note that such bijections always exist since the points on the diagonal Δ have infinite multiplicities. The bottleneck distance $\mathrm{d}_B^\infty(\mathrm{D}\mathcal{X}, \mathrm{D}\mathcal{Y})$ between $\mathrm{D}\mathcal{X}$ and $\mathrm{D}\mathcal{X}$ is the quantity $\min_\gamma \max_{p\in\mathrm{D}\mathcal{X}} \|p - \gamma(p)\|_\infty$, where γ ranges over all multi-bijections between $\mathrm{D}\mathcal{X}$ and $\mathrm{D}\mathcal{Y}$, and where $\|\cdot\|_\infty$ denotes the l^∞-norm.

Stability is an important property of persistence diagrams. It can be stated in terms of a measure of proximity between persistence modules called *interleaving* [Chazal et al. 2009]. Formally, two tame persistence modules $\mathcal{X}$ and $\mathcal{Y}$ are *(strongly) ε-interleaved* if there exist two families of homomorphisms $\{\phi_\beta : X^\beta \to Y^{\beta-\varepsilon}\}_{\beta\in\mathbb{R}}$ and $\{\psi_\beta : Y^\beta \to X^{\beta-\varepsilon}\}_{\beta\in\mathbb{R}}$, such that for all values $\beta' \geq \beta$ the following diagrams of vector spaces commute:

$$\begin{tikzcd}
X^{\beta'+\varepsilon} \arrow[rrr] \arrow[dr, "\phi_{\beta'+\varepsilon}"'] & & & X^{\beta-\varepsilon} & & X^{\beta'-\varepsilon} \arrow[r] & X^{\beta-\varepsilon} \\
& Y^{\beta'} \arrow[r] & Y^\beta \arrow[ur, "\psi_\beta"'] & & Y^{\beta'} \arrow[ur, "\psi_{\beta'}"] \arrow[r] & Y^\beta \arrow[ur, "\psi_\beta"'] & \\
& X^{\beta'} \arrow[r] & X^\beta \arrow[dr, "\phi_\beta"] & & X^{\beta'} \arrow[r] \arrow[dr, "\phi_{\beta'}"'] & X^\beta \arrow[dr, "\phi_\beta"] & \\
Y^{\beta'+\varepsilon} \arrow[rrr] \arrow[ur, "\psi_{\beta'+\varepsilon}"] & & & Y^{\beta-\varepsilon} & & Y^{\beta'-\varepsilon} \arrow[r] & Y^{\beta-\varepsilon}
\end{tikzcd} \tag{2}$$

Intuitively, the commutativity of these diagrams means that every generator appearing (resp. dying) in $\mathcal{X}$ at a given time $\beta \in \mathbb{R}$ must appear (resp. die) in $\mathcal{Y}$ within the time range $[\beta-\varepsilon, \beta+\varepsilon]$, and vice-versa. The currently most general stability theorem in persistence theory says that any ε-interleaved pair of tame persistence modules has ε-close persistence diagrams in the bottleneck distance [Chazal et al. 2009; Chazal et al. 2012].

[8] Note that we depart from the usual way of introducing persistence by reversing the time flow, which goes from $+\infty$ to $-\infty$ here. This choice is purely formal and does not affect the validity of the theory.

In the context of clustering, we will primarily focus on persistence modules $\mathcal{X}$ induced at 0-dimensional homology level by the sequence of superlevel-sets of a real-valued function f. Consider the nested family of closed superlevel-sets $F^\alpha = f^{-1}([\alpha, +\infty))$, and take for $\{X^\alpha\}_{\alpha\in\mathbb{R}}$ the induced family of 0-dimensional homology groups $H_0(F^\alpha)$, connected by the homomorphisms $H_0(F^\alpha) \to H_0(F^\beta)$ induced by the canonical inclusions $F^\alpha \hookrightarrow F^\beta$ for all $\alpha \geq \beta$. This persistence module encodes the evolution of the path-connectivity of the superlevel-sets F^α as parameter α decreases from $+\infty$ to $-\infty$, and its persistence diagram $\mathrm{D}\mathcal{X}$ is precisely what we called the persistence diagram of f (noted $\mathrm{D}_0 f$) in the first part of the paper.

7. SAMPLING THE SUPERLEVEL-SETS OF f

In our analysis we use the following classical notion of sampling density, where $\mathrm{d}_\mathbb{X}$ denotes the geodesic distance in the Riemannian manifold $\mathbb{X}$:

Definition 7.1. Given a subset $\mathbb{Y} \subseteq \mathbb{X}$ and a parameter $\varepsilon > 0$, P is a *geodesic ε-sample of* $\mathbb{Y}$ if every point of $\mathbb{Y}$ lies within geodesic distance ε of P, that is: $\forall y \in \mathbb{Y}$, $\min_{p\in P} \mathrm{d}_\mathbb{X}(y, p) \leq \varepsilon$.

Since the points of P are drawn according to f in i.i.d. fashion, the more points are drawn the more chances we have that P satisfies the above condition over some prescribed superlevel-set F^α. This simple fact is proved formally in Theorem 7.2 below. Before stating the theorem, we need to introduce a few measure-theoretic quantities. Given a subset A of $\mathbb{X}$ and a parameter $r > 0$, let $\mathcal{V}_r(A) \geq 0$ denote the infimum of the Hausdorff measures achieved by geodesic balls of radius r centered in A, that is:

$$\mathcal{V}_r(A) = \inf_{x\in A} \mathcal{H}^m(B_\mathbb{X}(x, r)), \text{ where } B_\mathbb{X}(x, r) = \{y \in \mathbb{X},\ \mathrm{d}_\mathbb{X}(x, y) \leq r\}. \tag{3}$$

Let also $\mathcal{N}_r(A) \in \mathbb{N} \cup \{+\infty\}$ be the *r-covering number* of A, that is, the minimum number of closed geodesic balls of same radius r needed to cover A (the balls do not have to be centered in A).

THEOREM 7.2. *Let $\mathbb{X}$ be an m-dimensional Riemannian manifold, and $f : \mathbb{X} \to \mathbb{R}$ a c-Lipschitz probability density function. Consider a set P of n points sampled according to f in i.i.d. fashion. Then, for any parameters $\varepsilon > 0$ and $\alpha > c\varepsilon$, we are guaranteed that P forms an ε-sample of F^α with probability at least $1 - \mathcal{N}_{\varepsilon/2}(F^\alpha)\ e^{-n(\alpha - c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha)}$.*

PROOF. If $\mathcal{N}_{\varepsilon/2}(F^\alpha) = +\infty$ or $\mathcal{V}_{\varepsilon/2}(F^\alpha) = 0$, then the lower bound on the probability of success given in the conclusion is non-positive, therefore its holds trivially.

Assume from now on that $\mathcal{N}_{\varepsilon/2}(F^\alpha) < +\infty$ and $\mathcal{V}_{\varepsilon/2}(F^\alpha) > 0$. Consider a family $\{B_i\}_{1\leq i\leq l}$ of closed geodesic balls of same radius $\frac{\varepsilon}{2}$ such that $F^\alpha \subseteq \bigcup_{i=1}^{l} B_i$ and $l = \mathcal{N}_{\varepsilon/2}(F^\alpha)$ is minimal. For each integer i in the range $[1, l]$, let p_i be a point of $B_i \cap F^\alpha$. Such a point exists because otherwise the cover would not be minimal. Since f is c-Lipschitz, at every point $x \in B_i$ we have $f(x) \geq f(p_i) - c\ \mathrm{d}_\mathbb{X}(x, p_i) \geq \alpha - c\varepsilon > 0$. Therefore,

$$\forall i \in \{1, \cdots, l\},\ \int_{B_i} f\ d\mathcal{H}^m \geq (\alpha - c\varepsilon)\mathcal{H}^m(B_i) \geq (\alpha - c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha).$$

Let E_i denote the event that $P \cap B_i = \emptyset$. Then, $\cup_i E_i$ is the event that at least one ball B_i contains no point of P. When the complement of this event occurs, the triangle inequality tells us that P is a geodesic ε-sample of F^α, and so our goal is to work out an upper bound on the probability $\Pr[\cup_i E_i]$. For each event E_i taken separately, we have

$$\Pr[E_i] = \left(1 - \int_{B_i} f\ d\mathcal{H}^m\right)^n \leq \left(1 - (\alpha - c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha)\right)^n.$$

Then, by the union bound, we have

$$\Pr[\cup_i E_i] \leq \sum_{i=1}^{l} \Pr[E_i] \leq l\left(1-(\alpha-c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha)\right)^n.$$

Observe now that the quantity $e^{-x}+x-1$ is non-negative for all $x \geq 0$. Letting x be equal to $(\alpha-c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha)$, we obtain

$$1-(\alpha-c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha) \leq e^{-(\alpha-c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha)},$$

which implies

$$\Pr[\cup_i E_i] \leq l\left(1-(\alpha-c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha)\right)^n \leq l\ e^{-n(\alpha-c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha)} = \mathcal{N}_{\varepsilon/2}(F^\alpha)\ e^{-n(\alpha-c\varepsilon)\mathcal{V}_{\varepsilon/2}(F^\alpha)}.$$

□

Theorem 7.2 can be interpreted in various different ways:

- When the probability density function f is given and a fixed superlevel-set F^α ($\alpha > 0$) is considered, the theorem ensures that after drawing sufficiently many points according to f in i.i.d. fashion the superlevel set F^α will be densely sampled with high probability.
- Conversely, when the set P of sample points is fixed and a target sampling parameter ε is given, the theorem ensures that for large enough values[9] of α the superlevel-set F^α is ε-sampled by P with high probability. In particular, α has to be larger than $c\varepsilon$.

In both scenarios, the probability of success is influenced by two quantities that are intrinsic to the Riemannian manifold $\mathbb{X}$: the covering number $\mathcal{N}_{\varepsilon/2}(F^\alpha)$, and the minimum geodesic ball measure $\mathcal{V}_{\varepsilon/2}(F^\alpha)$. In particular, the probability of success can be positive only when $\mathcal{N}_{\varepsilon/2}(F^\alpha)$ is finite and $\mathcal{V}_{\varepsilon/2}(F^\alpha)$ is positive, two conditions that are met by a large class of Riemannian manifolds $\mathbb{X}$, including the ones with bounded absolute sectional curvature (among which are the compact Riemannian manifolds and the Euclidean spaces):

LEMMA 7.3. *If $\mathbb{X}$ is a complete Riemannian manifold with bounded absolute sectional curvature, then for any $\alpha > 0$ and any $\varepsilon < 2\varrho(\mathbb{X})$ we have $\mathcal{N}_{\varepsilon/2}(F^\alpha) < +\infty$ and $\mathcal{V}_{\varepsilon/2}(F^\alpha) > 0$.*

PROOF. Let $\alpha > 0$. Since $\mathbb{X}$ is complete with bounded absolute sectional curvature, the Bishop-Gunther inequality [Gallot et al. 2004, Theorem 3.101] ensures that $\mathcal{V}_r(\mathbb{X}) > 0$ for all values r within the range $(0, \varrho(\mathbb{X}))$. This holds in particular for $r = \varepsilon/2$, and so we have $\mathcal{V}_{\varepsilon/2}(F^\alpha) \geq \mathcal{V}_{\varepsilon/2}(\mathbb{X}) > 0$.

To show that $\mathcal{N}_{\varepsilon/2}(F^\alpha)$ is finite, take any $\frac{\varepsilon}{4}$-packing of F^α, i.e. any set $S \subseteq F^\alpha$ such that $\mathrm{d}_{\mathbb{X}}(s,s') > \frac{\varepsilon}{2}$ for all pairs of points $s, s' \in S$, $s \neq s'$. Let $r = \min\{\frac{\alpha}{2c}, \frac{\varepsilon}{4}\} > 0$. Since f is c-Lipschitz, we have

$$\forall s \in S,\ \forall x \in B_{\mathbb{X}}(s,r),\ f(x) \geq f(s) - cr \geq \frac{\alpha}{2},$$

which means that the geodesic ball $B_{\mathbb{X}}(s,r)$ is included in the superlevel-set $F^{\alpha/2}$. Moreover, the geodesic balls in the collection $\{B_{\mathbb{X}}(s,r)\}_{s\in S}$ are pairwise-disjoint since $r \leq \frac{\varepsilon}{4}$ and S is an $\frac{\varepsilon}{4}$-packing. As a result, we have

$$\mathcal{H}^m(F^{\alpha/2}) \geq \mathcal{H}^m\left(\bigcup_{s\in S} B_{\mathbb{X}}(s,r)\right) = \sum_{s\in S} \mathcal{H}^m\left(B_{\mathbb{X}}(s,r)\right) \geq \mathcal{V}_r(F^\alpha)\ |S|. \tag{4}$$

[9] As α grows, $\mathcal{N}_{\varepsilon/2}(F^\alpha)$ decreases while $\mathcal{V}_{\varepsilon/2}(F^\alpha)$ increases, therefore the probability of success increases.

Now, since f is a probability density function, we have

$$1 = \int_{\mathbb{X}} f \; d\mathcal{H}^m \geq \int_{F^{\alpha/2}} f \; d\mathcal{H}^m \geq \frac{\alpha}{2} \; \mathcal{H}^m(F^{\alpha/2}). \tag{5}$$

It follows from Eqs. (4)-(5) that $|S| \leq \frac{2}{\alpha \; \mathcal{V}_r(F^\alpha)}$. Since this inequality holds for any $\frac{\varepsilon}{4}$-packing S of F^α, we conclude by the Kolmogorov-Tikhomirov inequality [Kolmogorov and Tikhomirov 1961] that $\mathcal{N}_{\varepsilon/2}(F^\alpha) \leq \frac{2}{\alpha \; \mathcal{V}_r(F^\alpha)}$, which is finite since both α and $\mathcal{V}_r(F^\alpha)$ are positive. □

8. APPROXIMATING THE PERSISTENCE DIAGRAM OF f

Recall that in our analysis we are assuming the neighborhood graph used by the clustering algorithm to be the δ-Rips graph $R_\delta(P)$. In this section, we are also assuming that the merging parameter τ is set to $+\infty$.

During the merging phase (described in Section 2), the algorithm builds a nested family of subgraphs of $R_\delta(P)$ by inserting the vertices one at a time, in decreasing order of their function values. Each time a vertex v is inserted, all the edges of its *upper star* (*i.e.* the edges of $R_\delta(P)$ that connect v to vertices with higher function values) are inserted as well. We call this family the *upper-star Rips filtration*, noted $\mathcal{R}^f_\delta(P)$, and we write it formally as follows:

$$\mathcal{R}^f_\delta(P) = \{R_\delta(P \cap F^\alpha)\}_{\alpha \in \mathbb{R}},$$

where $R_\delta(P \cap F^\alpha)$ is the δ-Rips graph of the vertex subset $P \cap F^\alpha$, and where parameter α decreases from $+\infty$ to $-\infty$. Since each graph $R_\delta(P \cap F^\alpha)$ is finite, the family $\mathcal{R}^f_\delta(P)$ induces a tame persistence module at 0-dimensional homology level. The persistence diagram output by the algorithm is precisely the persistence diagram of this module, noted $\mathrm{D}_0\mathcal{R}^f_\delta(P)$, and our goal is to determine to what extent it is close to $\mathrm{D}_0 f$.

This scenario is reminiscent of the one considered in [Chazal et al. 2011], where the following approximation result was proven[10]:

THEOREM 8.1 ([CHAZAL ET AL. 2011]). *Let $\mathbb{X}$ be a compact Riemannian manifold, possibly with boundary, and $f : \mathbb{X} \to \mathbb{R}$ a c-Lipschitz function. Let also P be a geodesic ε-sample of $\mathbb{X}$. If $\varepsilon < \frac{1}{4}\varrho(\mathbb{X})$, then for any $\delta \in [4\varepsilon, \; \varrho(\mathbb{X}))$, the bottleneck distance between $\mathrm{D}_0 f$ and $\mathrm{D}_0\mathcal{R}^f_\delta(P)$ is at most $c\delta$.*

Unfortunately, this result is not directly applicable in our context because our scenario differs in the following crucial ways:

1. in our case the manifold $\mathbb{X}$ may not be compact, for instance when it is some Euclidean space $\mathbb{R}^m$;
2. in our case the point cloud P may not be dense over the entire manifold $\mathbb{X}$, especially when the points are drawn from a probability distribution whose support does not cover $\mathbb{X}$ entirely.

Our main result (Theorem 8.2 below) addresses these two issues, assuming that the point cloud P forms a dense sampling of some superlevel-set of the function f, as guaranteed with high probability by Theorem 7.2. In the statement of the theorem, Q^{NE}_α, Q^{SE}_α , Q^{SW}_α , and Q^{NW}_α denote respectively the quadrants $(\alpha, +\infty] \times (\alpha, +\infty]$, $(\alpha, +\infty] \times [-\infty, \alpha]$, $[-\infty, \alpha] \times [-\infty, \alpha]$, and $[-\infty, \alpha] \times (\alpha, +\infty]$ in the extended plane $\overline{\mathbb{R}}^2$.

[10] The result of [Chazal et al. 2011] holds in fact for homology groups of arbitrary dimensions, but it uses two upper-star Rips filtrations in parallel in the algorithm: $\mathcal{R}^f_{\delta/2}(P)$ and $\mathcal{R}^f_\delta(P)$. As reported in Section 4.3 of that paper, in the special case of 0-dimensional homology, using both filtrations or only $\mathcal{R}^f_\delta(P)$ gives exactly the same results.

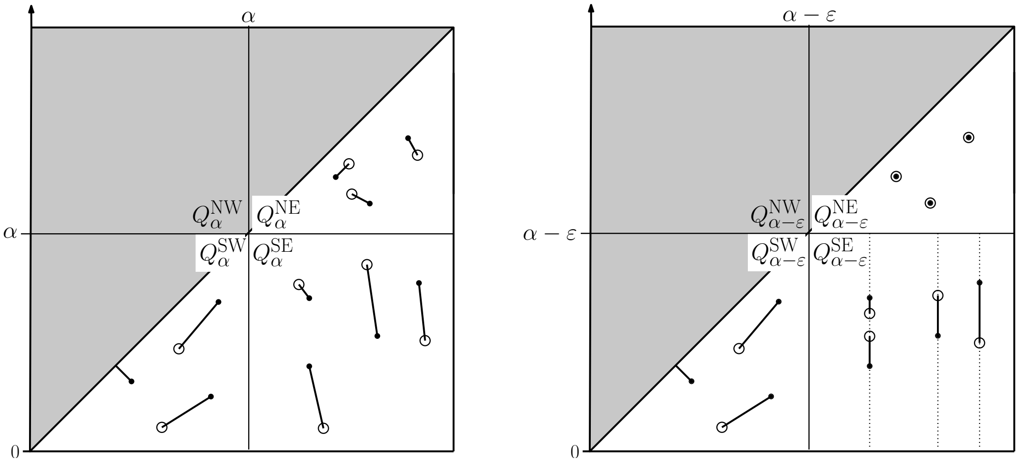

α
α
0
Q_α^{NW}
Q_α^{NE}
Q_α^{SW}
Q_α^{SE}
α − ε
α − ε
0
$Q_{\alpha-\varepsilon}^{\mathrm{NW}}$
$Q_{\alpha-\varepsilon}^{\mathrm{NE}}$
$Q_{\alpha-\varepsilon}^{\mathrm{SW}}$
$Q_{\alpha-\varepsilon}^{\mathrm{SE}}$

Fig. 15. Left: the multi-bijection of Theorem 8.2. Right: for the proof of Lemma 8.3.

THEOREM 8.2. *Let $\mathbb{X}$ be a Riemannian manifold, possibly non-compact, possibly with boundary. Assume that its convexity radius $\varrho(\mathbb{X})$ is positive. Let $P \subseteq \mathbb{X}$ be a finite point cloud and $f : \mathbb{X} \to \mathbb{R}$ a c-Lipschitz function. Then, for any positive $\delta < \varrho(\mathbb{X})$, for any $\alpha \in \mathbb{R}$ such that P is a geodesic $\frac{\delta}{4}$-sample of $F^\alpha = f^{-1}([\alpha, \infty))$, there is a multi-bijection $\gamma : \mathrm{D}_0 f \to \mathrm{D}_0 \mathcal{R}_\delta^f(P)$ such that:*

(i) $\forall p \in \mathrm{D}_0 f \cap Q_\alpha^{\mathrm{NE}}$, $\|p - \gamma(p)\|_\infty \leq c\delta$.

(ii) $\forall q \in \mathrm{D}_0 \mathcal{R}_\delta^f(P) \cap Q_\alpha^{\mathrm{NE}}$, $\|\gamma^{-1}(q) - q\|_\infty \leq c\delta$.

(iii) $\forall p \in \mathrm{D}_0 f \cap Q_\alpha^{\mathrm{SE}}$, $|p_x - \gamma(p)_x| \leq c\delta$.

(iv) $\forall q \in \mathrm{D}_0 \mathcal{R}_\delta^f(P) \cap Q_\alpha^{\mathrm{SE}}$, $|\gamma^{-1}(q)_x - q_x| \leq c\delta$.

The theorem is illustrated in Figure 15 (left). Assertions (i)-(ii) ensure that the multi-bijection γ does not move the points of both diagrams by more than $c\delta$ within the upper-right quadrant Q_α^{NE} corresponding to the superlevel-set of f that is $\frac{\delta}{4}$-sampled by P. In cases where P is a $\frac{\delta}{4}$-sample of the entire manifold $\mathbb{X}$ ($\alpha = -\infty$), assertions (i)-(ii) imply that the bottleneck distance between both persistence diagrams is at most $c\delta$, as stated in Theorem 8.1.

Assertions (iii)-(iv) provide weaker guarantees in the lower-right quadrant Q_α^{SE}, by ensuring that every 0-dimensional homology generator appearing at time $\alpha_b > \alpha$ in the superlevel-sets filtration of f must appear within $[\alpha_b - c\delta, \alpha_b + c\delta]$ in the upper-star filtration $\mathcal{R}_\delta^f(P)$, and vice-versa. By contrast, death times are not fully controlled: if the homology generator dies at time $\alpha_d < \alpha$ in the superlevel-sets filtration of f, then all we can say is that its death time in $\mathcal{R}_\delta^f(P)$ must be less than $\alpha + c\delta$, because if it were not then by (ii) the point of $\mathrm{D}_0 f$ corresponding to the generator would be located in Q_α^{NE} instead of Q_α^{SE}.

Finally, due to the potentially low sampling density outside the superlevel-set F^α, there is no guarantee concerning the portion of $\mathrm{D}_0 f$ lying in the quadrant Q_α^{SW} located to the left of the vertical line $x = \alpha$. This part of the diagram corresponds indeed to homological generators appearing at times less than α in the superlevel-sets filtration of f, which may or may not be captured in $\mathcal{R}_\delta^f(P)$.

PROOF OF THEOREM 8.2. The key to the proof of the theorem is the following technical result, whose purely algebraic proof is deferred to Appendix A:

LEMMA 8.3. *Let $\mathcal{X}$ and $\mathcal{Y}$ be two tame persistence modules that are (strongly) ε-interleaved above some given time $\alpha \in \mathbb{R}$. Then, there is a multi-bijection $\gamma : \mathrm{D}\mathcal{X} \to \mathrm{D}\mathcal{Y}$ satisfying assertions* (i) *through* (iv) *of Theorem 8.2, with $\mathrm{D}_0 f$ replaced by $\mathrm{D}\mathcal{X}$, with $\mathrm{D}_0\mathcal{R}^f_\delta(P)$ replaced by $\mathrm{D}\mathcal{Y}$, and with $c\delta$ replaced by ε.*

A few words of explanation are in order. Two tame persistence modules $\mathcal{X}$ and $\mathcal{Y}$ are ε-interleaved *above a given time* $\alpha \in \mathbb{R}$ if there exist two families of homomorphisms $\{\phi_\beta : X^\beta \to Y^{\beta-\varepsilon}\}_{\beta\geq\alpha}$ and $\{\psi_\beta : Y^\beta \to X^{\beta-\varepsilon}\}_{\beta\geq\alpha}$ such that the diagrams of Eq. (2) commute for all values $\beta' \geq \beta \geq \alpha$. Intuitively, the commutativity of these diagrams means that every generator appearing (resp. dying) in $\mathcal{X}$ at some time $\beta \geq \alpha$ must appear (resp. die) in $\mathcal{Y}$ within $[\beta-\varepsilon, \beta+\varepsilon]$, and vice-versa. This statement is the analog of assertions (i)-(ii) of Theorem 8.2. Furthermore, every generator appearing in $\mathcal{X}$ at time $\beta_b \geq \alpha$ and dying at time $\beta_d \leq \alpha$ must appear within $[\beta_b - \varepsilon, \beta_b + \varepsilon]$ and die at some time below $\alpha + \varepsilon$ in $\mathcal{Y}$, and vice-versa. This statement is the analog of assertions (iii)-(iv) of Theorem 8.2.

With Lemma 8.3 at hand, the proof of the theorem becomes a straightforward adaptation of the proof of Theorem 8.1 given in [Chazal et al. 2011]. Indeed, exactly the same sequence of arguments as in [Chazal et al. 2011, §3.1] shows that there exist two families of homomorphisms $\{\phi_\beta : H_0(F^\beta) \to H_0(R_\delta(P \cap F^{\beta-c\delta}))\}_{\beta\geq\alpha}$ and $\{\psi_\beta : H_0(R_\delta(P \cap F^\beta)) \to H_0(F^{\beta-c\delta})\}_{\beta\geq\alpha}$ that make the persistence modules $\{H_0(F^\beta)\}_{\beta\in\mathbb{R}}$ and $\{H_0(R_\delta(P \cap F^\beta))\}_{\beta\in\mathbb{R}}$ (strongly) $c\delta$-interleaved above time α. It follows then from Lemma 8.3 that there is a multi-bijection $\gamma : \mathrm{D}_0 f \to \mathrm{D}_0\mathcal{R}^f_\delta(P)$ satisfying assertions (i) through (iv). □

9. ESTIMATING THE NUMBER OF PROMINENT PEAKS OF f

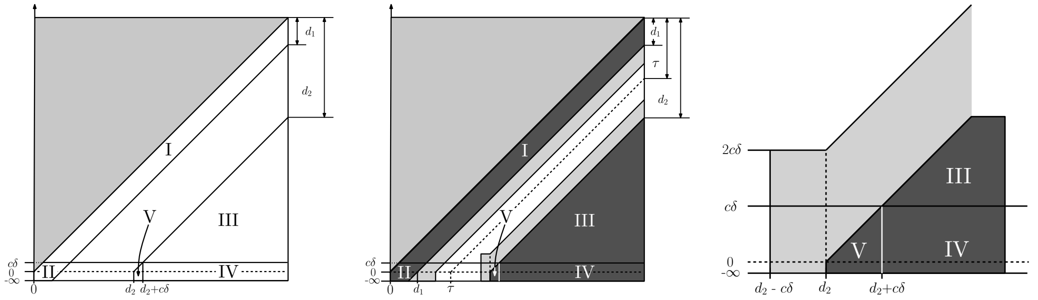
In this section, we prove that the algorithm can recover the correct number of clusters provided that the peaks of the density function f are prominent enough compared to the topological noise. To state the result formally we need to introduce some notation for partitioning the persistence diagram of f.

For any $d > 0$, we call Δ_d the shifted diagonal line $y = x - d$. Let Δ^{S}_d denote the closed half-plane lying below Δ_d, and Δ^{N}_d the open half-plane lying above Δ_d. Similarly, we call Λ^{W}_d (resp. Λ^{E}_d) the closed (resp. open) half-plane lying to the left (resp. right) of the vertical line $x = d$, and Λ^{S}_d (resp. Λ^{N}_d) the closed (resp. open) half-plane lying below (resp. above) the horizontal line $y = d$. Definition 4.1 can now be restated as follows:

Definition 9.1. Given two values $d_2 > d_1 \geq 0$, the persistence diagram of f is called (d_1, d_2)-*separated* if it has the following structure:

$$\mathrm{D}_0 f = \mathrm{D}_1 \cup \mathrm{D}_2, \text{ where } \mathrm{D}_1 \subset \Delta^{\mathrm{N}}_{d_1} \text{ and } \mathrm{D}_2 \subset \Delta^{\mathrm{S}}_{d_2} \cap \Lambda^{\mathrm{E}}_{d_2}.$$

As mentioned after Definition 4.1, the condition that $\mathrm{D}_0 f$ is partitioned into two disjoint subsets $\mathrm{D}_1 \subset \Delta^{\mathrm{N}}_{d_1}$ and $\mathrm{D}_2 \subset \Delta^{\mathrm{S}}_{d_2}$ with $d_2 > d_1$ can be interpreted as a signal-to-noise ratio condition: the relevant peaks of f (in D_2) must be significantly more prominent than the non-relevant ones (in D_1) for the algorithm to be able to detect the correct number of clusters. The additional condition that $\mathrm{D}_2 \subset \Lambda^{\mathrm{E}}_{d_2}$ follows the description of the extra filtering step performed by the algorithm after the merging phase, and it stems from the fact that only some superlevel-set F^α of f can be densely sampled by the input point set P, as expressed in Theorem 7.2. Due to a lack of sample points outside F^α, the persistence diagram of the upper-star Rips filtration built by the al-



(a) Regions of $\mathrm{D}_0 f$ (b) Images of the regions through γ (c) Close-up of Region V

Fig. 16. For the proof of Theorem 9.2.

gorithm cannot be controlled in the region $\Lambda^{\mathrm{W}}_{\alpha}$, which must therefore be discarded as illustrated in Figure 6(a).

THEOREM 9.2. *Let $\mathbb{X}$ be a Riemannian manifold with positive convexity radius, and let $f : \mathbb{X} \to \mathbb{R}$ be a c-Lipschitz probability density function. If $\mathrm{D}_0 f$ is (d_1, d_2)-separated, with $d_2 > d_1 \geq 0$, then for any positive parameter $\delta < \min\{\varrho(\mathbb{X}),\ \frac{d_2-d_1}{5c}\}$ and any threshold $\tau \in (d_1 + 2c\delta,\ d_2 - 3c\delta)$, on any input of n sample points drawn according to f in an i.i.d. fashion the number of clusters computed by the algorithm is equal to the number of peaks of f of prominence at least d_2 with probability at least $1 - \mathcal{N}_{\delta/8}(F^{c\delta})e^{-n\frac{3}{4}c\delta\mathcal{V}_{\delta/8}(F^{c\delta})}$.*

PROOF. Let $\alpha = c\delta$ and $\varepsilon = \delta/4$. According to Theorem 7.2, the input point set P forms a $\frac{\delta}{4}$-sample of the superlevel-set $F^{c\delta}$ with probability at least $1 - \mathcal{N}_{\delta/8}(F^{c\delta})e^{-n\frac{3}{4}c\delta\mathcal{V}_{\delta/8}(F^{c\delta})}$. Assume from now on that P is indeed a $\frac{\delta}{4}$-sample of $F^{c\delta}$. By Theorem 8.2, there is a multi-bijection $\gamma : \mathrm{D}_0 f \to \mathrm{D}_0\mathcal{R}^f_\delta(P)$ satisfying conditions (i) through (iv) of Theorem 8.2. Let us prove that under these conditions the diagram of $\mathcal{R}^f_\delta(P)$ is separated into two parts, one of which is in (multi-)bijection with the set of peaks of f of prominence at least d_2. The proof requires us to analyze where an arbitrary point p of $\mathrm{D}_0 f$ can be mapped to by γ. Referring to Figure 16(a), we split our analysis into five different cases depending on the region of $\mathrm{D}_0 f$ that contains p. We first consider Regions I and II, which correspond to the case $p \in \mathrm{D}_1$, and we show that their images through γ are included in $\Delta^{\mathrm{N}}_{d_1+2c\delta} \cup \Lambda^{\mathrm{W}}_{d_1+2c\delta}$:

- p lies in Region I, *i.e.* $p \in \Delta^{\mathrm{N}}_{d_1} \cap \Lambda^{\mathrm{N}}_{c\delta}$. Then, we have $p \in Q^{\mathrm{NE}}_{c\delta}$, and (i) implies that $\|p - \gamma(p)\|_\infty \leq c\delta$. Therefore, $\gamma(p) \in \Delta^{\mathrm{N}}_{d_1+2c\delta}$.
- p lies in Region II, *i.e.* $p \in \Delta^{\mathrm{N}}_{d_1} \cap \Lambda^{\mathrm{S}}_{c\delta}$. Then, a quick computation (see Figure 16(b)) shows that p lies in $\Lambda^{\mathrm{W}}_{d_1+c\delta}$. If $\gamma(p)$ were located in $\Lambda^{\mathrm{E}}_{d_1+2c\delta}$, then (iv) would imply that $p = \gamma^{-1}(\gamma(p)) \in \Lambda^{\mathrm{E}}_{d_1+c\delta}$, thereby raising a contradiction. Therefore, $\gamma(p) \in \Lambda^{\mathrm{W}}_{d_1+2c\delta}$.

Thus, $\gamma(\mathrm{D}_1) \subseteq \Delta^{\mathrm{N}}_{d_1+2c\delta} \cup \Lambda^{\mathrm{W}}_{d_1+2c\delta}$. We now proceed with Regions III, IV, V, which correspond to the case $p \in \mathrm{D}_2$, and we show that their images through γ do not intersect $\Delta^{\mathrm{N}}_{d_1+2c\delta} \cup \Lambda^{\mathrm{W}}_{d_1+2c\delta}$:

- p lies in Region III, *i.e.* $p \in \Delta^{\mathrm{S}}_{d_2} \cap \Lambda^{\mathrm{N}}_{c\delta} = \Delta^{\mathrm{S}}_{d_2} \cap \Lambda^{\mathrm{N}}_{c\delta} \cap \Lambda^{\mathrm{E}}_{d_2+c\delta}$. Then, we have $p \in Q^{\mathrm{NE}}_{c\delta}$ and therefore $\|\gamma(p) - p\|_\infty \leq c\delta$, by (i), which implies that $\gamma(p) \in \Delta^{\mathrm{S}}_{d_2-2c\delta} \cap \Lambda^{\mathrm{E}}_{d_2}$, which is disjoint from $\Delta^{\mathrm{N}}_{d_1+2c\delta} \cup \Lambda^{\mathrm{W}}_{d_1+2c\delta}$ since by hypothesis we have $d_2 > d_1 + 4c\delta$.

- p lies in Region IV, *i.e.* $p \in \Lambda^{\mathrm{S}}_{c\delta} \cap \Lambda^{E}_{d_2+c\delta}$. Then, (iii) implies that $\gamma(p) \in \Lambda^{\mathrm{E}}_{d_2}$. In addition, we have $\gamma(p) \in \Lambda^{\mathrm{S}}_{2c\delta}$ since otherwise $\gamma(p)$ would belong to $Q^{\mathrm{NE}}_{c\delta}$ and by (ii) $p = \gamma^{-1}(\gamma(p))$ would belong to $\Lambda^{\mathrm{N}}_{c\delta}$, a contradiction. Thus, we have $\gamma(p) \in \Lambda^{\mathrm{S}}_{2c\delta} \cap \Lambda^{\mathrm{E}}_{d_2}$, which is disjoint from $\Delta^{\mathrm{N}}_{d_1+2c\delta} \cup \Lambda^{\mathrm{W}}_{d_1+2c\delta}$ since by hypothesis we have $d_2 > d_1 + 4c\delta$.
- p lies in Region V, *i.e.* $p \in \Delta^{\mathrm{S}}_{d_2} \cap \Lambda^{\mathrm{E}}_{d_2} \cap \Lambda^{\mathrm{W}}_{d_2+c\delta}$. Then, p belongs to $Q^{\mathrm{SE}}_{c\delta}$, therefore (iii) implies that $\gamma(p) \in \Lambda^{\mathrm{E}}_{d_2-c\delta}$. In addition, $\gamma(p)$ must lie in $\Lambda^{\mathrm{S}}_{2c\delta}$ or we have a contradiction by (ii) as in the previous case. Hence, $\gamma(p) \in \Lambda^{\mathrm{E}}_{d_2-c\delta} \cap \Lambda^{\mathrm{S}}_{2c\delta}$, which is disjoint from $\Delta^{\mathrm{N}}_{d_1+2c\delta} \cup \Lambda^{\mathrm{W}}_{d_1+2c\delta}$ since by hypothesis we have $d_2 > d_1 + 5c\delta$.

Thus, the persistence diagram $\mathrm{D}_0\mathcal{R}^f_\delta(P)$ is partitioned into two disjoint subsets: $\mathrm{D}^{\mathcal{R}}_1$ and $\mathrm{D}^{\mathcal{R}}_2$, which are the respective images of D_1 and D_2 through γ, and which lie respectively in the disjoint regions $\gamma(\mathrm{I} \cup \mathrm{II})$ and $\gamma(\mathrm{III} \cup \mathrm{IV} \cup \mathrm{V})$, as depicted in Figure 16(b). Then, for any choice of parameter τ within the range $(d_1 + 2c\delta,\ d_2 - 3c\delta)$, the subset $\mathrm{D}^{\mathcal{R}}_2$ (as well as D_2) is located in the region $\Delta^{\mathrm{S}}_\tau \cap \Lambda^{\mathrm{E}}_\tau$, whereas $\mathrm{D}^{\mathcal{R}}_1$ (as well as D_1) is located in its complement $\Delta^{\mathrm{N}}_\tau \cup \Lambda^{\mathrm{W}}_\tau$. This implies that the algorithm discards $\mathrm{D}^{\mathcal{R}}_1$ and keeps only $\mathrm{D}^{\mathcal{R}}_2$, which has same (finite) total multiplicity as D_2 since both sets contain no point of the diagonal Δ and are in multi-bijection. This concludes the proof of the theorem. □

10. APPROXIMATING THE BASINS OF ATTRACTION OF THE PROMINENT PEAKS OF f

The next natural question is whether the clusters output by the algorithm are faithful approximations to the actual basins of attraction of the underlying probability density function f. Using the terminology of Section 2, given a parameter $\tau \geq 0$ and a peak m_p of f of prominence at least τ, we call basin of attraction of m_p of parameter τ, noted $B_\tau(m_p)$, the union of the ascending regions of all the peaks mapped to m_p through the iterated root map r^*_τ, as per Eq. (1). Recall that the root map r takes each peak m_q of f and maps it to the higher peak $r(m_q)$ such that the connected component generated by m_q in the superlevel-sets filtration of f gets merged by persistence into the component generated by $r(m_q)$. The iterated root map r^*_τ iterates this process until some peak of prominence at least τ is reached. Given such a peak m_p, we call $\alpha_\tau(m_p)$ the time at which the connected component generated by m_p first gets connected to the one generated by another peak of prominence at least τ. Assuming $\mathrm{D}_0 f$ to be (d_1, d_2)-separated and τ to lie within the range $[d_1, d_2]$, we have the following inequalities:

$$\forall m_p \text{ s.t. } p_x - p_y \geq \tau,\ p_x - d_2 \geq \alpha_\tau(m_p) \geq p_y. \tag{6}$$

The first inequality follows from the fact that for any peak $m_q \neq m_p$ of prominence at least τ, $C(m_p, \alpha)$ and $C(m_q, \alpha)$ cannot get connected with each other above time $\alpha = p_x - d_2$, because otherwise the prominence of the younger connected component would be less than d_2 and therefore less than τ since $\mathrm{D}_0 f$ is (d_1, d_2)-separated. The second inequality follows from the fact that, at time $\alpha = p_y$, $C(m_p, \alpha)$ gets connected to another connected component, of prominence higher than $p_x - p_y \geq \tau$, which means that time $\alpha_\tau(m_p)$ has been reached.

As reported in Section 4, guaranteeing that the entire basins of attraction of the prominent peaks of f are approximated by the output of the algorithm is hopeless. However, Theorem 10.1 below gives a partial approximation guarantee (where we abuse notations by writing $B_\tau(p)$ for $B_\tau(m_p)$ and $\alpha_\tau(p)$ for $\alpha_\tau(m_p)$):

THEOREM 10.1. *Let $\mathbb{X}$ be a Riemannian manifold with positive convexity radius, and let $f : \mathbb{X} \to \mathbb{R}$ be a c-Lipschitz probability density function. If $\mathrm{D}_0 f$ is (d_1, d_2)-separated, with $d_2 > d_1 \geq 0$, then for any positive parameter $\delta < \min\{\varrho(\mathbb{X}),\ \frac{d_2 - d_1}{5c}\}$ and any threshold $\tau \in (d_1 + 2c\delta,\ d_2 - 3c\delta)$, on any input P of n sample points drawn according to f in an i.i.d. fashion the following is true with probability at least*

$1-\mathcal{N}_{\delta/8}(F^{c\delta})e^{-n\frac{3}{4}c\delta\mathcal{V}_{\delta/8}(F^{c\delta})}$*: for each point* $p\in \mathrm{D}_2$ *there is a cluster* $B^{\mathcal{R}}_\tau(p)$ *output by the algorithm such that* $B^{\mathcal{R}}_\tau(p)\cap F^\alpha = B_\tau(p)\cap L\cap F^\alpha$ *at all times* $\alpha\in(\alpha_\tau(p)+d_1+\frac{5}{2}c\delta,\ p_x]$.

In plain words, the conclusion of the theorem means that, within the superlevel-set F^α, the cluster $B^{\mathcal{R}}_\tau(p)$ is the *trace* of the basin of attraction $B_\tau(p)$ over the point cloud P. This holds from the time p_x at which the basin $B_\tau(p)$ appears in the superlevel-sets filtration of f, almost until the time $\alpha_\tau(p)$ at which $B_\tau(p)$ ceases to be disconnected from the other basins of attraction of parameter τ in the filtration. In view of Eq. (6), the duration of this phase is at least $d_2-d_1-\frac{5}{2}c\delta>0$, which as in Theorem 9.2 can be interpreted as a signal-to-noise ratio condition. As explained in Section 4 and illustrated in Figures 4 and 5, below time $\alpha_\tau(p)$ it is not possible to guarantee the approximation of the basin of attraction $B_\tau(p)$ on all instances.

The rest of Section 10 is devoted to the proof of Theorem 10.1. A noticeable feature of our proof is to not depend on a particular choice of pseudo-gradient edges within the Rips graph $R_\delta(P)$ during the mode-seeking phase of the algorithm (see Section 2). Indeed, it holds as long as the following conditions are met:

- every vertex is the origin of one pseudo-gradient edge,
- every pseudo-gradient edge connects its origin to a neighbor with a higher function value.

This feature is an indicator of the stability of our clustering technique, and it opens the door to various strategies for selecting the pseudo-gradient edges in the graph.

PROOF OF THEOREM 10.1. According to Theorem 7.2, the point cloud P forms a $\frac{\delta}{4}$-sample of $F^{c\delta}$ with probability at least $1-\mathcal{N}_{\delta/8}(F^{c\delta})e^{-n\frac{3}{4}c\delta\mathcal{V}_{\delta/8}(F^{c\delta})}$. We assume from now on that P is indeed a $\frac{\delta}{4}$-sample of $F^{c\delta}$.

The equality $B^{\mathcal{R}}_\tau(p)\cap F^\alpha = B_\tau(p)\cap L\cap F^\alpha$ will be proved by mutual inclusion: $B^{\mathcal{R}}_\tau(p)\cap F^\alpha\subseteq B_\tau(p)\cap L\cap F^\alpha$ (Lemma 10.6) and $B^{\mathcal{R}}_\tau(p)\cap F^\alpha\supseteq B_\tau(p)\cap L\cap F^\alpha$ (Lemma 10.7). We begin with a series of easy technical results (Lemmas 10.2 through 10.5) that will be key to proving the theorem:

LEMMA 10.2. *For any* $p,q\in\mathrm{D}_2$ *and any* $\alpha,\alpha'\in\mathbb{R}$*, if* $p\neq q$ *then*

$$\forall x\in B_\tau(p)\cap F^\alpha,\ \forall y\in B_\tau(q)\cap F^{\alpha'},\ \mathrm{d}_{\mathbb{X}}(x,y)\geq\frac{\max\{\alpha-\alpha_m,\ 0\}+\max\{\alpha'-\alpha_m,\ 0\}}{c},$$

where $\alpha_m=\min\{\alpha_\tau(p),\ \alpha_\tau(q)\}$.

PROOF. If $\alpha>f(m_p)$ or $\alpha'>f(m_q)$, then $B_\tau(p)\cap F^\alpha=\emptyset$ or $B_\tau(q)\cap F^{\alpha'}=\emptyset$ and the conclusion holds trivially. If $B_\tau(p)\cap F^\alpha\neq\emptyset$ and $B_\tau(q)\cap F^{\alpha'}\neq\emptyset$, then take $x\in B_\tau(p)\cap F^\alpha$, $y\in B_\tau(q)\cap F^\alpha$, and consider a shortest path[11] $[x,y]$ between x and y in $\mathbb{X}$. Let z be a point of $[x,y]$ where the value of f is minimal. Since $B_\tau(p)$ or $B_\tau(q)$ cannot get connected to any other basin of attraction of paramater τ above time α_m, we have $f(z)\leq\alpha_m$. We deduce that $\mathrm{d}_{\mathbb{X}}(x,z)\geq\frac{\alpha-\alpha_m}{c}$ and $\mathrm{d}_{\mathbb{X}}(y,z)\geq\frac{\alpha'-\alpha_m}{c}$, since f is a c-Lipschitz function. Note that these lower bounds are negative when $\alpha,\alpha'<\alpha_m$. Since z is on a shortest path between x and y, we conclude that

$$\mathrm{d}_{\mathbb{X}}(x,y)=\mathrm{d}_{\mathbb{X}}(x,z)+\mathrm{d}_{\mathbb{X}}(z,y)\geq\frac{\max\{\alpha-\alpha_m,\ 0\}+\max\{\alpha'-\alpha_m,\ 0\}}{c}.$$

□

[11] Since we did not make any assumption regarding the existence of shortest paths between arbitrary points on the manifold $\mathbb{X}$, it may happen that no shortest path exists between x and y. However, we can always consider paths $[x,y]$ of length at most $\mathrm{d}_{\mathbb{X}}(x,y)+\zeta$, for arbitrarily small values $\zeta>0$.

For any $p \in \mathrm{D}_2$, we let

$$v_p = \mathrm{argmax}_{v \in B_\tau(p) \cap P} f(v).$$

This point is well-defined because $B_\tau(p) \cap P$ is not empty. Indeed, by Lemma 10.2 and Eq. (6) we have $\min\{\mathrm{d}_{\mathbb{X}}(m_p, y) \mid y \in \mathbb{X} \setminus B_\tau(p)\} \geq \frac{f(m_p) - \alpha_\tau(p)}{c} \geq \frac{d_2}{c} > \frac{\tau}{c} \geq \delta \geq \min\{\mathrm{d}_{\mathbb{X}}(m_p, x) \mid x \in P\}$, which implies that $B_\tau(p)$ contains at least one point of P.

LEMMA 10.3. *For any $p \in \mathrm{D}_2$, we have $f(m_p) \geq f(v_p) \geq f(m_p) - c\frac{\delta}{4}$.*

PROOF. The first inequality follows from the definition of m_p as the argmax of f over $B_\tau(p)$, which contains v_p. To prove the second inequality, we use our assumption that P forms a $\frac{\delta}{4}$-sample of $F^{c\delta}$ and therefore of F^{d_2} since $d_2 \geq c\delta$ by hypothesis. Then, because p belongs to $\mathrm{D}_2 \subset \Lambda^{\mathrm{E}}_{d_2}$, we have $m_p \in F^{d_2}$ and therefore there is a point $v \in P$ such that $\mathrm{d}_{\mathbb{X}}(v, m_p) \leq \delta/4$. Since f is c-Lipschitz, we have $f(v) \geq f(m_p) - c\delta/4$. To complete the proof, we only need to show that v actually lies in the basin $B_\tau(p)$, which will imply that $f(v_p) \geq f(v) \geq f(m_p) - c\delta/4$. By Lemma 10.2, the geodesic distance of m_p to $\mathbb{X} \setminus B_\tau(p)$ is at least $\frac{f(m_p) - \alpha_\tau(p)}{c} = \frac{p_x - \alpha_\tau(p)}{c}$, which by Eq. (6) is at least $\frac{d_2}{c}$, which by hypothesis is greater than 5δ. It follows then from the triangle inequality that the geodesic distance of v to $\mathbb{X} \setminus B_\tau(p)$ is strictly positive, which means that $v \in B_\tau(p)$. □

It follows from the previous results that v_p is a peak of f in the Rips graph $R_\delta(P)$. Indeed, Lemma 10.3 guarantees that $f(v_p) \geq f(m_p) - c\delta/4 = p_x - c\delta/4$, which by Eq. (6) is at least $\alpha_\tau(p) + d_2 - c\delta/4$. Therefore, Lemma 10.2 ensures that the geodesic distance of v_p to $\mathbb{X} \setminus B_\tau(p)$ is at least $\frac{d_2}{c} - \frac{\delta}{4}$, which by hypothesis is greater than δ. This implies that every neighbor v of v_p in the Rips graph $R_\delta(P)$ lies in the basin $B_\tau(p)$, and by definition of v_p that $f(v) \leq f(v_p)$. Thus, v_p is a local maximum in $R_\delta(P)$. As a result, at time $f(v_p)$ a new connected component $C^{\mathcal{R}}(v_p, f(v_p))$ appears in the upper-star Rips filtration $\mathcal{R}^f_\delta(P)$, or more precisely in the subgraph $R_\delta(P \cap F^\alpha)$. In homological terms, this connected component is *generated* by the peak v_p. Its lifespan is encoded as a point $p^{\mathcal{R}}$ in the persistence diagram $\mathrm{D}_0\mathcal{R}^f_\delta(P)$. Note that this point may or may not be identical to the point $\gamma(p)$ associated with p by the multi-bijection introduced in the proof of Theorem 9.2. Defining regions $\mathrm{D}^{\mathcal{R}}_1$ and $\mathrm{D}^{\mathcal{R}}_2$ as in the proof of Theorem 9.2, we have:

LEMMA 10.4. *For all $p \in \mathrm{D}_2$, $p^{\mathcal{R}} \in \mathrm{D}^{\mathcal{R}}_2$.*

PROOF. At any time $\alpha \in (\alpha_\tau(p) + c\delta/2, f(v_p)]$, Lemma 10.2 guarantees that every point of $P \cap F^\alpha \cap B_\tau(p)$ (including v_p itself) is disconnected from every point of $P \cap F^\alpha \setminus B_\tau(p)$ in the subgraph $R_\delta(P \cap F^\alpha)$, therefore the connected component $C^{\mathcal{R}}(v_p, \alpha)$ is included in $B_\tau(p)$. This implies that v_p remains the argmax of f over $C^{\mathcal{R}}(v_p, \alpha)$, and therefore that $C^{\mathcal{R}}(v_p, \alpha)$ still exists as an independent connected component in the subgraph $R_\delta(P \cap F^\alpha)$. It follows that $p^{\mathcal{R}}_y \leq \alpha_\tau(p) + c\delta/2$, which in turn implies that $p^{\mathcal{R}}_x - p^{\mathcal{R}}_y \geq f(v_p) - \alpha_\tau(p) - c\delta/2$. By Lemma 10.3, this quantity is at least $f(m_p) - \alpha_\tau(p) - 3c\delta/4 = p_x - \alpha_\tau(p) - 3c\delta/4$, which by Eq. (6) is at least $d_2 - 3c\delta/4$. Thus, $p^{\mathcal{R}}$ lies in $\Delta^{\mathrm{S}}_{d_2 - 3c\delta/4} \subset \Delta^{\mathrm{S}}_{d_2 - 3c\delta}$. In addition, we have $p^{\mathcal{R}}_x = f(v_p) \geq f(m_p) - \frac{c\delta}{4} = p_x - \frac{c\delta}{4}$, which is at most $d_2 - c\frac{\delta}{4}$ since by hypothesis $p \in \mathrm{D}_2 \subset \Lambda^{\mathrm{E}}_{d_2}$. Hence, $p^{\mathcal{R}}$ also lies in $\Lambda^{\mathrm{E}}_{d_2 - c\delta/4} \subset \Lambda^{\mathrm{E}}_{d_2 - 3c\delta}$, which proves that $p^{\mathcal{R}} \in \mathrm{D}^{\mathcal{R}}_2$ since $d_2 > d_1 + 5c\delta$. □

According to Lemma 10.4, $p \mapsto p^{\mathcal{R}}$ is a map $\mathrm{D}_2 \to \mathrm{D}^{\mathcal{R}}_2$. This map is clearly injective, since by definition $p^{\mathcal{R}}$ corresponds to the connected component of $\mathcal{R}^f_\delta(P)$ generated by the peak v_p which belongs to the basin $B_\tau(p)$ and to none other. In fact, the map is bijective since by Theorem 9.2 the cardinalities of D_2 and $\mathrm{D}_2^{\mathcal{R}}$ are the same. Another important consequence of Lemma 10.4 is that v_p is in fact the generator of a whole cluster output by the algorithm. We call $B_\tau^{\mathcal{R}}(p)$ this cluster.

Given a point $x \in P$, we denote by $r(x)$ the root of the tree to which x is attached during the mode-seeking phase of the algorithm (see Section 2). For each merge of an entry e into another entry e' performed in the union-find data structure during the merging phase of the algorithm, we call e' the *root* of e, noted $e' = r(e)$. We can then iterate the root map, starting at x, until we reach the root of the cluster containing x in the output of the algorithm. This root is denoted $r_\tau^*(x)$, by analogy with the continuous setting. By construction, $r_\tau^*(x)$ is the only peak of f (within the Rips graph $R_\delta(P)$) of prominence at least τ in its cluster. Therefore, in the persistence diagram $\mathrm{D}_0\mathcal{R}_\delta^f(P)$, $r_\tau^*(x)$ corresponds to some point $q \in \mathrm{D}_2^{\mathcal{R}}$. Let $p \in \mathrm{D}_2$ be such that $p^{\mathcal{R}} = q$. Such a point exists since the map $p \mapsto p^{\mathcal{R}}$ is a bijection $\mathrm{D}_2 \to \mathrm{D}_2^{\mathcal{R}}$. The cluster containing x in the output of the algorithm is then $B_\tau^{\mathcal{R}}(p)$, and its root is $r_\tau^*(x) = v_p$.

LEMMA 10.5. *$\forall x \in P$, $\forall \alpha \leq f(x) - d_1 - 2c\delta$, $C^{\mathcal{R}}(x, \alpha) = C^{\mathcal{R}}(r_\tau^*(x), \alpha)$.*

PROOF. By definition of the root $r(x)$, there is a path from x to $r(x)$ in the Rips graph $R_\delta(P)$ such that f increases along this path. This means that x and $r(x)$ belong to the same connected component of the subgraph $R_\delta(P \cap F^{f(x)})$. Since $\alpha \leq f(x)$, we deduce that $C^{\mathcal{R}}(x, \alpha) = C^{\mathcal{R}}(r(x), \alpha)$.

For convenience, we let $x_0 = r(x)$, $x_1 = r(x_0)$, $\cdots$, $x_{l-1} = r(x_{l-2})$, and $x_l = r(x_{l-1}) = r_\tau^*(x)$. We have $f(x_l) \geq f(x_{l-1}) \geq \cdots \geq f(x_0) \geq f(x)$. By construction, the cluster output by the algorithm that contains the x_i does not contain any peak of f of prominence τ or more beside x_l. This means that, for any $i < l$, the peak x_i is less than τ-prominent and therefore corresponds to some point of $\mathrm{D}_1^{\mathcal{R}}$ in the diagram $\mathrm{D}_0\mathcal{R}_\delta^f(P)$. It follows in particular that the prominence of x_i is less than $d_1 + 2c\delta$, which means that $C^{\mathcal{R}}(x_i, f(x_i) - d_1 - 2c\delta) = C^{\mathcal{R}}(x_{i+1}, f(x_i) - d_1 - 2c\delta)$. Now, we have $f(x_i) - d_1 - 2c\delta \geq f(x) - d_1 - 2c\delta \geq \alpha$, which implies that $C^{\mathcal{R}}(x_i, \alpha) = C^{\mathcal{R}}(x_{i+1}, \alpha)$. Since this is true for all $i < l$, we conclude that $C^{\mathcal{R}}(x_0, \alpha) = C^{\mathcal{R}}(x_1, \alpha) = \cdots = C^{\mathcal{R}}(x_l, \alpha) = C^{\mathcal{R}}(r_\tau^*(x), \alpha)$. This fact, combined with the observation that $C^{\mathcal{R}}(x, \alpha) = C^{\mathcal{R}}(r(x), \alpha) = C^{\mathcal{R}}(x_0, \alpha)$, concludes the proof of the lemma. □

We are now ready to prove our first inclusion:

LEMMA 10.6. *For all $p \in \mathrm{D}_2$ and all $\alpha > \alpha_\tau(p) + d_1 + \frac{5}{2}c\delta$, $B_\tau^{\mathcal{R}}(p) \cap F^\alpha \subseteq B_\tau(p) \cap P \cap F^\alpha$.*

PROOF. For any $\alpha > f(v_p)$, $B_\tau^{\mathcal{R}}(p) \cap F^\alpha$ is empty and so the inclusion holds trivially. Assume from now on that $\alpha_\tau(p) + d_1 + \frac{5}{2}c\delta < \alpha \leq f(v_p)$, and consider a point $x \in B_\tau^{\mathcal{R}}(p) \cap F^\alpha$. Since $f(x) \geq \alpha$, Lemma 10.5 guarantees that $C^{\mathcal{R}}(x, \alpha - d_1 - 2c\delta) = C^{\mathcal{R}}(r_\tau^*(x), \alpha - d_1 - 2c\delta)$. In other words, x and $r_\tau^*(x)$ belong to the same connected component of the subgraph $R_\delta(P \cap F^{\alpha - d_1 - 2c\delta})$. Since by hypothesis $\alpha - d_1 - 2c\delta$ is greater than $\alpha_\tau(p) + c\delta/2$, Lemma 10.2 ensures that every point of $P \cap F^{\alpha - d_1 - 2c\delta} \cap B_\tau(p)$, including $v_p = r_\tau^*(x)$ itself, is disconnected from every point of $P \cap F^{\alpha - d_1 - 2c\delta} \setminus B_\tau(p)$ in the subgraph $R_\delta(P \cap F^{\alpha - d_1 - 2c\delta})$. This implies that x belongs to $B_\tau(p)$. □

We now proceed with the inclusion in the other direction:

LEMMA 10.7. *For all $p \in \mathrm{D}_2$ and all $\alpha > \alpha_\tau(p) + d_1 + \frac{5}{2}c\delta$, $B_\tau(p) \cap P \cap F^\alpha \subseteq B_\tau^{\mathcal{R}}(p) \cap F^\alpha$.*

PROOF. Since by definition v_p is the argmax of f over $B_\tau(p) \cap P$, for all $\alpha > f(v_p)$ the set $B_\tau(p) \cap P \cap F^\alpha$ is empty and so the inclusion holds trivially. Assume from now on that $\alpha_\tau(p) + d_1 + \frac{5}{2}c\delta < \alpha \leq f(v_p)$, and let $x \in B_\tau(p) \cap P \cap F^\alpha$. Let $q \in \mathrm{D}_2$ be such that $v_q = r_\tau^*(x)$. Since $f(x) \geq \alpha$, Lemma 10.5 guarantees that $C^{\mathcal{R}}(x, \alpha - d_1 - 2c\delta) =$

$C^{\mathcal{R}}(v_q, \alpha - d_1 - 2c\delta)$. Now, since $\alpha - d_1 - 2c\delta > \alpha_\tau(p) + c\delta/2$, Lemma 10.2 ensures that every point of $P \cap F^{\alpha-d_1-2c\delta} \cap B_\tau(p)$, including x itself, is disconnected from every point of $P \cap F^{\alpha-d_1-2c\delta} \setminus B_\tau(p)$ in the subgraph $R_\delta(P \cap F^{\alpha-d_1-2c\delta})$. This implies that v_q belongs to $B_\tau(p)$, and therefore that $v_q = v_p$. Hence, x belongs to $B_\tau^{\mathcal{R}}(p)$. □

The conclusion of Theorem 10.1 follows from the mutual inclusions stated in Lemmas 10.6 and 10.7. □

11. ROBUSTNESS OF THE APPROACH

In the previous sections we assumed the input function $\tilde{f}$ to be the true density function f. In many practical scenarios however, density values are not supplied and must be estimated from the data set P. In this section, we show that the output of the algorithm is robust to small perturbations of these values, thus making our approach practical. More precisely, we assume the density estimator $\tilde{f}$ to approximate the true density function f over the point cloud P within an additive error η:

$$\sup_{v \in P} |\tilde{f}(v) - f(v)| < \eta. \tag{7}$$

Then, without any modification to the algorithm, the persistence diagram given as output still approximates the persistence diagram of f, with a slighty degraded approximation bound:

THEOREM 11.1. *Under the hypotheses of Theorem 8.2, and assuming that Eq. (7) is satisfied by $\tilde{f}$, for any positive $\delta < \varrho(\mathbb{X})$ and any $\alpha > 0$ the following holds with probability at least $\left(1 - \mathcal{N}_{\delta/8}(F^\alpha)\, e^{-|P|(\alpha - c\delta/4)\mathcal{V}_{\delta/8}(F^\alpha)}\right)$: there is a multi-bijection γ between the persistence diagrams of f and of the upper-star filtration $\mathcal{R}_\delta^{\tilde{f}}(P)$ induced by $\tilde{f}$ on the Rips graph $R_\delta(P)$, such that:*

(i) $\forall p \in \mathrm{D}_0 f \cap Q^{\mathrm{NE}}_{\alpha+\eta}$, $\|p - \gamma(p)\|_\infty \leq c\delta + \eta$.

(ii) $\forall q \in \mathrm{D}_0 \mathcal{R}_\delta^{\tilde{f}}(P) \cap Q^{\mathrm{NE}}_{\alpha+\eta}$, $\|\gamma^{-1}(q) - q\|_\infty \leq c\delta + \eta$.

(iii) $\forall p \in \mathrm{D}_0 f \cap Q^{\mathrm{SE}}_{\alpha+\eta}$, $|p_x - \gamma(p)_x| \leq c\delta + \eta$.

(iv) $\forall q \in \mathrm{D}_0 \mathcal{R}_\delta^{\tilde{f}}(P) \cap Q^{\mathrm{SE}}_{\alpha+\eta}$, $|\gamma^{-1}(q)_x - q_x| \leq c\delta + \eta$.

PROOF. Eq. (7) implies that the upper-star filtrations $\mathcal{R}_\delta^{f}(P)$ and $\mathcal{R}_\delta^{\tilde{f}}(P)$ are interleaved with respect to inclusion:

$$\forall \alpha \in \mathbb{R},\ R_\delta(P \cap F^\alpha) \subseteq R_\delta(P \cap \tilde{F}^{\alpha-\eta}) \subseteq R_\delta(P \cap F^{\alpha - 2\eta}).$$

Therefore, their induced persistence modules at 0-dimensional homology level are (strongly) η-interleaved. As a consequence, the bottleneck distance between their persistence diagrams is bounded by η. In addition, it follows from Theorem 7.2 that P forms a $\frac{\delta}{4}$-sample of the superlevel-set F^α with probability $1 - \mathcal{N}_{\delta/8}(F^\alpha)\, e^{-|P|(\alpha - c\delta/4)\mathcal{V}_{\delta/8}(F^\alpha)}$. So, with the same probability P statisfies the assumptions of Theorem 8.2, which implies that $\mathrm{D}_0 f$ and $\mathrm{D}_0 \mathcal{R}_\delta^{\tilde{f}}(P)$ satisfy assertions (i) through (iv) of Theorem 8.2. The result follows. □

The above theorem replaces Theorem 8.2, and the rest of the analysis unfolds in the same way as before.

Density estimation is an extensive research area and many methods to estimate the values of f from the data set P can be used in practice (see *e.g.* [Devroye and

Lugosi 2001]). Identifying the families of density estimators $\tilde{f}$ that satisfy Eq. (7) in full generality is beyond the scope of this paper. Nevertheless, in some cases constructing such an estimator is not too difficult, which we will now illustrate in the Euclidean setting with a simple kernel-based estimator.

Suppose the Riemannian manifold $\mathbb{X}$ is the Euclidean space $\mathbb{R}^m$. Let P be a finite set of data points sampled according to some probability density function $f : \mathbb{R}^m \to \mathbb{R}$. We assume that the coordinates of the points of P are given, so that their pairwise Euclidean distances can be computed exactly. The density f can then be approximated using the following *ball estimator*:

$$\tilde{f}_r(x) = \frac{1}{\mathcal{V}_r} \frac{|P \cap \mathcal{B}(x,r)|}{|P|}, \tag{8}$$

where $\mathcal{B}(x,r)$ is a concise replacement for $\mathcal{B}_{\mathbb{R}^m}(x,r)$, the Euclidean m-ball of center x and radius r, and where $\mathcal{V}_r$ is a concise replacement for $\mathcal{V}_r(\mathbb{R}^m)$, the volume of any Euclidean m-ball of radius r.

LEMMA 11.2. *If f is c-Lipschitz, then for any value of parameter r and any $\zeta \geq 0$, we have $\sup_{v\in P} |\tilde{f}_r(v) - f(v)| \leq cr + \zeta$ with probability at least $1 - |P|e^{-2|P|(\zeta\mathcal{V}_r)^2}$.*

PROOF. Let μ be the measure associated with the density function f. Given a point $v \in P$, we know from the path-connectivity of $\mathcal{B}(v,r)$ and from the Intermediate Value Theorem that there is a point $x \in \mathcal{B}(v,r)$ such that $f(x)$ equals the average value of f inside the ball, that is:

$$f(x) = \frac{\mu(\mathcal{B}(v,r))}{\mathcal{V}_r}. \tag{9}$$

Since f is c-Lipschitz, we have $|f(v) - f(x)| \leq cr$. Combined with Eq. (9), this gives:

$$\left| f(v) - \frac{\mu(\mathcal{B}(v,r))}{\mathcal{V}_r} \right| \leq cr. \tag{10}$$

In addition, the Bounded Differences Inequality tells us that for any $\xi > 0$, we have:

$$\left| \frac{|P \cap \mathcal{B}(v,r)|}{|P|} - \mu(\mathcal{B}(v,r)) \right| \leq \xi \tag{11}$$

with probability at least $1 - e^{-2|P|\xi^2}$. Letting $\xi = \zeta\mathcal{V}_r$ in the above expression and combining it with Eqs. (8) and (10), we obtain:

$$\left| \tilde{f}_r(v) - f(v) \right| \leq cr + \zeta \tag{12}$$

with probability at least $1 - e^{-2|P|(\zeta\mathcal{V}_r)^2}$. The lemma follows then from the application of the union bound on the set P. □

Notice that the ball estimator (8) strongly relies on the property that the volume of a Euclidean m-ball of radius r in $\mathbb{R}^m$ does not depend on the location of its center. This is not the case in general Riemannian manifolds. To overcome this issue it is possible to consider kernels of the following form:

$$\tilde{g}(x) = \frac{\sum_{p\in P} K(d_{\mathbb{X}}(p,x))}{|P|}, \tag{13}$$

where $K : \mathbb{R} \to \mathbb{R}$ is a non negative function such that $\int_{-\infty}^{\infty} K(u)du = 1$ and $K(u) = K(-u)$. Then, under some conditions, $\tilde{g}$ can be seen as an estimator of the convolution of f with $K \circ d_{\mathbb{X}}$, assuming that P has been sampled according to f. We refer the reader to [Devroye and Gyorfi 1985; Tsybakov 2008] for further details on kernel-based density estimation.

Perturbing distances. Slightly increasing the Rips parameter value used in the algorithm makes the output also robust to small perturbations of the geodesic distances between the data points. In the analysis, this very mild change to the algorithm allows one to combine a result from [Chazal et al. 2011] (namely Theorem 4) with our Theorem 8.2, making the latter resilient to some degree of fuzziness in the values of the geodesic distances. The formal statements and proofs are technical and do not bear any conceptual novelty, furthermore a very similar analysis was already performed in [Chazal et al. 2011, §3.3], therefore we refer the reader to that paper for the details.

12. CONCLUSION

We have introduced a new clustering algorithm that combines a classical mode-seeking step with a novel persistence-based cluster merging step. It is straightforward to implement and provably robust to noise. Rather than rely on heuristics, it returns structural information about the modes of the density function in the form of a persistence diagram, which allows the user to see the relationship between the choice of parameter values and the number of obtained clusters. In many cases this diagram provides insights into the correct number of clusters, which can be automatically inferred by further processing. Our method can work with any density estimator and any metric, including Euclidean, geodesic, and diffusion distances. The point is that the persistence diagram only displays the information that is present in the density function and underlying space (known through the input distance matrix).

Our theoretical developments provide an understanding of when the data has a clear number of clusters, and which parts of the clusters are stable under small perturbations of the input. This opens up the possibility of doing soft-clustering, where each point is assigned to a cluster with some probability. Finally we note that, because we use a topological framework, additional features can be extracted from the data through higher-dimensional persistence diagrams [Chazal et al. 2011], such as the circular structure of the rings in the synthetic data set of Figure 10(a) — although it is not yet clear how this type of information can be exploited.

Acknowledgements

The authors wish to thank the anonymous referees for their careful reading of the paper and their insightful comments. This work was carried out within the associated teams TGDA and CoMeT. It was suported by ANR grant GIGA, NSF grants FODAVA 0808515, DMS 0900700, and CCF 1011228, by a Stanford-France grant, and by a gift from Google, Inc.

REFERENCES

G. Biau, F. Chazal, D. Cohen-Steiner, L. Devroye, and C. Rodrgues. 2011. A Weighted k-Nearest Neighbor Density Estimate for Geometric Inference. *Electronic Journal of Statistics* 5 (2011), 204–237.

Gunnar Carlsson. 2009. Topology and Data. *Bull. Amer. Math. Soc.* 46, 2 (2009), 255–308.

Gunnar Carlsson, Tigran Ishkhanov, Vin Silva, and Afra Zomorodian. 2008. On the Local Behavior of Spaces of Natural Images. *Int. J. Computer Vision* 76, 1 (2008), 1–12. DOI:http://dx.doi.org/10.1007/s11263-007-0056-x

G. Carlsson, A. Zomorodian, A. Collins, and L. Guibas. 2004. Persistence Barcodes for Shapes. In *Proc. Symp. Geom. Process.* 127–138.

F. Chazal and D. Cohen-Steiner. 2007. *Geometric Inference*. to appear as a book chapter, Springer. http://www-sop.inria.fr/geometrica/team/Frederic.Chazal/

F. Chazal, D. Cohen-Steiner, L. J. Guibas, M. Glisse, and S. Y. Oudot. 2009. Proximity of Persistence Modules and their Diagrams. In *Proc. 25th ACM Sympos. Comput. Geom.* 237–246.

Frédéric Chazal, Vin de Silva, Marc Glisse, and Steve Y. Oudot. 2012. *The structure and stability of persistence modules*. Research Report arXiv:1207.3674 [math.AT]. http://arxiv.org/abs/1207.3674

F. Chazal, L. J. Guibas, S. Y. Oudot, and P. Skraba. 2011. Analysis of Scalar Fields over Point Cloud Data. *Discrete and Computational Geometry* 46, 4 (December 2011), 743–775.

Wen-Yen Chen, Yangqiu Song, Hongjie Bai, Chih-Jen Lin, and Edward Y. Chang. 2008. *PSC: Parallel Spectral Clustering*. http://www.cs.ucsb.edu/~wychen/sc.

John D. Chodera, Nina Singhal, Vijay S. Pande, Ken A. Dill, and William C. Swope. 2007. Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics. *The Journal of Chemical Physics* 126, 15 (2007), 155101.

John D. Chodera, William C. Swope, Jed W. Pitera, and Ken A. Dill. 2006. Long-Time Protein Folding Dynamics from Short-Time Molecular Dynamics Simulations. *Multiscale Modeling & Simulation* 5, 4 (2006), 1214–1226. DOI: http://dx.doi.org/10.1137/06065146X

John D. Chodera, William C. Swope, Jed W. Pitera, Chaok Seok, and Ken A. Dill. 2007. Use of the Weighted Histogram Analysis Method for the Analysis of Simulated and Parallel Tempering Simulations. *Journal of Chemical Theory and Computation* 3, 1 (2007), 26–41. DOI: http://dx.doi.org/10.1021/ct0502864

D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. 2005. Stability of Persistence Diagrams. In *Proc. 21st ACM Sympos. Comput. Geom.* 263–271.

D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. 2007. Stability of Persistence Diagrams. *Discrete Comput. Geom.* 37, 1 (2007), 103–120.

D. Comaniciu and P. Meer. 2002. Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 24, 5 (May 2002), 603–619.

T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. 2001. *Introduction to Algorithms* (2nd ed.). MIT Press, Cambridge, MA.

Luc Devroye and Laszlo Gyorfi. 1985. *Nonparametric Density Estimation: The L1 View*. Wiley Series in Probability and Statistics.

L. Devroye and G. Lugosi. 2001. *Combinatorial Methods in Density Estimation*. Springer.

H. Edelsbrunner and J. Harer. 2007. Persistent homology — a survey. *In Twenty Years After, AMS* (2007).

H. Edelsbrunner, J. Harer, and A. Zomorodian. 2001. Hierarchical Morse Complexes for Piecewise Linear 2-Manifolds. In *Proc. 17th Annu. Sympos. Comput. Geom.* 70–79.

H. Edelsbrunner, D. Letscher, and A. Zomorodian. 2002. Topological Persistence and Simplification. *Discrete Comput. Geom.* 28 (2002), 511–533.

M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. 2nd Internat. Conf. on Knowledge Discovery and Data Mining (KDD-96)*. AAAI Press, 226–231.

S. Gallot, D. Hulin, and J. Lafontaine. 2004. *Riemannian Geometry* (3 ed.). Springer.

Robert Ghrist. 2007. Barcodes: the persistent topology of data. *Amer. Math. Soc. Current Events Bulletin* 45, 1 (January 2007), 61–75.

J.A. Hartigan. 1975. *Clustering Algorithms*. Wiley.

T. Hastie, R. Tibshirani, and J. H. Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer. Second edition.

A. Hatcher. 2001. *Algebraic Topology*. Cambridge Univ. Press. http://www.math.cornell.edu/~hatcher/

Wilhelm Huisinga and Bernd Schmidt. 2005. Advances in Algorithms for Macromolecular Simulation, chapter Metastability and dominant eigenvalues of transfer operators. *Lecture Notes in Computational Science and Engineering. Springer* (2005).

A. Kolmogorov and V. Tikhomirov. 1961. ϵ-entropy and ϵ-capacity of sets of functions. *Translations of the AMS* 17 (1961), 277–364.

W. L. Koontz, P. M. Narendra, and K. Fukunaga. 1976. A Graph-Theoretic Approach to Nonparametric Cluster Analysis. *IEEE Trans. on Computers* 24 (September 1976), 936–944.

Stuart P. Lloyd. 1982. Least squares quantization in pcm. *IEEE Trans. on Information Theory* 28, 2 (1982), 129–136.

D. M. Mount and S. Arya. 2010. ANN: A Library for Approximate Nearest Neighbor Searching, version 1.1.2. (2010). http://www.cs.umd.edu/~mount/ANN/.

S. Paris and F. Durand. 2007. A Topological Approach to Hierarchical Segmentation using Mean Shift. In *Proc. IEEE conf. on Computer Vision and Pattern Recognition*.

J. Sander, M. Ester, H.-P. Kriegel, and X. Xu. 1998. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. *Data Mining and Knowledge Discovery* 2, 2 (1998), 169–194.

Y. A. Sheikh, E. Khan, and T. Kanade. 2007. Mode-seeking by Medoidshifts. In *Proc. 11th IEEE Internat. Conf. on Computer Vision (ICCV 2007)*.

D. L. Theobald. 2005. Rapid calculation of RMSDs using a quaternion-based characteristic polynomial. *Acta Crystallographica Section A: Foundations of Crystallography* 61, 4 (2005), 478–480.

Godfried T. Toussaint. 1980. The Relative Neighbourhood Graph of a Finite Planar Set. *Pattern Recognition* 12 (1980), 261–268.

Alexandre B. Tsybakov. 2008. *Introduction to Nonparametric Estimation*. Springer Publishing Company, Incorporated.

A. Vedaldi and S. Soatto. 2008. Quick Shift and Kernel Methods for Mode Seeking. In *Proc. European Conf. on Computer Vision (ECCV)*.

Ulrike von Luxburg. 2007. A Tutorial on Spectral Clustering. *Statistics and Computing* 17, 4 (2007), 395–416.

A. Zomorodian and G. Carlsson. 2005. Computing Persistent Homology. *Discrete Comput. Geom.* 33, 2 (2005), 249–274.

A. APPENDIX — PROOF OF LEMMA 8.3

Let $\mathcal{X} = \{X^\beta\}_{\beta\in\mathbb{R}}$ and $\mathcal{Y} = \{Y^\beta\}_{\beta\in\mathbb{R}}$ be two tame persistence modules that are (strongly) ε-interleaved above some given time α. Let $\{x^\beta_{\beta'} : X^{\beta'} \to X^\beta\}_{\beta'\geq\beta}$ be the family of homomorphisms associated with $\mathcal{X}$, and $\{y^\beta_{\beta'} : Y^{\beta'} \to Y^\beta\}_{\beta'\geq\beta}$ the family of homomorphisms associated with $\mathcal{Y}$. We define a new persistence module $\tilde{\mathcal{X}}$ from $\mathcal{X}$ as follows:

$$\begin{cases} \forall\beta \geq \alpha-\varepsilon,\ \tilde{X}^\beta = X^\beta \\ \forall\beta < \alpha-\varepsilon,\ \tilde{X}^\beta = 0 \end{cases} \qquad \begin{cases} \forall\beta \geq \alpha-\varepsilon,\ \forall\beta'\geq\beta,\ \tilde{x}^\beta_{\beta'} = x^\beta_{\beta'} \\ \forall\beta < \alpha-\varepsilon,\ \forall\beta'\geq\beta,\ \tilde{x}^\beta_{\beta'} = 0 \end{cases} \tag{14}$$

Clearly, $\tilde{x}^\beta_{\beta'} \circ \tilde{x}^{\beta'}_{\beta''} = x^\beta_{\beta'} \circ x^{\beta'}_{\beta''} = x^\beta_{\beta''} = \tilde{x}^\beta_{\beta''}$ when $\beta \geq \alpha-\varepsilon$, whereas $\tilde{x}^\beta_{\beta'} \circ \tilde{x}^{\beta'}_{\beta''} = 0 = \tilde{x}^\beta_{\beta''}$ when $\beta < \alpha - \varepsilon$. Thus, $\tilde{\mathcal{X}}$ is indeed a persistence module. The fact that $\tilde{x}^\beta_{\beta'} = x^\beta_{\beta'}$ whenever $\beta' \geq \beta \geq \alpha - \varepsilon$ implies that $\mathrm{D}\mathcal{X} \cap Q^{\mathrm{NE}}_{\alpha-\varepsilon} = \mathrm{D}\tilde{\mathcal{X}} \cap Q^{\mathrm{NE}}_{\alpha-\varepsilon}$, by the definition of persistence diagram[12]. Let then $\gamma_X : \mathrm{D}\mathcal{X} \to \mathrm{D}\tilde{\mathcal{X}}$ be a multi-bijection such that γ_X and γ_X^{-1} leave the points within $Q^{\mathrm{NE}}_{\alpha-\varepsilon}$ fixed. We will show that the total multiplicities of $\mathrm{D}\mathcal{X}$ and $\mathrm{D}\tilde{\mathcal{X}}$ are equal within any given vertical half-line $\{\beta'\}\times[-\infty, \beta]$ where $\beta' > \beta \geq \alpha-\varepsilon$, which will enable us to further assume that γ_X and γ_X^{-1} only move the points vertically within the lower-right quadrant $Q^{\mathrm{SE}}_{\alpha-\varepsilon}$, as illustrated in Figure 15 (right).

Using the terminology introduced in [Chazal et al. 2009], given any $\eta > 0$ we discretize $\mathcal{X}$ and $\tilde{\mathcal{X}}$ over the integer scale $\alpha - \varepsilon + \eta\mathbb{Z}$, to get respectively $\mathcal{X}_{\alpha-\varepsilon+\eta\mathbb{Z}}$ and $\tilde{\mathcal{X}}_{\alpha-\varepsilon+\eta\mathbb{Z}}$. Their persistence diagrams are then snapped onto the regular grid $(\alpha - \varepsilon + \eta\mathbb{Z}) \times (\alpha - \varepsilon + \eta\mathbb{Z})$, as per Theorem 3.7 of [Chazal et al. 2009] (the snapping directions are reversed here, since time flows from $+\infty$ to $-\infty$). For any integers $i > j \in \mathbb{Z}$, the total multiplicity of $\mathrm{D}\mathcal{X}_{\alpha-\varepsilon+\eta\mathbb{Z}}$ within the vertical half-line $\{\alpha-\varepsilon+i\eta\} \times [-\infty,\ \alpha-\varepsilon+j\eta]$ is given by the sum of the multiplicities of the points $(\alpha-\varepsilon+i\eta,\ \alpha-\varepsilon+(j-k)\eta)$ for k ranging over $\mathbb{N}\cup\{+\infty\}$:

$$\mu^{\mathrm{tot}}_{\eta,i,j}(\mathrm{D}\mathcal{X}_{\alpha-\varepsilon+\eta\mathbb{Z}}) = \mu(\alpha-\varepsilon+i\eta, -\infty) + \sum_{k\in\mathbb{N}} \mu(\alpha-\varepsilon+i\eta,\ \alpha-\varepsilon+(j-k)\eta), \tag{15}$$

where by definition[13] the multiplicity of point $(\alpha-\varepsilon+i\eta,\ \alpha-\varepsilon+(j-k)\eta)$, $k\in\mathbb{N}$, is given by:

$$\begin{aligned} \mu(\alpha-\varepsilon+i\eta,\ \alpha-\varepsilon+(j-k)\eta) \ =\ & \left(\operatorname{rank} x^{\alpha-\varepsilon+(j-k+1)\eta}_{\alpha-\varepsilon+i\eta} - \operatorname{rank} x^{\alpha-\varepsilon+(j-k+1)\eta}_{\alpha-\varepsilon+(i+1)\eta}\right) \\ & - \left(\operatorname{rank} x^{\alpha-\varepsilon+(j-k)\eta}_{\alpha-\varepsilon+i\eta} - \operatorname{rank} x^{\alpha-\varepsilon+(j-k)\eta}_{\alpha-\varepsilon+(i+1)\eta}\right). \end{aligned} \tag{16}$$

Since the persistence module $\mathcal{X}$ is tame, the ranks in Eq. (16) are finite. Notice that the sequence $\left(\operatorname{rank} x^{\alpha-\varepsilon+(j-k)\eta}_{\alpha-\varepsilon+i\eta}\right)_{k\in\mathbb{N}}$ is non-increasing, bounded from below by zero, and takes integer values. Hence, it becomes stationary after a while. The same argument holds for the sequence $\left(\operatorname{rank} x^{\alpha-\varepsilon+(j-k)\eta}_{\alpha-\varepsilon+(i+1)\eta}\right)_{k\in\mathbb{N}}$, and so the difference $\left(x^{\alpha-\varepsilon+(j-k)\eta}_{\alpha-\varepsilon+i\eta} - \operatorname{rank} x^{\alpha-\varepsilon+(j-k)\eta}_{\alpha-\varepsilon+(i+1)\eta}\right)$ is the same for all values of k beyond some finite

[12] The part of $\mathrm{D}\mathcal{X}$ that lies in the quadrant $Q^{\mathrm{NE}}_{\alpha-\varepsilon}$ is fully determined by the ranks of the homomorphisms $x^\beta_{\beta'}$ for $\beta' \geq \beta \geq \alpha - \varepsilon$, and the same goes for $\mathrm{D}\tilde{\mathcal{X}}$. We refer the reader to Section 3 in [Chazal et al. 2009] for the technical details.

[13] See Definition 3.2 in [Chazal et al. 2009] and recall that coordinates are reversed here because time flows from $+\infty$ to $-\infty$.

threshold k_m. We then have $\mu(\alpha-\varepsilon+i\eta,\ \alpha-\varepsilon+(j-k)\eta)=0$ for all $k>k_m$, and the sum in Eq. (15) is a finite sum where most of the terms of Eq. (16) cancel out:

$$\begin{aligned}\mu^{\mathrm{tot}}_{\eta,i,j}(\mathrm{D}\mathcal{X}_{\alpha-\varepsilon+\eta\mathbb{Z}}) \;=\; &\mu(\alpha-\varepsilon+i\eta,-\infty)+\left(\mathrm{rank}\ x_{\alpha-\varepsilon+i\eta}^{\alpha-\varepsilon+(j+1)\eta}-\mathrm{rank}\ x_{\alpha-\varepsilon+(i+1)\eta}^{\alpha-\varepsilon+(j+1)\eta}\right)\\ &-\left(\mathrm{rank}\ x_{\alpha-\varepsilon+i\eta}^{\alpha-\varepsilon+(j-k_m)\eta}-\mathrm{rank}\ x_{\alpha-\varepsilon+(i+1)\eta}^{\alpha-\varepsilon+(j-k_m)\eta}\right).\end{aligned} \tag{17}$$

In addition, the term $\left(\mathrm{rank}\ x_{\alpha-\varepsilon+i\eta}^{\alpha-\varepsilon+(j-k_m)\eta}-\mathrm{rank}\ x_{\alpha-\varepsilon+(i+1)\eta}^{\alpha-\varepsilon+(j-k_m)\eta}\right)$ in Eq. (17) is by definition[14] equal to the multiplicity of point $(\alpha-\varepsilon+i\eta,-\infty)$. Hence,

$$\mu^{\mathrm{tot}}_{\eta,i,j}(\mathrm{D}\mathcal{X}_{\alpha-\varepsilon+\eta\mathbb{Z}})=\mathrm{rank}\ x_{\alpha-\varepsilon+i\eta}^{\alpha-\varepsilon+(j+1)\eta}-\mathrm{rank}\ x_{\alpha-\varepsilon+(i+1)\eta}^{\alpha-\varepsilon+(j+1)\eta}. \tag{18}$$

The same is true for $\tilde{\mathcal{X}}_{\alpha-\varepsilon+\eta\mathbb{Z}}$ (which is tame since $\mathcal{X}_{\alpha-\varepsilon+\eta\mathbb{Z}}$ is), that is:

$$\mu^{\mathrm{tot}}_{\eta,i,j}(\mathrm{D}\tilde{\mathcal{X}}_{\alpha-\varepsilon+\eta\mathbb{Z}})=\mathrm{rank}\ \tilde{x}_{\alpha-\varepsilon+i\eta}^{\alpha-\varepsilon+(j+1)\eta}-\mathrm{rank}\ \tilde{x}_{\alpha-\varepsilon+(i+1)\eta}^{\alpha-\varepsilon+(j+1)\eta}. \tag{19}$$

Assuming now that $i>j\geq -1$, *i.e.* that the endpoint of the vertical half-line $\{\alpha-\varepsilon+i\eta\}\times[-\infty,\alpha-\varepsilon+j\eta]$ lies on or above the horizontal line $y=\alpha-\varepsilon-\eta$, we have $\tilde{x}_{\alpha-\varepsilon+i\eta}^{\alpha-\varepsilon+(j+1)\eta}=x_{\alpha-\varepsilon+i\eta}^{\alpha-\varepsilon+(j+1)\eta}$ and $\tilde{x}_{\alpha-\varepsilon+(i+1)\eta}^{\alpha-\varepsilon+(j+1)\eta}=x_{\alpha-\varepsilon+(i+1)\eta}^{\alpha-\varepsilon+(j+1)\eta}$, and so $\mu^{\mathrm{tot}}_{\eta,i,j}(\mathrm{D}\tilde{\mathcal{X}}_{\alpha-\varepsilon+\eta\mathbb{Z}})=\mu^{\mathrm{tot}}_{\eta,i,j}(\mathrm{D}\mathcal{X}_{\alpha-\varepsilon+\eta\mathbb{Z}})$ by Eqs. (18) and (19). Since this is true for any $\eta>0$, it follows from the definition of persistence diagram that the total multiplicities of the diagrams $\mathrm{D}\mathcal{X}$ and $\mathrm{D}\tilde{\mathcal{X}}$ in any vertical half-line $\{\beta'\}\times[-\infty,\beta]$ with $\beta'>\beta\geq\alpha-\varepsilon$ are the same. We may thus further assume that the multi-bijection $\gamma_X:\mathrm{D}\mathcal{X}\to\mathrm{D}\tilde{\mathcal{X}}$ defined above is such that γ_X and γ_X^{-1} move the points within the lower-right quadrant $Q^{\mathrm{SE}}_{\alpha-\varepsilon}$ vertically, in addition to keeping the points within the upper-right quadrant $Q^{\mathrm{NE}}_{\alpha-\varepsilon}$ fixed.

The same construction as in Eq. (14) can be applied to the tame persistence module $\mathcal{Y}$, thus yielding another tame persistence module $\tilde{\mathcal{Y}}$. By the same sequence of arguments as above, we know that there is a multi-bijection $\gamma_Y:\mathrm{D}\mathcal{Y}\to\mathrm{D}\tilde{\mathcal{Y}}$ such that γ_Y and γ_Y^{-1} move the points within $Q^{\mathrm{SE}}_{\alpha-\varepsilon}$ vertically while keeping the points within $Q^{\mathrm{NE}}_{\alpha-\varepsilon}$ fixed.

Observe now that the newly-introduced persistence modules $\tilde{\mathcal{X}}$ and $\tilde{\mathcal{Y}}$ are (strongly) ε-interleaved. Indeed, let $\{\phi_\beta: X^\beta\to Y^{\beta-\varepsilon}\}_{\beta\geq\alpha}$ and $\{\psi_\beta: Y^\beta\to X^{\beta-\varepsilon}\}_{\beta\geq\alpha}$ be two families of homomorphisms that make $\mathcal{X}$ and $\mathcal{Y}$ (strongly) ε-interleaved above time α. We define two new families of homomorphisms between $\tilde{\mathcal{X}}$ and $\tilde{\mathcal{Y}}$, indexed over $\mathbb{R}$, as follows:

$$\begin{cases}\forall\beta\geq\alpha,\ \tilde{\phi}_\beta=\phi_\beta \text{ and } \tilde{\psi}_\beta=\psi_\beta,\\ \forall\beta<\alpha,\ \tilde{\phi}_\beta=0 \text{ and } \tilde{\psi}_\beta=0.\end{cases}$$

The fact that these two families of homomorphisms make the diagrams of Eq. (2) commute for all $\beta'\geq\beta\geq\alpha$ comes from the fact that $\{\phi_\beta\}_{\beta\geq\alpha}$ and $\{\psi_\beta\}_{\beta\geq\alpha}$ themselves make the diagrams commute. The fact that the families $\{\tilde{\phi}_\beta\}_{\beta\in\mathbb{R}}$ and $\{\tilde{\psi}_\beta\}_{\beta\in\mathbb{R}}$ make the diagrams commute across and below time α comes from the fact that they are identically zero below time α. Thus, $\tilde{\mathcal{X}}$ and $\tilde{\mathcal{Y}}$ are (strongly) ε-interleaved over whole $\mathbb{R}$, which implies by the Extended Stability Theorem (Theorem 4.4 in [Chazal et al. 2009]) that there is a multi-bijection $\tilde{\gamma}:\mathrm{D}\tilde{\mathcal{X}}\to\mathrm{D}\tilde{\mathcal{Y}}$ that moves the points by at most ε in the l^∞-distance. The map $\gamma=\gamma_Y^{-1}\circ\tilde{\gamma}\circ\gamma_X$ is then a multi-bijection $\mathrm{D}\mathcal{X}\to\mathrm{D}\mathcal{Y}$ satisfying assertions (i) through (iv) of Theorem 8.2. This concludes the proof of Lemma 8.3.

[14] See Definition 3.2 and the related comments in Section 3.1 of [Chazal et al. 2009].