Proceedings of Symposia in Applied Mathematics

Topological Data Analysis

Afra Zomorodian

Abstract. Scientific data is often in the form of a finite set of noisy points, sampled from an unknown space, and embedded in a high-dimensional space. Topological data analysis focuses on recovering the topology of the sampled space. In this chapter, we look at methods for constructing combinatorial representations of point sets, as well as theories and algorithms for effective computation of robust topological invariants. Throughout, we maintain a computational view by applying our techniques to a dataset representing the conformation space of a small molecule.

1. Introduction

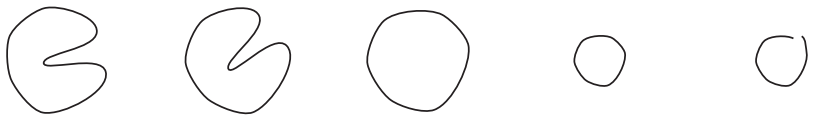
Topological data analysis is a subarea of computational topology that develops topological techniques for robust analysis of scientific data. To clarify our task, we begin this chapter by examining the three words that constitute the title. We then lay out a two-step pipeline around which the rest of the chapter is organized. We focus on intuition in this section, formalizing the concepts in the remainder of the chapter.

1.1. Topology. Geometry studies shapes. For instance, we think of the closed curve in Figure 1(a) as having the same shape as the curve in Figure 1(b), even though the two curves are not identical pointwise. If we translate the first curve by about an inch, and rotate it by 30 degrees, we get the second curve. Even though we have transformed the curve, we believe its shape has not changed. In this sense, geometry classifies objects according to properties that do not change under certain permissible transformations. Felix Klein introduced this expansive definition of geometry in his famous *Erlangen Program* in 1872 [**45**]. Restricting to the group of *rigid* transformations yields *Euclidean geometry.* Through its rigidity, this geometry has a fine granularity when viewed as a classification system. If we enlarge the group of approved transformations, we may obtain other classifications that are coarser and may capture more *qualitative* information about shapes.

2010 *Mathematics Subject Classification.* Primary 55N35, 55U05, 55-04; Secondary 55U10, 68T10, 62H99.

Key words and phrases. Čech, alpha, Vietoris-Rips, witness, simplicial complex, cubical complex, persistent homology, multidimensional persistence, zigzag, tidy sets.

This chapter was completed at Dartmouth College, where the author's research was partially supported by ONR N 00014-08-1-0908 and NSF CAREER CCF-0845716.



(a) Curve (b) Rigidly moved (c) Deformed (d) Shrunk (e) Cut

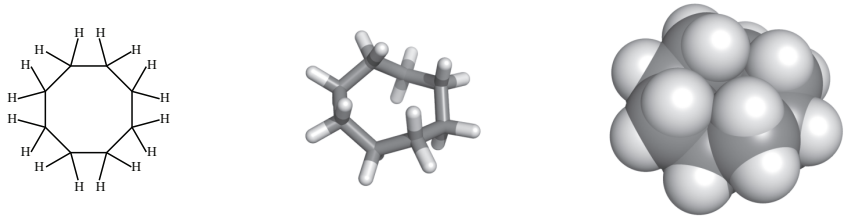
FIGURE 1. The closed curve (a) is transformed under rigid motion (b), deformed using homeomorphisms (c) including scaling (d), and finally cut (e). The final transformation changes the curve's connectivity as the closed loop becomes a path.

Topology allows the larger group of *homeomorphisms* that deform an object by stretching or shrinking, as we do for the curve in Figures 1(c) and (d). Under any homeomorphism, the curve remains a *Jordan curve* that divides the plane into two regions. It is only by cutting the curve that we change its *topology* from a closed loop to a path. Neither *cutting*, nor its inverse, *gluing*, are permissible in topology as they change the way an object is connected. Topology, then, classifies a shape according to its *connectivity*, such as its number of pieces, loops, or presence of boundary. The main object of study in topology is a *topological space*: the most general form of a space that still retains a notion of connectivity.

1.2. Data. Data is processed mainly on digital computers and communicated via packet switching. As such, data is stored in a finite representation, such as the IEEE standard for floating-point arithmetic, resulting in *discretization error*. Moreover, acquisition devices are usually imperfect, adding *noise* to data. Therefore, we think of *data* as a finite set of discrete noisy samples, such as two-dimensional images from digital cameras, terrains from satellite observations [**61**], sampled three-dimensional surfaces from laser scanners [**67**], voxelized MRI scans of the human body [**59**], or snapshots from simulated protein folding trajectories [**32**]. Abstract spaces may also be modeled with discrete samples, as the following example demonstrates.

EXAMPLE 1.1 (conformation space). To understand molecular motion, we need to characterize the molecule's possible shapes. For instance, consider the molecule *cyclooctane* with formula C_8H_{16}. Structurally, cyclooctane has a ring of eight carbon atoms, each bonded to a pair of hydrogen atoms, as shown in its chemical diagram in Figure 2(a) [**4**]. A *conformation* of a molecule is a potential shape it may assume. We visualize a conformation of cyclooctane using two models in Figures 2(b) and (c). To specify a conformation, we need to map every atom in the molecule to a point in $\mathbb{R}^3$. Since cyclooctane has 24 atoms, each conformation, such as the one in Figure 2, may be viewed as a single point in $\mathbb{R}^{72}$. The set of all physically realizable conformations of a molecule is its *conformation space*. We may model this space with a set of finite samples. Figure 3 shows three-dimensional projections of a dataset of 6,400 samples from the cyclooctane conformation space [**4**].

1.3. Analysis. Suppose that we are given a set of data points S embedded in some d-dimensional space $\mathbb{Y}$. We assume that this data is sampled from some



(a) Chemical Diagram (b) One conformation (Stick) (c) Hard Sphere Model

FIGURE 2. Cyclooctane: Chemical diagram (a) and two visualizations of a conformation (b) and (c).

unknown k-dimensional subspace $\mathbb{X} \subseteq \mathbb{Y}$, where $k \leq d$. Both the geometry and the topology of $\mathbb{X}$ are lost during sampling. Our goal in *analysis* is recovering information about $\mathbb{X}$ from the given dataset S. Properties of the embedding space $\mathbb{Y}$ are *extrinsic*, while properties of the unknown space $\mathbb{X}$ are *intrinsic*. For example, S has extrinsic dimension d, but intrinsic dimension k. In analysis, we try to recover intrinsic information, given only extrinsic information.

EXAMPLE 1.2 (spiral). Consider the spiral in Figure 4. The data points (a) are embedded in $\mathbb{R}^2$, so, the extrinsic dimension is 2. We may use the Euclidean metric of the embedding space to compute distances between points (b). The points are sampled, however, from a spiral (c). Since the spiral is a one-dimensional curve, its intrinsic dimension is 1. Note that the *geodesic* distance (d) may be very different from the *embedding* distance in (b).

Every analysis method makes fundamental assumptions about the unknown space $\mathbb{X}$. *Principal Component Analysis (PCA)* assumes that $\mathbb{X}$ is a linear subspace, a flat hyperplane with no curvature [**40, 73**]. ISOMAP assumes that $\mathbb{X}$ is intrinsically flat, but is isometrically embedded, like the spiral in Figure 4(c). The method also assumes that $\mathbb{X}$ is a single convex patch with the topology of a disc [**24**]. The method of *Hessian eigenmaps*, a refinement of *locally linear embeddings (LLE)* [**65**], also assumes isometric embedding, but relaxes the restriction on topology [**24**]. A large class of methods from computer graphics and computational geometry focus on

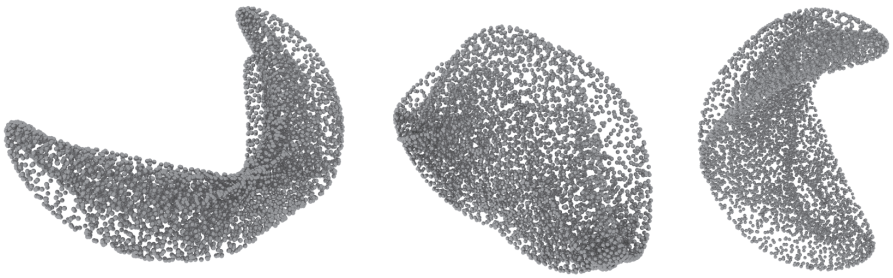


FIGURE 3. Three-dimensional projections of 6,400 samples of the conformation space of cyclooctane [**4**].

(a) Samples in $\mathbb{R}^2$ (b) Embedding (c) Spiral (d) Geodesic

FIGURE 4. Spiral. Two-dimensional samples (a) and extrinsic embedding distance (b). Original spiral (c) and intrinsic geodesic distance (d).

surface reconstruction from samples. These methods assume that $\mathbb{X}$ is a closed surface without self-intersection. Additionally, they often assume that $\mathbb{X}$ is smooth and that the sampling is sufficiently dense, respecting the unknown *local feature size* of the original surface [**23**].

The methods above are all instances of *manifold learning*, where the key assumption is that $\mathbb{X}$ is a *manifold*, that is, it is locally Euclidean [**60**]. But most real-world point sets are sampled from spaces that violate nearly all the above assumptions.

EXAMPLE 1.3 (reconstruction). It is already clear from Figure 3 that the conformation space of cyclooctane in Example 1.1 has non-manifold structure, visible as potential self-intersections in the three-dimensional embeddings. Indeed, the reconstructed conformation space is a two-dimensional surface with non-manifold structure [**50**]. We embed this surface in $\mathbb{R}^3$ using ISOMAP in Figure 5. Topologically, the conformation space is the Klein bottle glued to the two-dimensional sphere along two rings [**49**].

Note that the conformation space of cyclooctane violates every assumption made by prior analysis techniques. Reconstructing a surface with non-manifold

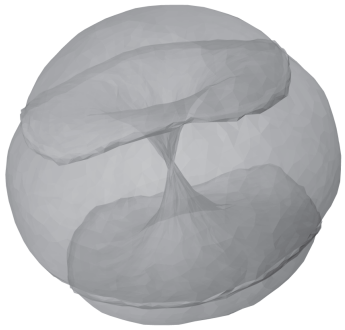
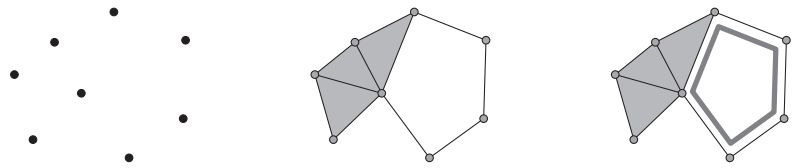


FIGURE 5. The reconstructed conformation space of cyclooctane is a two-dimensional surface with self-intersections [**50**]. Topologically, it is the Klein bottle glued to the sphere along two rings [**49**].



(a) Points S (b) Representation K (c) Invariant

FIGURE 6. Analysis Pipeline. The input is a set of points S (a). Section 3 describes the first step, the geometric process of going from (a) to a representation K (b). Sections 4 and 5 describe the second step, the combinatorial process of going from (b) to a topological invariant, such as the cycle (c).

structure is a challenging problem in computational geometry [**50**]. Once we have the surface, we may also recover the topology of the conformation space. But if we are only interested in topology, surface reconstruction is excessive as topology is a much coarser classification system.

Having examined the three words in *topological data analysis*, we may now define our task in this chapter.

DEFINITION 1.4 (topological data analysis). Given a finite dataset $S \subseteq \mathbb{Y}$ of noisy points sampled from an unknown space $\mathbb{X}$, *topological data analysis* recovers the topology of $\mathbb{X}$, assuming both $\mathbb{X}$ and $\mathbb{Y}$ are topological spaces.

The assumption here is much weaker than those of geometric analysis techniques: We do not assume manifold structure, smoothness, lack of curvature, or the existence of a metric. Correspondingly, our goal is modest and coarse.

1.4. Pipeline. Traditional topological analysis uses the two-step pipeline summarized in Figure 6. Given a finite set of points (a):

(1) We first approximate the unknown space $\mathbb{X}$ in a combinatorial structure K, as shown in Figure 6(b). We devote Section 3 to such structures and the methods for constructing them.
(2) We then compute topological invariants of K, such as the cycle in Figure 6(c). We devote Section 4 to classic topological invariants and Section 5 to modern multiscale invariants.

Topological invariants of K provide approximations to properties of $\mathbb{X}$ as finitely represented by S. While the pipeline is effective, it is not computationally feasible for large point sets embedded in high dimensions. For this reason, we describe methods for combining the two steps in Section 6.

Topological data analysis is an applied field, concerned with theory that facilitates analysis of real-world datasets. Therefore, we return at the end of each section to our motivating dataset, the cyclooctane conformation space, applying techniques toward its analysis and providing empirical results. Throughout this chapter, all computation is on a 64-bit GNU/Linux machine with a 2.4 GHz dual-core Xeon processor and 2 GB RAM. Our software is not threaded and uses only one core.

2. Background

We begin by formalizing topological spaces and describing two topological classifications. We then introduce simplicial complexes, the primary combinatorial structure that we will use for representation We end this section by specifying a general scheme that is the basis for a few of the methods for constructing complexes in the next section. Throughout, our aim is not to be comprehensive, but pedagogical. Recent surveys on topological analysis include Ghrist [**35**] and Carlsson [**5**]. For a broader introduction to computational topology, see [**78**].

2.1. Topology. Intuitively, a topological space is a set of points, each of whom knows its neighbors. A *topology* on a set X is a subset $T \subseteq 2^X$ such that:

(1) If $S_1, S_2 \in T$, then $S_1 \cap S_2 \in T$.
(2) If $\{S_J \mid j \in J\} \subseteq T$, then $\cup_{j \in J} S_j \in T$.
(3) $\emptyset, X \in T$.

The pair $\mathbb{X} = (X, T)$ is a *topological space*. A set $S \in T$ is an *open set* and its complement in X is *closed*. We often abuse notation by using $p \in \mathbb{X}$ for $p \in X$ when the topology is clear from context. A subset $A \subseteq X$ with *induced topology* $T_A = \{S \cap A \mid S \in T\}$ is a *subspace* $\mathbb{A}$ of $\mathbb{X}$. A familiar example of a topological space is the *d-dimensional Euclidean space* $\mathbb{R}^d$, where we use the Euclidean metric to measure distances and define open sets. We may also turn any subset of a Euclidean space into a topological space by using the induced topology.

A function $f : \mathbb{X} \to \mathbb{Y}$ is *continuous* if for every open set A in $\mathbb{Y}$, $f^{-1}(A)$ is open in $\mathbb{X}$. A *homeomorphism* $f \colon \mathbb{X} \to \mathbb{Y}$ is a bijection such that both f and f^{-1} are continuous. Given a homeomorphism $f \colon \mathbb{X} \to \mathbb{Y}$, we say that $\mathbb{X}$ is *homeomorphic* to $\mathbb{Y}$. As homeomorphism is an equivalence relation on topological spaces, we also say that $\mathbb{X}$ and $\mathbb{Y}$ have the same *topological type*, denoted $\mathbb{X} \approx \mathbb{Y}$. The topological type is the finest level of classification available in topology.

A *homotopy* is a family of maps $f_t : \mathbb{X} \to \mathbb{Y}$, $t \in [0, 1]$, such that the associated map $F : \mathbb{X} \times [0, 1] \to \mathbb{Y}$ given by $F(x, t) = f_t(x)$ is continuous. Here, $\mathbb{X} \times [0, 1]$ is a topological space whose open sets are products of the open sets of $\mathbb{X}$ and the open sets of $[0, 1]$, viewed as a subspace of $\mathbb{R}$ [**38**]. Then, $f_0, f_1 : \mathbb{X} \to \mathbb{Y}$ are *homotopic* via the homotopy f_t, denoted $f_0 \simeq f_1$. A map $f : \mathbb{X} \to \mathbb{Y}$ is a *homotopy equivalence* if there exists a map $g : \mathbb{Y} \to \mathbb{X}$, such that $f \circ g \simeq 1_{\mathbb{Y}}$ and $g \circ f \simeq 1_{\mathbb{X}}$, where $1_{\mathbb{X}}, 1_{\mathbb{Y}}$ are the identity maps on the respective spaces. Given a homotopy equivalence $f : \mathbb{X} \to \mathbb{Y}$, we say that $\mathbb{X}$ and $\mathbb{Y}$ are *homotopy equivalent* and have the same *homotopy type*, denoted $\mathbb{X} \simeq \mathbb{Y}$, as homotopy equivalence is an equivalence relation on topological spaces. A space with the homotopy type of a point is *contractible*. Homotopy type is a coarser classification than topological type. For example, a disc is contractible, but not homeomorphic, to a point.

2.2. Simplicial Complex. Simplicial complexes are popular in topological data analysis due to their structural simplicity. Intuitively, a simplicial complex is similar to a hypergraph, where we represent a relationship between $(n + 1)$ nodes with a n-dimensional simplex. Formally, a *simplicial complex* is a set K of finite sets closed under the subset relation: If $\sigma \in K$ and $\tau \subseteq \sigma$, then $\tau \in K$. Here, σ is a *simplex* (plural *simplices*) and τ is a *face* of σ, its *coface*. The (-1)-*simplex* $\emptyset$ is a face of any simplex. A simplex is *maximal* if it has no proper coface in K. If $\sigma \in K$ has cardinality $|\sigma| = n + 1$, we call σ a n-*simplex* of *dimension* n, denoted

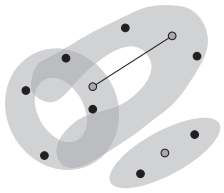


FIGURE 7. A set of 8 black points, an open cover of 3 sets, and the nerve of the cover: a simplicial complex with 3 vertices and 1 edge.

$\dim(\sigma) = n$. Generalizing, if the maximum dimension of a simplex in K is d, we call K a d-dimensional complex, $\dim(K) = d$.

Our notion of dimensionality for simplices stems from our ability to realize a n-simplex geometrically as a n-dimensional subspace of $\mathbb{R}^d$, $d \geq n$, namely, the convex hull of $(n+1)$ affinely-independent points [**38**]. In this view, an n-simplex is called a *vertex*, an *edge*, a *triangle*, or a *tetrahedron* for $0 \leq n \leq 3$, respectively. A key property of a realized simplex is that it is contractible. A simplicial complex K may be embedded in Euclidean space as the union of its geometrically realized simplices such that they only intersect along shared faces. This union is the *underlying space* $|K| = \cup_{\sigma \in K} \sigma$ of K, a topological space. Topological invariants, such as homotopy type, do not depend on a particular geometric realization of a complex.

EXAMPLE 2.1. Figure 6(b) displays a geometric realization of a simplicial complex with 8 vertices, 11 edges, and 3 triangles. The triangles and four of the edges defining the hole are maximal.

A *subcomplex* is a subset $L \subseteq K$ that is also a simplicial complex. An important subcomplex is the *n-skeleton* consisting of simplices in K of dimension less than or equal to n. The 1-skeleton of a simplicial complex is a graph.

2.3. Cover and Nerve. For the rest of this chapter, we assume we are given a finite set of data points S, sampled from some unknown space $\mathbb{X}$, and embedded in some topological space $\mathbb{Y}$, as described in Section 1.3. A key idea in topological analysis is to approximate $\mathbb{X}$ locally using pieces of the embedding space $\mathbb{Y}$. An *open cover* of S is

$$U = \{\mathbb{U}_i\}_{i \in I}, \quad \mathbb{U}_i \subseteq \mathbb{Y},$$

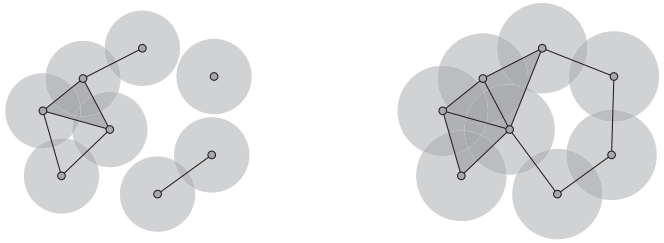
where I is an indexing set, $S \subseteq \cup_i \mathbb{U}_i$, and $\mathbb{U}_i$ are open. The *nerve* N of U is

(1) $\emptyset \in N$, and
(2) If $\cap_{j \in J} \mathbb{U}_j \neq \emptyset$ for $J \subseteq I$, then $J \in N$.

Clearly, the nerve is a simplicial complex.

EXAMPLE 2.2. Figure 7 displays a set of 8 black points, an open cover of 3 sets, and the nerve of this cover: a simplicial complex with 3 vertices and 1 edge.

The union of the sets in an open cover is our approximation of the unknown $\mathbb{X}$. Its nerve serves as a finite combinatorial representation to be used in computation. If the sets in the cover do not hide interesting topology, either within themselves or in their intersection patterns, all topology is exposed within the nerve. Formally, a cover U is *good* if all $\mathbb{U}_i$ are contractible and so are all their nonempty finite



(a) C_{ϵ_1} (b) C_{ϵ_2}, $\epsilon_2 > \epsilon_1$

FIGURE 8. The Čech complex C_ϵ is the nerve of a cover of ϵ-balls. We show the complex at two scales $0 < \epsilon_1 < \epsilon_2$.

intersections. Clearly, the cover in Figure 7 is not good as the leftmost set is an annulus, and its intersection with the middle set has two pieces. By Leray's *Nerve Lemma*, the nerve of a good cover is homotopy equivalent to the cover, that is, the union of the sets in the cover [**3, 64**]. This lemma is the basis of a few of the methods for representing point sets in the next section.

3. Combinatorial Representations

In this section, we focus on the first step of the analysis pipeline in Figure 6. Our input is a finite point set $S \subseteq \mathbb{Y}$. We construct combinatorial representations K that approximate the space $\mathbb{X}$ from which S was sampled. In the remainder of this section, we assume that $\mathbb{Y}$ is a metric space with metric $\mathrm{d}\colon \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}$. A number of the algebraic methods may be extended to non-metric spaces easily, while the geometric methods, such as the alpha complex, require the Euclidean metric. As promised, we end this section by constructing a representation for the cyclooctane dataset.

3.1. Čech Complex. Let $B_\epsilon(x)$ be the *open ball* of radius ϵ centered at x. That is, for $\epsilon \in \mathbb{R}$ and $x \in \mathbb{Y}$,

$$B_\epsilon(x) = \{y \in \mathbb{Y} \mid \mathrm{d}(x, y) < \epsilon\}.$$

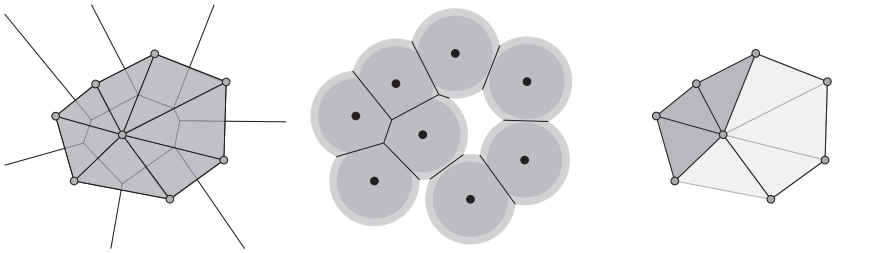
Given $S \subseteq \mathbb{Y}$ and $\epsilon \in \mathbb{R}$, we center an ϵ-ball at each point to get a cover:

$$U_\epsilon = \{B_\epsilon(x) \mid x \in S\}.$$

The *Čech complex* C_ϵ is the nerve of this cover [**38**]. Since balls are convex and convex sets are contractible, the cover is good and its nerve captures the topology of the cover.

EXAMPLE 3.1. Figures 8 show covers U_{ϵ_1} and U_{ϵ_2} at two scales $0 < \epsilon_1 < \epsilon_2$. The nerve of each cover is drawn above it. Note that each nerve is homotopy equivalent to its cover: The cover and nerve in (a) both have 3 components and 1 hole, while the cover and nerve in (b) both have 1 component and 1 hole.

We may compute a Čech complex at each scale ϵ. Clearly, $C_0 = \emptyset$ and C_∞ is an $(|S|-1)$-simplex. That is, the Čech complex may have a much higher dimension than the embedding space $\mathbb{Y}$. Since an n-simplex has 2^{n+1} faces, the complex may become massive at higher scales.



(a) Voronoi & Delaunay (b) Restricted Voronoi regions (c) Alpha subcomplex A_{ϵ_2}

FIGURE 9. A dataset, its Voronoi diagram and its nerve, the Delaunay complex (a). The restricted Voronoi regions (b) form a cover whose nerve is the alpha complex (c), shown as a subcomplex of the lighter Delaunay complex.

The Čech complex is not computed in practice due to its computational complexity. The uniform ball radii imply an assumption of uniform sampling on the input, which is not valid in real-world datasets. We could use non-uniform radii to form a cover, and this idea has been explored in other methods, such as the alpha complex described in the next section.

3.2. Alpha Complex. To reduce the size of the complex, we limit its dimension by using the geometry of the embedding space. Given $S \subseteq \mathbb{Y}$, the *Voronoi region* $R(x)$ of a point $x \in S$ is the set of points in $\mathbb{Y}$ closest to it:

$$R(x) = \{y \in \mathbb{Y} \mid d(x, y) \leq d(x', y), \forall x' \in S,\, x' \neq x\}.$$

The *Voronoi diagram* is the set of all Voronoi regions for points in S. This diagram may be viewed as a closed cover for $\mathbb{Y}$. The *Delaunay complex* is the nerve of the Voronoi diagram. The Voronoi cover and its nerve are fundamental geometric objects and have been extensively studied within computational geometry [**20**].

EXAMPLE 3.2. Figure 9(a) displays the Voronoi diagram for our example point set, and overlays its nerve, the Delaunay complex.

We now use the Voronoi diagram to restrict the interactions of the ϵ-ball cover from the previous section. For each point $x \in S$, we intersect its ϵ-ball and Voronoi region to get a *restricted Voronoi region*. The set of all restricted regions forms a new cover:

$$U_\epsilon = \{B_\epsilon(x) \cap R(x) \mid x \in S\}.$$

The *alpha complex* A_ϵ is the nerve of this cover [**27, 28**]. By construction, $A_0 = \emptyset$, A_∞ is the Delaunay complex, and A_ϵ is a subcomplex of the Delaunay complex, for any ϵ. Moreover, the alpha and Čech complexes are homotopy equivalent. Unlike the Čech complex, however, the maximum dimension of the alpha complex is limited to the embedding dimension, provided S is in *general position*, a theoretical assumption that may be enforced computationally [**76**].

EXAMPLE 3.3. Figure 9(b) overlays the restricted Voronoi regions for our example point set at the two scales used in Figure 8. Figure 9(c) shows the alpha complex A_{ϵ_2} as a subcomplex of the Delaunay complex. At this scale, the complex is the same as the Čech complex C_{ϵ_2} in Figure 8(b).

(a) Graph & Cliques (b) VR complex $V_{2\epsilon_1}$

FIGURE 10. The highlighted maximal cliques of the 2ϵ-neighborhood graph (a) become the maximal simplices of the the VR complex $V_{2\epsilon_1}$ (b).

We construct alpha complexes by first building the Delaunay complex. For each simplex of the Delaunay complex, we compute the minimum scale at which the simplex enters the Alpha complex. Then, we sort the simplices by their minimum scale to get a partial order of simplices. We may now form the alpha complex at any scale ϵ using this ordering. Since the Delaunay complex is finite, the alpha complex may change only at a finite number of critical scales as we increase the scale ϵ from 0 to infinity.

Using uniform radii implies an implicit assumption of uniform sampling. This assumption may be removed by generalizing the alpha complex. We may assume, for instance, that the point set is weighted, where the weight of a point is related to the local feature size. We may then use the *power metric* to define a *power diagram* cover and its nerve, the *regular triangulation* [**1, 29**]. Alternatively, we may define non-uniform radii using the local density of the point set itself to get the *conformal alpha complex* [**12**].

Efficient algorithms and software exist for computing Delaunay complexes, and in turn, alpha complexes in 2 and 3 dimensions [**14**], so the complex is well-suited for topological analysis in low dimensions. The construction of the Delaunay complex is difficult in higher dimensions, although progress is being made [**2**].

3.3. Vietoris-Rips Complex. The Vietoris-Rips complex is popular in topological analysis due to the ease of its construction even in higher dimensions. Unlike the previous complexes, it is based on a graph, instead of a cover. Given $S \subseteq \mathbb{Y}$ and $\epsilon \in \mathbb{R}$, let $G_\epsilon = (S, E_\epsilon)$ be the *ϵ-neighborhood graph* on S, where

$$E_\epsilon = \{\{u, v\} \mid \mathrm{d}(u, v) \leq \epsilon,\ u \neq v \in S\}.$$

A *clique* in a graph is the subset of vertices that induces a complete subgraph [**18**]. A clique is *maximal* if it cannot be made any larger. The *clique complex*, also called the *flag complex*, has the maximal cliques of a graph as its maximal simplices [**46**]. The *Vietoris-Rips complex* V_ϵ is the clique complex of the ϵ-neighborhood graph [**37, 74**]. We refer to the complex as the *VR complex* for brevity.

EXAMPLE 3.4. Figure 10 shows the construction of a VR complex for our point set. We begin with a $2\epsilon_1$-neighborhood graph (a) so that the VR complex is comparable to Čech and alpha complexes at scale ϵ_1. The graph has 5 maximal cliques, highlighted by gray ovals. Each maximal clique becomes a maximal simplex

in the VR complex $V_{2\epsilon_1}$ (b). Note that the VR complex is different than the Čech complex C_{ϵ_1} in Figure 8(a).

As the example illustrates, the VR complex is not always homotopy equivalent to the Čech complex, so we may view it as an approximation. Clearly, $C_\epsilon \subseteq V_{2\epsilon}$. Moreover, the Čech and VR complexes can be shown to be related homologically [**36**]. We will describe this classification level of topology in Section 4.

Like its Čech counterpart, the VR complex may be as large as a $(|S|-1)$-dimensional simplex. This extremity occurs whenever the neighborhood graph is complete. In practice, we usually only require and construct a n-skeleton for some $n \leq |S|$. We also compute the VR complex $V_{\hat{\epsilon}}$ at some maximum scale $\hat{\epsilon} \in \mathbb{R}$. For each simplex $\sigma \in V_{\hat{\epsilon}}$, we compute the minimum ϵ at which the simplex enters the VR complex, with the vertices entering at $\epsilon = 0$ and the edges at their length. We then sort the simplices according to this value, extracting the VR complex for any $0 < \epsilon \leq \hat{\epsilon}$ as a prefix of this ordering. As for the alpha complex, since the VR complex is finite, there is only a finite number of critical scales at which the complex changes.

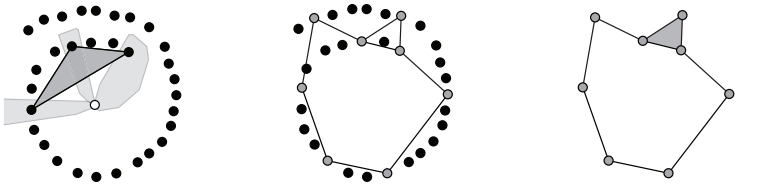
For analysis of small point sets in low dimensions, the VR complex is usually computed using ad-hoc methods. For an in-depth study of its construction for large point sets in higher dimensions, see [**79**]. Public software for building VR complexes is available [**56, 66**]. Currently, the VR complex is one of the few practical methods for topological analysis in high dimensions.

3.4. Witness Complex. Since the VR complex may be massive, we try to approximate it with smaller number of vertices. We motivate the complex in this section by reinterpreting the Delaunay complex from Section 3.2.

Example 3.5. Consider the spiral point set in Figure 11(a). The triangle is in the Delaunay complex because the Voronoi regions of its three vertices intersect in the white Voronoi vertex. The white vertex is equidistant from the three vertices of the triangle.

Given $S \subseteq \mathbb{Y}$, a *strong witness* $w \in \mathbb{Y}$ is equidistant from the points in $\sigma \subseteq S$, *witnessing* the creation of a Delaunay simplex σ, such as the triangle in the example. We are motivated to search for strong witnesses within $\mathbb{Y}$ to construct the Delaunay complex, but this approach is not feasible as the set of strong witnesses has measure zero. So, we relax the definition of a witness: A *weak witness* $w \in \mathbb{Y}$ is closer to points in $\sigma \subseteq S$ than $S - \sigma$. The set of weak witnesses for a n-simplex form its region in the order-$(n+1)$ Voronoi diagram of S and has positive measure [**20**]. Moreover, if a simplex and all its faces have weak witnesses, the simplex also has a strong witness [**21**].

We may build the Delaunay complex on a sample set $S \subseteq \mathbb{Y}$, allowing witnesses to be anywhere in $\mathbb{Y}$. The resulting Delaunay complex captures the topology of $\mathbb{Y}$. But we are interested in the unknown space $\mathbb{X}$ with our only knowledge being the set of samples $S \subseteq \mathbb{Y}$. Therefore, we mimic the process above and replace $\mathbb{Y}$ with S. Let $L \subseteq S$ be the set of *landmarks* and the remaining points, $W = S - L$ be the set of potential witnesses. For $\epsilon \in \mathbb{R}$, the *ϵ-witness graph* is the graph $G = (L, E_\epsilon)$, where $\{l_1, l_2\} \in E_\epsilon$ if there exists a weak witness $w \in W$ that is closer to l_i than any other landmark, and $\mathrm{d}(w, l_i) \leq \epsilon$ for $i = 1, 2$. The *(weak) witness complex* W_ϵ is the clique complex of this ϵ-witness graph.



(a) Strong witness (b) ∞-witness graph (c) Witness complex W_∞

FIGURE 11. The triangle in (a) is Delaunay as the Voronoi regions of its vertices intersect in the white vertex. The witness graph (b) is built on the gray landmark points, with the remaining black points acting as witnesses. The witness complex (c) is the clique complex of this graph.

EXAMPLE 3.6. Figure 11(b) shows the witness graph for the spiral point set with $\epsilon = \infty$. The graph is built on the gray landmarks, with the remaining black points acting as potential witnesses for the edges. The witness complex W_∞ in Figure 11(c) is the clique complex of this graph.

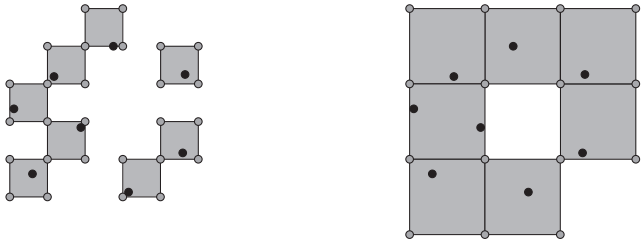
It is clear that the witness complex depends on the chosen landmarks. The choice and size of the landmark set remains an art rather than a science. For analysis, it is best to bootstrap by choosing multiple sets of landmark points and seeing if the result is replicable.

The weak witness complex is the simplest of a family of witness complexes [**21**]. The software package *JPlex* [**66**] can compute several types of witness complexes. The witness graph is easily constructable by computing the $|L| \times |S|$ distance matrix using *k-nearest neighbors* [**57**]. Since it is a clique complex, the weak witness complex may be expanded from this graph using the algorithms designed for the Vietoris-Rips complex [**79**]. Currently, the witness complex is one of the few practical methods for topological analysis of large datasets.

3.5. Cubical Complex. A cubical complex is another type of combinatorial structure used in topological analysis. Informally, a cubical complex is a *cell complex*, where the cells are now *cubes* of different dimensions, rather than simplices [**38**]. A n-cube is called a *vertex*, an *edge*, a *square*, or a *cube* for $0 \leq n \leq 3$, respectively. Like a geometrically realized simplicial complex, a pair of cubes in a cubical complex only intersect along shared faces, which are lower-dimensional cubes.

Given $S \subseteq \mathbb{Y}$, where $\mathbb{Y}$ is a d-dimensional Euclidean space, we may easily construct a cubical complex at scale ϵ by covering $\mathbb{Y}$ with a grid of d-dimensional cubes with side ϵ. The *cubical complex* Q_ϵ is simply the rasterization of S on this grid: If an ϵ-cube c contains any point $s \in S$, then $c \in Q_\epsilon$. The cubical complex is dependent on the orientation of the grid. Also, all cubes are maximal and of the same dimension, so the complex is *pure*.

Alternatively, we may view a cubical complex as a cover, taking the nerve to get a simplicial complex. However, there is no need to do this, as all algorithms that require a simplicial complex as input extend easily to other cell complexes, such as the cubical complex.



(a) Q_{ϵ_1} (b) $Q_{2\epsilon_1}$

FIGURE 12. Cubical complexes Q_ϵ on top of grids for our point set at scales ϵ_1 and $2\epsilon_1$. Compare with Čech, alpha, and VR complexes in Figures 8, 9(c), and 10(b), respectively.

EXAMPLE 3.7. Figure 12 shows cubical complexes Q_{ϵ_1} and $Q_{2\epsilon_1}$ extracted from conforming grids. While the cubical complexes are not comparable to simplicial complexes combinatorially, they may capture similar topological features. The cubical complex Q_{ϵ_1} has the same homotopy type as the VR complex $V_{2\epsilon_1}$ in Figure 10(b), and the cubical complex $Q_{2\epsilon_1}$ has the same homotopy type as the Čech complex C_{ϵ_2} in Figure 8(b), as well as the alpha complex A_{ϵ_2} in Figure 9(c).

Cubical complexes arise naturally in analysis of two- and three-dimensional rasterized images, such as in the discrete simulation of *dynamical systems* [**42**]. Thresholding a grayscale image, we get an black and white image, where we may interpret the set of black pixels or voxels as a cubical complex. Since these complexes are based on grids, each cube may only be connected to neighboring cubes. This regularity in connectivity allows for tailored algorithms and heuristics to compute topological invariants of cubical complexes [**15, 63**]. We recommend the chapter by Marian Mrozek in this volume as an introduction to current techniques in dynamical systems and cubical homology.

3.6. Analysis. We have now provided multiple structures and methods for the first step of the analysis pipeline depicted in Figure 6. We end this section by completing this step for our motivating dataset of the cyclooctane conformation space from Example 1.1. Recall that the cyclooctane has 8 carbon and 16 hydrogen atoms. The locations of the carbons determine the locations of the hydrogens through energy minimization, so we limit our parameterization to the coordinates of the carbons [**4**]. Therefore, the *cyclooctane dataset* $\mathcal{S}$ is a set of 6,400 points embedded in $\mathbb{Y} = \mathbb{R}^{24}$.

We use the VR complex from Section 3.3 as $\mathcal{S}$ is not too large and is embedded in a high-dimensional Euclidean space [**79**]. To get an idea of scale in $\mathcal{S}$, we compute the maximum interpoint distance between closest pairs of points. This distance is 0.18, so we set the maximum scale to be $\hat{\epsilon} = 0.4$. We first build the neighborhood graph at this scale in 0.29 seconds. With 6,400 vertices and 76,657 edges, the graph is sparse with only 0.4% of the possible edges. We construct the 4-skeleton of the VR complex in 10.83 seconds using the INCREMENTAL-VR algorithm [**79**]. The resulting complex has 3,034,973 simplices, but only 66,179 critical ϵ values at which the complex grows. As described in Section 3.3, we may now extract the

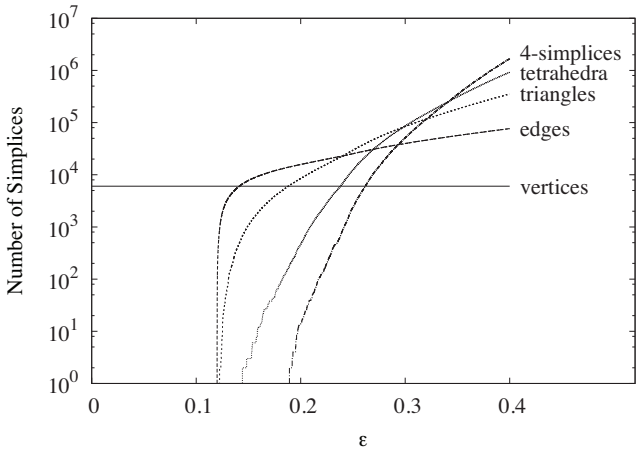


FIGURE 13. Number of n-simplices in the 4-skeleton of the VR complex V_ϵ for the cyclooctane dataset, with $0 \leq n \leq 4$ and $0 \leq \epsilon \leq 0.4$.

complex V_ϵ for any $0 < \epsilon \leq 0.4$. Figure 13 plots the number of n-simplices in this ϵ range for $0 \leq n \leq 4$. The size of the complex grows exponentially with dimension, as expected.

4. Topological Invariants

We now assume that we have a combinatorial representation K, such as a simplicial or cubical complex. In this section, we start on the second step of the analysis pipeline depicted in Figure 6: Computing topological invariants. We begin by formally defining an invariant. We then introduce two classic topological invariants: the Euler characteristic and homology. We conclude the section by returning to the cyclooctane dataset, analyzing the VR complex just built in the previous section.

4.1. Definition. Recall from Section 2.1 that homeomorphisms provide the finest level of classification under topology. The *Homeomorphism Problem* asks whether topological spaces $\mathbb{X}$ and $\mathbb{Y}$ are homeomorphic. This problem is undecidable even when restricted to manifolds of dimension greater than three [**48**]. Since we remain interested in analyzing real-world datasets, we need effective algorithms, so we must lower our expectations and look for partial solutions. These partial solutions come in the form of topological invariants.

Formally, a *topological invariant* is a map f that assigns the same object to homeomorphic spaces, that is:

$$\mathbb{X} \approx \mathbb{Y} \implies f(\mathbb{X}) = f(\mathbb{Y})$$

Note that an invariant is only useful through its contrapositive,

$$f(\mathbb{X}) \neq f(\mathbb{Y}) \implies \mathbb{X} \not\approx \mathbb{Y},$$

TABLE 1. Cell complexes K that are homotopy equivalent to the 1-sphere $\mathbb{S}^1$ have Euler characteristic $\chi(K) = 0$.

Figure	K	type	# n-cells			$\chi(K)$
			0	1	2	
8(b)	C_{ϵ_2}	Čech	8	11	3	0
11(c)	W_∞	witness	8	9	1	0
12(b)	$Q_{2\epsilon_1}$	cubical	15	22	7	0

as the converse of an implication is not true. Therefore, the *trivial* invariant that assigns the same object to all spaces is useless. On the other hand, the *complete* invariant that assigns different objects to non-homeomorphic spaces solves the homeomorphism problem. Most invariants are *incomplete*, falling in the spectrum between these two extremes. An incomplete topological invariant is a classification that is coarser than, but respects, the topological type. In general, the more powerful an invariant, the harder it is to compute it. Naturally, we look for invariants that assign finitely representable objects, as we intend to store them on computers.

We have already seen one topological invariant: homotopy equivalence. Since it is an invariant, we have $\mathbb{X} \approx \mathbb{Y} \implies \mathbb{X} \simeq \mathbb{Y}$. Unfortunately, the general problem of homotopy equivalence is also intractable [**48**], so we must look for less powerful invariants.

4.2. Euler Characteristic. Our first invariant assigns a single integer to a topological space. Let K be any cell complex, such as a simplicial or a cubical complex. The *Euler characteristic* $\chi(K)$ is

$$\chi(K) = \sum_{\sigma \in K} (-1)^{\dim \sigma} = \sum_{n=0}^{\dim K} (-1)^n c_n, \tag{4.1}$$

where c_n is the number of n-dimensional cells in K. The Euler characteristic is an integer invariant for $|K|$ up to homotopy type, so we get the same integer for different complexes whose underlying spaces are homotopy equivalent. That is, for cell complexes K_1, K_2, $|K_1| \simeq |K_2| \implies \chi(K_1) = \chi(K_2)$.

EXAMPLE 4.1 ($\mathbb{S}^1$). The *1-dimensional sphere (1-sphere)* $\mathbb{S}^1$ is a space that is homeomorphic to a circle, such as any closed curve in Figure 1. The 1-sphere has Euler characteristic $\chi(\mathbb{S}^1) = 0$. Table 1 lists three complexes from the previous section that are homotopy equivalent to the 1-sphere. It also verifies that their Euler characteristic is zero using Equation (4.1).

4.3. Simplicial Homology. Instead of an integer, the *homology* invariant assigns a group to a topological space. Homology is quite popular in topological data analysis as it is effectively computable. We define homology for simplicial complexes, but the theory extends to arbitrary topological spaces, and the algorithms extend to arbitrary cell complexes, such as cubical complexes [**38**].

Let K be a simplicial complex, and suppose we fix an order on its set of vertices. An *orientation* of a n-simplex $\sigma = \{v_0, v_1, \ldots, v_n\} \in K$, is an equivalence class of orderings on its vertices, where $(v_0, v_1, \ldots, v_n) \sim (v_{\tau(0)}, v_{\tau(1)}, \ldots, v_{\tau(n)})$ if the parity of the permutation τ is even. An *oriented simplex* is a simplex with an orientation, denoted as sequence $[\sigma]$. For notation brevity, we list oriented simplices as strings

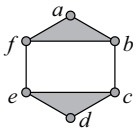


FIGURE 14. A 2-dimensional simplicial complex for Example 4.2.

rather than sequences, so $[v_0, v_1, v_2] \equiv v_0v_1v_2$. The *nth chain group* $C_n(K)$ *of* K is the free Abelian group on K's set of oriented n-simplices. We will abuse notation by dropping K in the notation when the complex is clear from context. An element $c \in C_n$ is an n-chain, $c = \sum_i c_i[\sigma_i]$, with n-simplices $\sigma_i \in K$ and *coefficients* $c_i \in \mathbb{Z}$. Given such a chain c, the *boundary homomorphism* $\partial_n \colon C_n \to C_{n-1}$ is a homomorphism defined linearly by its action on any oriented simplex in c:

$$\partial_n[v_0, \ldots, v_n] = \sum_i (-1)^i [v_0, \ldots, \hat{v_i}, \ldots, v_n],$$

where $\hat{v_i}$ indicates that v_i is deleted from the vertex sequence. We also define $\partial_0 \equiv 0$. A fundamental property of the boundary operator is that $\partial_n \circ \partial_{n+1} \equiv 0$ for all $n \geq 0$. The boundary operator connects the chain groups into a *chain complex* C_*:

$$\cdots \to C_{n+1} \xrightarrow{\partial_{n+1}} C_n \xrightarrow{\partial_n} C_{n-1} \to \cdots .$$

Given any chain complex, the *nth homology group* H_n is:

$$H_n = \ker \partial_n \,/\, \operatorname{im} \partial_{n+1}, \tag{4.2}$$

where ker and im are the *kernel* and *image* of a linear operator, respectively. An n-chain z is an *n-cycle* if $z \in \ker \partial_n$; it is also an *n-boundary* if $z \in \operatorname{im} \partial_{n+1}$. Since $\partial_n \circ \partial_{n+1} \equiv 0$, all boundaries are cycles and $\operatorname{im} \partial_{n+1}$ forms a subgroup of $\ker \partial_n$. Two cycles in the same homology class are *homologous*. Homology is a invariant for $|K|$ up to homotopy type. That is, for simplicial complexes K_1, K_2, $|K_1| \simeq |K_2| \implies H_n(K_1) = H_n(K_2)$ for all $n \geq 0$.

EXAMPLE 4.2. Consider the simplicial complex with labeled vertices in Figure 14. We place the alphabetic ordering on the vertices. The triangle $\{a, b, f\}$ has two orientations: $[a, b, f] = -[b, a, f]$, or $abf = -baf$ using our string notation. The 1-chain $ab + bc$ has boundary

$$\partial_1(ab + bc) = \partial_1(ab) + \partial_1(bc) = (b - a) + (c - b) = c - a,$$

so the 0-chain $c - a$ is a 0-boundary as it is in $\operatorname{im} \partial_1$. The boundary of the 1-chain $h = bc + ce + ef - bf$ is 0, so $h \in \ker \partial_1$ is a 1-cycle. Since h does not bound a 2-chain, h belongs to a non-trivial homology class. The 1-cycle $h' = bc + cd + de + ef - bf$ is homologous to h as their difference is the boundary $cd + de - ce$. Both cycles describe the single hole in the complex.

In order to understand the structure of algebraic invariants, such as homology, we use the following three-step approach:

(1) Correspondence,
(2) Classification, and
(3) Parameterization.

In the first step, we identify the algebraic structure. In the second step, we obtain a complete classification of the structure, up to isomorphism. In the third step, we parameterize the classification. We follow this approach for homology of a simplicial complex [**26, 38**]:

(1) Correspondence: The nth homology H_n of a simplicial complex is a group, or equivalently, a $\mathbb{Z}$-module, where $\mathbb{Z}$ is the *ring of coefficients*. We may, instead, construct modules over other rings R. Since the complex K is finite, H_n becomes a finitely generated R-module.
(2) Classification: Suppose R is a principle ideal domain (PID), such as $\mathbb{Z}$. Any finitely generated R-module decomposes uniquely into the form:
$$\bigoplus_{i=1}^{\beta_n} R \oplus \bigoplus_{j=1}^{m} R/t_j R,$$
for integers $\beta_n \geq 0$ and nonzero nonunit elements $t_j \in R$, such that $t_j | t_{j+1}$.
(3) Parameterization: The left direct sum is the *free* submodule and is characterized by its *Betti number* $\beta_n = \operatorname{rank} H_n$. The right direct sum is the *torsion* submodule and is characterized by its *torsion coefficients* t_j. The set of $m+1$ elements $\{\beta_n\} \cup \{t_j\}_j$ is the parameterization. Over a field k of coefficients, H_n simplifies to a k-vector space with *dimension* $\beta_n = \dim H_n$, so the parameterization is simply the integer β_n.

There is a one-to-one correspondence between the parameterization and finitely generated R-modules, so this parameterization is a complete invariant up to isomorphism. We have a full characterization of homology, provided we compute over PIDs.

The invariance of the Euler characteristic is derived from the invariance of homology. For a topological space $\mathbb{X}$, the *Euler-Poincaré formula* states that

$$\chi(\mathbb{X}) = \sum_n (-1)^n \beta_n. \tag{4.3}$$

Compare the formula with the previous definition in Equation (4.1) in Section 4.2. This formula emphasizes that χ can be defined purely in terms of homology and depends only on the homotopy type of $\mathbb{X}$. That is, $\chi(\mathbb{X})$ is independent of the choice of cell complex representing $\mathbb{X}$.

For torsion-free spaces in three-dimensions, the Betti numbers have intuitive meaning as a consequence of the *Alexander Duality*: β_0 counts the number of connected *components*; β_1 is the rank of any basis for the *tunnels*; β_2 counts the number of enclosed spaces or *voids*.

Example 4.3. Table 2 lists the Betti numbers for some of the topological spaces and cell complexes that we have seen so far. For instance, the reconstructed surface has one component, one tunnel, and two voids. The table also lists the Euler characteristics for the spaces, this time computed by Equation (4.3). We see that homology is a more refined invariant than the Euler characteristic. For example, the surface and C_{ϵ_1} have the same χ, but different β_0 and β_2. We can distinguish the two spaces with homology, but not with the Euler characteristic.

Having characterized homology, we next turn to its computation. Since the boundary operator $\partial_n \colon C_n \to C_{n-1}$ is linear, it has a matrix M_n in terms of a choice of bases for C_n and C_{n-1}. We may use oriented n-simplices as a basis for C_n

TABLE 2. Topological spaces and cell complexes, and their Betti numbers β_n and Euler characteristics χ.

Figure	Space	β_0	β_1	β_2	χ
1(a)	curve	1	1	0	0
4(c)	spiral	1	0	0	1
5	surface	1	1	2	2
8(a)	C_{ϵ_1}	3	1	0	2
8(b)	C_{ϵ_2}	1	1	0	0
10(b)	$V_{2\epsilon_1}$	3	0	0	3
11(c)	W_∞	1	1	0	0
12(a)	Q_{ϵ_1}	3	0	0	3
12(b)	$Q_{2\epsilon_1}$	1	1	0	0
14	complex	1	1	0	0

in each dimension. Computing the kernel and image in Equation (4.2) is equivalent to computing the *null space* of the matrix for ∂_n, and the *range space* of the matrix for ∂_{n+1}, respectively.

Over PIDs, the *reduction* algorithm reduces each matrix to the *Smith normal form*, from which the parameterization may be read [**26**]. Over $\mathbb{Z}$, neither the size of the matrix entries nor the number of operations in $\mathbb{Z}$ is polynomially bounded for reduction. There are sophisticated polynomial algorithms based on modular arithmetic [**69**], although reduction is still preferred in practice [**25**].

Over fields, C_n is a vector space in each dimension and we compute its dimension with *Gaussian elimination* matches that of matrix multiplication [**72**]. In practice, topological analysis nearly always uses the field of two elements $\mathbb{Z}_2 = \mathbb{Z}/2\mathbb{Z}$ for coefficients, which simplifies computation even further. Each simplex is its own inverse so there is no need for orientation. The matrices have 0 or 1 entries, so the columns may be stored sparsely as lists of simplices with coefficient 1. We use only *elementary column operations* in Gaussian elimination to reduce the matrix to *column echelon form* and read off the dimension.

EXAMPLE 4.4. Over $\mathbb{Z}_2$, the matrix for ∂_1 for the complex in Figure 14 is:

$$\left[\begin{array}{c|cccccccc} & ab & bc & cd & de & ef & af & bf & ce \\ \hline a & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ b & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ c & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ d & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ e & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ f & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{array}\right],$$

where we augment the matrix to show the bases for C_1 and C_0. Applying Gaussian elimination, we reduce the matrix to column echelon form:

$$\left[\begin{array}{c|cccccccc} & ab & bc & cd & de & ef & z_1 & z_2 & z_3 \\ \hline a & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ b & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ c & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ d & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ e & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ f & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{array}\right],$$

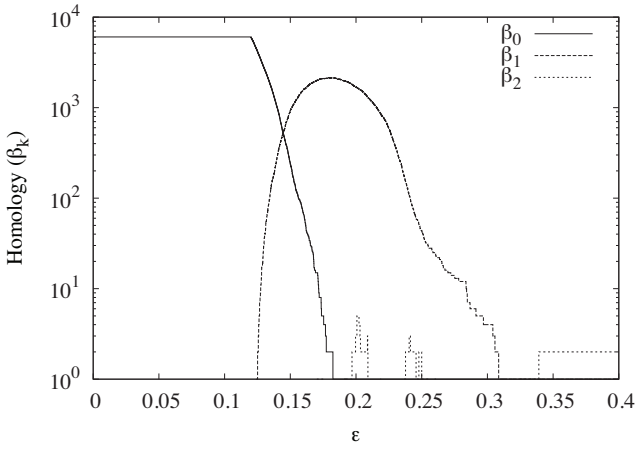


FIGURE 15. The Betti numbers β_n of the VR complex V_ϵ for the cyclooctane dataset $\mathcal{S}$, with $0 \leq n \leq 2$ and $0 \leq \epsilon \leq 0.4$.

where the basis elements

$$z_1 = af + ab + bc + cd + de + ef,$$
$$z_2 = bf + ab + bf,$$
$$z_3 = ce + cd + de,$$

are the generators of $\ker \partial_1$, so $\dim(\ker \partial_1) = 3$. Note that the three generators form a vector space of cycles in the 1-skeleton of the complex in Figure 14. Repeating the process for ∂_2, we get $\dim(\operatorname{im} \partial_2) = 2$. Therefore,

$$\dim H_1 = \dim(\ker \partial_1) - \dim(\operatorname{im} \partial_2) = 3 - 2 = 1,$$

and homology has captured the central hole in the complex.

4.4. Single-Scale Analysis. We have now looked at two topological invariants for the second step of the analysis pipeline. We end this section by completing this step for the cyclooctane dataset $\mathcal{S}$ using the 4-dimensional VR complex built in Section 3.6.

We compute homology over $\mathbb{Z}_2$ coefficients in 13.35 seconds using the persistence algorithm that we will encounter in Section 5.2. Figure 15 graphs the Betti numbers β_n for V_ϵ, $0 \leq \epsilon \leq 0.4$ and $0 \leq n \leq 2$. The Betti number β_3 is identically zero and is not plotted. We also do not consider β_4, as homology requires 5-simplices to determine if a 4-cycle is a boundary, but we only provide the 4-skeleton. As expected, the complex becomes connected starting at $\epsilon = 0.18$, the maximum interpoint distance of closest points. The Betti numbers of the complex match those of the conformation space surface for all $\epsilon \geq 0.3391$. Since we do not know the correct scale, however, we would not be able to determine the Betti numbers of the conformation space from this graph alone. But the complexes at different scales are related to each other. Our success in topological analysis requires analysis across scale to determine the topological features of the unknown space.

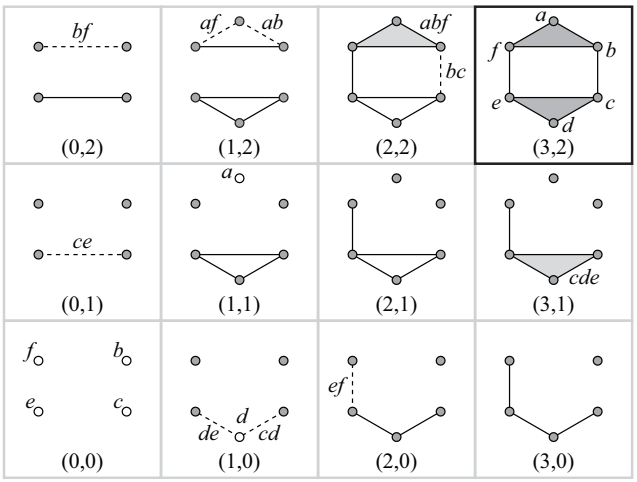


FIGURE 16. A bifiltration of the complex in Figure 14, now with coordinate $(3, 2)$. Simplices are highlighted and named at their critical coordinates.

5. Multiscale Invariants

In this section, we provide multiscale solutions for the second step of the analysis pipeline. We extend our combinatorial representation to approximate the unknown space at multiple scales. We then introduce three modern multiscale invariants that analyze the topology of the resulting multiscale structure, identifying robust features that persist across scale. We conclude, as usual, by applying our multiscale tools to the analysis of cyclooctane dataset.

5.1. Multifiltration Model. Our first model is based on notions from *Morse theory* [**52**]. Let $\mathbb{N} \subseteq \mathbb{Z}$ be the set of non-negative integers. For vectors in $\mathbb{N}^d$ or $\mathbb{R}^d$, we say $u \leq v$ if $u_i \leq v_i$ for all $1 \leq i \leq d$. The relation $\leq$ forms a partial order on $\mathbb{N}^d$ and $\mathbb{R}^d$. A cell complex K is *multifiltered* if we are given a family of subcomplexes $\{K_u\}_u$, where $u \in \mathbb{R}^d$, so that $K_u \subseteq K_v$ whenever $u \leq v$. Intuitively, a multifiltered complex only *grows* with increasing coordinate, so the model describes a monotonically growing space. We call the family of subspaces $\{K_u\}_u$ a *multifiltration*. A one-dimensional multifiltration is a *filtration*. All the simplicial methods in Section 3 give rise to filtrations with increasing ϵ, but cubical complexes do not as the vertices change at different scales.

A *critical coordinate* u for cell $\sigma \in K$ is a minimal coordinate, with respect to the partial order $\leq$, such that $\sigma \in K_u$. A multifiltered complex K where each cell σ has a unique critical coordinate u_σ is *one-critical* [**9**].

EXAMPLE 5.1. Figure 16 shows a two-dimensional multifiltration, a *bifiltration*, of the complex in Figure 14. The bifiltration is one-critical, with each simplex being highlighted and named at its unique critical coordinate.

A one-critical multifiltration is a natural model for scientific data. Suppose a sampled dataset $S \subseteq \mathbb{Y}$ is augmented with $d-1$ real-valued functions $f_j \colon S \to \mathbb{R}$

with $d > 1$. The functions measure information about the unknown space $\mathbb{X}$ at each point.

EXAMPLE 5.2 (graphics). In *computer graphics*, one approach to rendering surfaces is to construct a digitized model. A three-dimensional object is sampled by a range scanner that employs multiple cameras to sense the surface position as well as normals and textures [**71**]. Here, S is the set of positions, while the functions f_j are surface attributes, such as normal and texture, sampled at S.

We begin by approximating S with a filtered complex K, using any method that yields a filtration, such as the simplicial methods in Section 3. Suppose each cell $\sigma \in K$ enters the complex at scale $\epsilon(\sigma)$. To incorporate the functions f_j into topological analysis, we first extend them to the cells in the complex. For $\sigma \in K$ and f_j, let $f_j(\sigma)$ be the maximum value f_j takes on σ's vertices; that is, $f_j(\sigma) = \max_{v \in \sigma} f_j(v)$, where $v \in S$. This extension defines $d-1$ functions on the complex, $f_j \colon K \to \mathbb{R}$. We combine all filtration functions into a d-variate function $F \colon K \to \mathbb{R}^d$, where

$$F(\sigma) = (f_1(\sigma), f_2(\sigma), \ldots, f_{d-1}(\sigma), \epsilon(\sigma)) .$$

We multifilter K via the *sublevel sets* $\{K_u\}_u$ of F for $u \in \mathbb{R}^d$:

$$K_u = \{\sigma \in K \mid F(\sigma) \leq u\}.$$

Each simplex σ enters K_u at $u = F(\sigma)$ and remains in the complex for all $u \geq F(\sigma)$. Equivalently, $F(\sigma)$ is the unique critical coordinate at which σ enters the filtered complex. That is, the multifiltrations built by this process are always one-critical.

Finally, since complex K is finite, there are a finite number of critical coordinates in each dimension where the complex grows in the multifiltration. Restricting to the Cartesian product of these critical values, we parameterize the resulting discrete grid using $\mathbb{N}$ in each dimension. This parameterization gives us coordinates in $\mathbb{N}^d$ for a multifiltration, as shown for the bifiltration in Figure 16 [**10**].

5.2. Persistent Homology. We are now interested in the homology of our multiscale model for representing data. That is, we want to know the homology of the complexes at all scales, as well as the relationship between their homologies. Suppose we are given a multifiltration $\{K_u\}_u$, $u \in \mathbb{N}^d$. For each pair $u, v \in \mathbb{N}^d$ with $u \leq v$, $K_u \subseteq K_v$ by definition, so $K_u \hookrightarrow K_v$. Since homology is a functor, this inclusion induces a linear map

$$\iota_n(u, v) \colon H_n(K_u) \to H_n(K_v)$$

that maps an n-dimensional homology class within K_u to the one that contains it within K_v [**38**]. The *nth persistent homology* is $\operatorname{im} \iota_n$, the image of ι_n for all pairs $u \leq v \in \mathbb{N}^d$ [**10**].

For characterization and computation, we begin with one-parameter multifiltrations (filtrations). We follow the same algebraic approach we used for characterizing homology in Section 4.3. For a filtration, we have [**81**]:

(1) Correspondence: The nth persistent homology of a filtration over ring R is a graded $R[t]$-module, where $R[t]$ is the ring of polynomials with indeterminate t over R.

(2) Classification: Over fields k, $k[t]$ is a PID, and any graded $k[t]$-module decomposes uniquely into:

$$\bigoplus_{i=1}^{\ell} \Sigma^{\alpha_i} k[t] \ \oplus \ \bigoplus_{j=1}^{m} \Sigma^{\gamma_j} k[t]/(t^{\delta_j}),$$

where Σ^{α} denotes an α-shift upward in grading, and $\alpha_i, \gamma_j, \delta_j \in \mathbb{N}$.

(3) Parameterization: The classification gives us ℓ half-infinite intervals $[\alpha_i, \infty)$ and m finite intervals $[\gamma_j, \gamma_j + \delta_j)$. The *persistence barcode* is the multiset of these $\ell + m$ intervals and forms the parameterization [**11**].

There is a one-to-one correspondence between the parameterization and finitely generated graded $k[t]$-modules, so this parameterization is a complete invariant, up to isomorphism. Note that while $\mathbb{Z}$ is a PID, $\mathbb{Z}[t]$ is not, so the classification above does not extend to integer coefficients.

The barcode intervals have a natural interpretation. By the theory of persistent homology, each n-simplex either *creates* an n-dimensional homology class, or *destroys* an $(n-1)$-dimensional class by merging it with a class created earlier. For each class, persistence pairs its *creator* $\sigma \in K$ with the *destroyer* $\tau \in K$, if one exists. We may also pair the grades at which σ and τ enter the filtration to get an interval representing each class. This barcode interval is the *lifetime* of the homology class within the filtration. A half-infinite interval $[\alpha_i, \infty)$ represents a class that is created at α_i and still exists within the completed complex. A finite interval $[\gamma_j, \gamma_j + \delta_j)$ represents a class that is created at γ_j and lives only δ_j grades in the filtration, at which point it merges with the boundary class. Alternatively, we may form intervals using the ϵ at which the two simplices enter the filtration, getting a barcode that describes homology with respect to scale.

Example 5.3. In a multifiltration, any path with monotonically increasing coordinates is a filtration, such as the bottom row in the bifiltration in Figure 16. The filtered complex at coordinate $(3, 0)$ has 5 vertices and 3 edges. Figure 17 graphs β_0 for this filtration above the x-axis, where the unit is filtration grade. Below the axis, we see the β_0 barcode. Each interval is the lifetime of a connected component in this filtration. The left endpoint is labeled with the simplex that created the component. The right endpoint is labeled with the simplex that destroyed the component, if such a simplex exists. The component created by simplex d and destroyed by simplex cd immediately has zero lifetime, so we do not draw it. The barcode deconstructs the β_0 graph into a set of intervals. We may recover the β_0

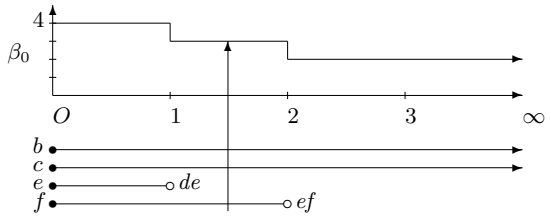


Figure 17. Above the x-axis, β_0 is plotted for the bottom row filtration of Figure 16. Below is the labeled β_0 barcode. Since the vertical arrow intersects three intervals at $x = 1.5$, $\beta_0 = 3$ at that x. The axis unit is filtration grade.

graph by sweeping a vertical line from left to right and counting the number of intervals that the line intersects, as demonstrated by the vertical arrow at $x = 1.5$ in the figure.

Persistence barcodes have been quite useful in topological data analysis. Suppose that a geometric process constructs a filtration so that the lifetime of a homology class denotes its significance. Then, we may use barcodes to separate topological noise from features. We have applied barcodes successfully in a number of areas, including shape description [**17**], biophysics [**43**], and computer vision [**8**].

Having characterized persistent homology, we next turn to its computation. Since $k[t]$ is a PID, the reduction algorithm for homology extends naturally to persistent homology. The matrix for a boundary operator now has polynomial entries from $k[t]$ that encode the filtration ordering. While the field homology of any single complex in a filtration is a vector space, the persistent homology of the filtration has torsion. This means that the reduction algorithm will require both row and column operations to reduce each matrix to Smith normal form.

EXAMPLE 5.4. For the filtration in Example 5.3, we use the n-simplices at their critical coordinates as a basis for the chain group C_n. Over $\mathbb{Z}_2[t]$, the matrix for ∂_1 of the filtration is:

$$\left[\begin{array}{c|ccc} & cd & de & ef \\ \hline b & 0 & 0 & 0 \\ c & t & 0 & 0 \\ d & 1 & 1 & 0 \\ e & 0 & t & t^2 \\ f & 0 & 0 & t^2 \end{array}\right].$$

For instance, $\partial_1(cd) = t \cdot c + 1 \cdot d$, as c enters the filtration one grade earlier than d. We may now reduce this matrix with the reduction algorithm as for regular homology.

Alternatively, we may utilize the *persistence algorithm*, which computes directly on matrices with field entries [**81**]. The algorithm takes advantage of the filtration ordering to require only elementary column operations, allowing it to represent matrices as columns and reduce them to column echelon form. There is a wonderful relationship between the reduction and persistence algorithms [**81**]. The latter has been refined over the years. For its most recent distillation, see [**78**]. The algorithm is implemented in several publicly available software packages [**56, 66**].

Finally, both the reduction and persistence algorithms may also generate descriptions of generators for homology classes, as we do in Example 4.4 by augmenting the boundary matrix. Traditionally, these descriptions are not generated as the focus is on the algebraic characterization of the homology groups. The generators are useful, however, within geometric applications of computational topology [**30, 82**]. We recommend Jeff Erickson's chapter in this volume for an introduction to current results on geometrically optimal generators.

5.3. Multidimensional Persistence. Our success in characterizing homology of filtrations motivates us to move to higher dimensional multifiltrations. Once again, we follow our algebraic approach. For a multifiltration, we have [**10**]:

(1) Correspondence: The nth homology of a multifiltration over field k is an n-graded A_n-module M, where $A_n = k[x_1, \ldots, x_n]$ is the n-graded module of polynomials with n indeterminates over k.
(2) Classification: Unlike its one-dimensional counterpart, A_n is not a PID and A_n-modules have no structure theorem. Nevertheless, we establish a full classification of this structure in terms of three invariants. The first invariant, $\xi_0(M)$ is the multiset of generators for the free approximation of M. The second invariant, $\xi_1(M)$ is the multiset of generators for the *free hull* of M. These invariants have intuitive meaning as analogs of the left and right endpoints of intervals in a barcode, respectively. Unfortunately, there is no way to *match* these endpoints consistently as the remaining invariant corresponds to the set of orbits in a set under group action.
(3) Parameterization: The third invariant corresponds to the set of orbits of an algebraic group action on an algebraic variety. Unfortunately, such a set is not, in general, an algebraic variety. The number of orbits may be uncountable, giving us a *continuous* invariant.

To summarize, no complete invariant exists for persistent homology of multifiltrations, in dimension higher than 1. The discrete invariants ξ_0, ξ_1 above do not capture persistence information, which is contained in the intervals, not their endpoints.

Instead, we may use an incomplete invariant. Recall that persistence is the image of the map $\iota_n(u, v)\colon H_n(K_u) \to H_n(K_v)$. The *nth rank invariant* is

$$\rho_n(u, v) = \operatorname{rank} \iota_n(u, v),$$

for all pairs $u \leq v \in \mathbb{N}^d$ [**10**]. The rank invariant is equivalent to the persistent barcode in the one-dimensional case, so it is complete when it can be. Unlike the barcode, the rank invariant extends to higher dimensions as an incomplete invariant.

We next turn to the computation of multidimensional persistence. We assume we are given a d-dimensional multifiltration of a cell complex K with m cells. Any pair $u \leq v \in \mathbb{N}^d$ defines a two-level one-dimensional filtration, where we map u to 0 and v to 1. We may compute the barcodes for this filtration in $\Theta(m^3)$ time using the persistence algorithm in Section 5.2. We then read $\rho_n(u, v)$ directly from the β_n-barcode: It is the number of intervals that contain both 0 and 1. To compute the full rank invariant, we need to consider all distinct pairs of complexes in a multifiltration that are comparable by the partial order $\leq$. Unfortunately, there are constructions with $\Theta(m^d)$ distinct complexes, implying $\Theta(m^{2d})$ comparable pairs, and a $\Theta(m^{2d+3})$ running time. To store the rank invariant, we also require $\Theta(m^{2d})$ space [**9**]. This is clearly not a feasible method.

For one-critical multifiltrations, described in Section 5.1, we can use more sophisticated algorithms. The n-graded chain modules C_n for one-critical multifiltrations are free, as each cell enters the complex only once. The boundary operator $\partial_n\colon C_n \to C_{n-1}$, in turn, is a homomorphism between free multigraded modules and may be written as a matrix with polynomial entries.

Example 5.5. For the bifiltration in Figure 16, we use n-simplices in their critical coordinates as a basis for chain group C_n. Over $A_2 = \mathbb{Z}_2[x_1, x_2]$, the matrix

for ∂_1 of the multifiltration is:

$$\left[\begin{array}{c|cccccccc} & ab & bc & cd & de & ef & af & bf & ce \\ \hline a & x_2 & 0 & 0 & 0 & 0 & x_1 & 0 & 0 \\ b & x_1x_2^2 & x_1^2x_2^2 & 0 & 0 & 0 & 0 & x_2^2 & 0 \\ c & 0 & x_1^2x_2^2 & x_1 & 0 & 0 & 0 & 0 & x_2 \\ d & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ e & 0 & 0 & 0 & x_1 & x_2^2 & 0 & 0 & x_2 \\ f & 0 & 0 & 0 & 0 & x_1^2 & x_1x_2^2 & x_2^2 & 0 \end{array}\right].$$

Recall from Equation 4.2 that the nth homology group is $H_n = \ker \partial_n \,/\, \operatorname{im} \partial_{n+1}$. To compute homology, we have three tasks, all of which may be recast into problems in computational commutative algebra [**19**].

(1) Compute the boundary module ($\operatorname{im} \partial_{n+1}$): This is the *submodule membership problem*. We first compute a Gröbner basis using the Buchberger algorithm. We then check membership using the division algorithm for multivariate polynomials.
(2) Compute the cycle module ($\ker \partial_n$): The problem is equivalent to computing the *syzygy submodule* using Schreyer's algorithm.
(3) Compute the quotient H_n: We need to test whether the generators of the syzygy submodule are in the boundary submodule. But this is simply an instance of our first task.

While the above solution is theoretically sound, it is practically infeasible. The SMP problem is a generalization of the *polynomial ideal membership problem* at the ring level, and PIMP is already EXPSPACE-complete, requiring exponential space and time [**54**]. The Buchberger algorithm is doubly-exponential, although impractical singly-exponential versions do exist.

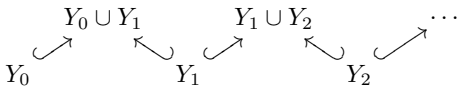
We can exploit the structure provided by a multigrading, however, to derive polynomial-time algorithms. The key insight is that we ensure that the matrix entries are always homogeneous monomials, as in Example 5.5. The resulting multigraded algorithms run in worst-case $O(m^3)$ space and $O(m^7)$ time, where m is the size of the multifiltration [**9**]. Empirically, the time bound seems to be tight. While the reduction in complexity is theoretically significant, this time bound still implies that multiparameter topological analysis is out of reach for large datasets.

5.4. Zigzag Persistence. We end our discussion of topological invariants with recent developments for extracting persistent information for yet another model for scientific data. A primary characteristic of our model in the section so far is that it is monotonically increasing as in a multifiltration $\{K_u\}_u$, subcomplexes nest: $K_u \subseteq K_v$ whenever $u \leq v$. But we have nonmonotonicity in a number of application areas.

EXAMPLE 5.6 (molecular rigidity). *Flexible docking* models biological macromolecules, such as protein-protein complexes, by allowing flexibility near active sites. We would like to identify flexible regions of a protein algorithmically. Based on the molecular conjecture [**70**], now a theorem [**44**], we model a protein by a multigraph, where covalent bonds, hydrogen bonds, salt bridges, and hydrophobic contacts or tethers are represented as edges [**39**]. The program FIRST partitions this multigraph into flexible and rigid regions by extending the *pebble game* to three-dimensions [**31**]. Covalent bonds, however, have picosecond vibrations that cause noncovalent bonds to be unstable, resulting in the *flickering* of edges in the

associated multigraph [**47**]. Flickering edges can be modeled by adding and deleting edges from a dynamically changing cell complex. The resulting history of the complex, however, is no longer a filtration.

Instead, we model nonmonotinicity as a sequence of topological spaces $\{Y_i\}_i \in \mathbb{N}$ that are not necessarily nested. Since any pair of spaces Y_i, Y_j include into their union $Y_i \cup Y_j$, we use unions of consecutive spaces from the sequence to build the following diagram:



where all maps are inclusions. Let $X_{2n} = Y_n$ and $X_{2n+1} = Y_n \cup Y_{n+1}$ for $n \in \mathbb{N}$ to rewrite the diagram as:

$$X_1 \hookrightarrow X_2 \hookleftarrow X_3 \hookrightarrow X_4 \hookleftarrow \cdots X_m,$$

where we have assumed that the resulting sequence has m terms. If our spaces are cell complexes, the right arrows here indicate cell *addition*, and the left arrows indicate cell *deletion*. We generalize this model further by allowing the maps to be homomorphisms:

$$X_1 \to X_2 \leftarrow X_3 \to X_4 \leftarrow \cdots \to X_m, \tag{5.1}$$

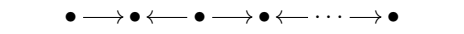
Since some maps could be identities, the general model is a family of complexes with forward or backward homomorphisms in any order. Due to the alternating directions of the maps, Diagram (5.1) is called a *zigzag* [**6**]. Note that if we only have arrows to the right in this diagram, and the arrows are inclusions, we get a diagram for a filtration:

$$X_1 \to X_2 \to X_3 \to X_4 \to \cdots \to X_m, \tag{5.2}$$

Over a field k, the homology of each complex is a k-vector space. Applying the nth homology functor to Diagram (5.1), we get

$$V_1 \to V_2 \leftarrow V_3 \to V_4 \leftarrow \cdots \to V_m, \tag{5.3}$$

where $V_j = H_n(X_j)$ is the nth homology of the jth space and the maps are induced maps at the homology level. Diagrams such as (5.3) are the objects of study in *representation theory* [**22, 34**]. A *quiver* is a pair $Q = (Q_0, Q_1)$, where Q_0 is a finite set of *vertices* and Q_1 is a finite set of *arrows (directed edges)* between them. That is, a quiver is a directed graph. The quiver for Diagram (5.3) is


(5.4)

A quiver has an *underlying undirected graph*. This graph is a *path* for our quiver:

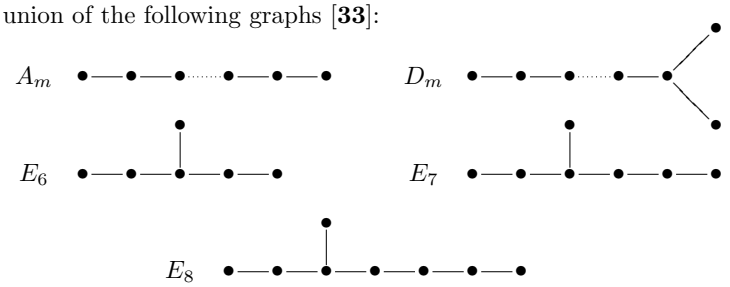
(5.5)

A *representation* V for a quiver Q is a collection $\{V_x \mid x \in Q_0\}$ of finite-dimensional k-vector space together with a collection $\{V_{ab} \colon V_a \to V_b \mid ab \in Q_1\}$ of k-linear maps. Diagram (5.3) is a representation for quiver (5.4). Given representation V, the *dimension vector* $d_V \colon Q_0 \to \mathbb{N}$ is $d_V(x) = \dim_k(V_x)$ for all $x \in Q_0$; that is, $d_V(x)$ is the dimension of the vector space at node x.

As with finitely-generated modules in Section 4.3 or finitely-generated graded $R[t]$-modules in Section 5.2, quivers have a classification theorem stipulating that every representation has a unique decomposition into a direct sum of *indecomposable*

representations, up to isomorphism and permutations of components. A quiver has *finite type* if it decomposes into a finite number of indecomposables. *Gabriel's Theorem* states that a quiver is of finite type iff the underlying undirected graph is a disjoint union of the following graphs [**33**]:

(5.6)



Kac's Theorem states that the set of dimension vectors of indecomposable representations of a quiver Q does not depend on the orientation of the arrows [**41**]. Following our algebraic approach, we have [**6**]:

(1) Correspondence: The nth homology of a zigzag of length m over a field is a representation of a quiver in Diagram (5.4).
(2) Classification: The zigzag quiver has an underlying undirected graph in Diagram (5.5), which is a path of length m, equivalent to type A_m in Diagram (5.6). By Gabriel's theorem, the zigzag quiver has finite type.
(3) Parameterization: By Kac's theorem, the invariant depends only on the underlying undirected graph. The model for one-dimensional persistence in Diagram (5.2) also gives a quiver of finite type A_m. We already know that persistent homology has barcodes as invariants. Therefore, as a corollary of Kac's theorem, zigzag persistence is parameterizable by barcodes.

Having characterized zigzag persistence, we next turn to its computation. From representation theory, one may derive a general scheme for computing zigzag persistence from of a representation, that is, given the set of vector spaces and linear maps as input. Given a topological space $\mathbb{X}$ and a Morse function $f\colon \mathbb{X} \to \mathbb{R}$ defined on it, one may compute the persistent homology of the level sets $f^{-1}(c)$ of this space using zigzag persistence [**7**]. Finally, a recent result gives an algorithm for computing zigzag persistence of a simplicial complex with a sequence of n additions and deletions in $O(M(n) + n^2 \log n)$ time, where $M(n)$ denotes the cost of multiplying two $n \times n$ matrices [**55**]. In this approach, a simplex with multiplicity is treated as multiple different simplices.

5.5. Multiscale Analysis. We have now looked at three multiscale invariants for the second step of the analysis pipeline. We end this section by completing this step for the cyclooctane dataset $\mathcal{S}$ using the 4-dimensional VR complex built in Section 3.6. The complex is already filtered by ϵ, as described in Section 3.3 and we have a filtration without any additional computation. Given a filtration, the natural model for analysis is persistent homology from Section 5.2.

We have already computed persistence barcodes in Section 4.4, as we used the persistence algorithm to compute homology, reading off the Betti numbers using the technique in Figure 17. Once again, there is no need for further computation. Figure 18 draws the 3,475 non-empty intervals of the β_1-barcode of this filtration. Most 1-cycles have short lifetimes: The average lifetime is 0.0382 with a standard deviation of 0.0260. But there is one cycle with a half-infinite interval, meaning

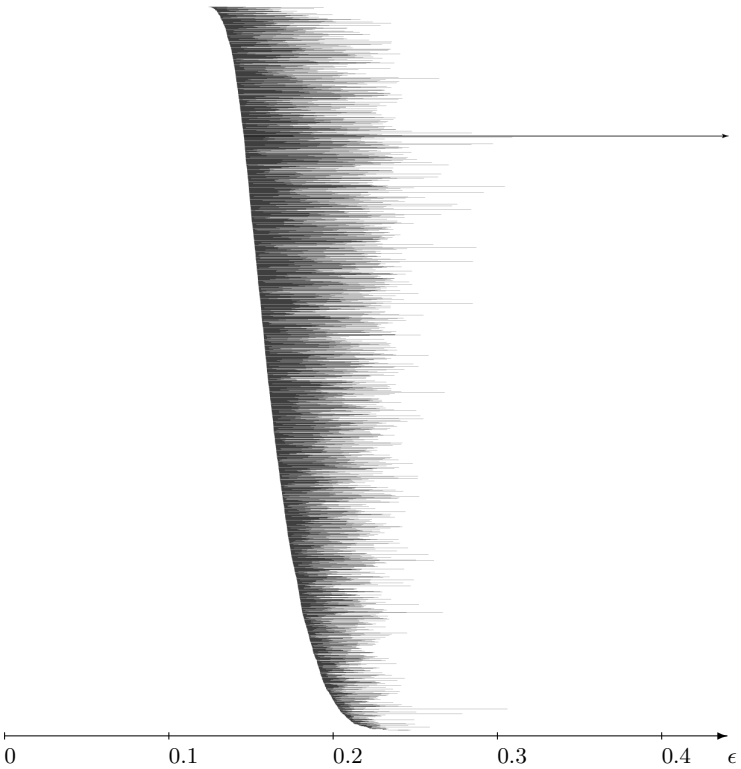


FIGURE 18. Cyclooctane β_1-barcode for the filtered VR complex V_ϵ, $0 \leq \epsilon \leq 0.4$, built in Section 3.6. There are 3,475 non-empty intervals, one of which is half-infinite.

that the cycle's homology class still exists at the maximum scale $\hat{\epsilon} = 0.4$. Even at this scale, this 1-cycle has had a life time of 0.2545, more than 8 standard deviations away from the average. Recall that the VR complex method from Section 3.3 is a geometric process that is based on local distances. Therefore, we are confident that this outlier is a topological feature and $\beta_1 = 1$ for this complex. By a similar procedure, we determine $\beta_0 = 1$ and $\beta_2 = 2$. Our results match the Betti numbers computed from the reconstructed surface of this dataset, as listed in Table 2.

We have now successfully completed a multiscale analysis of the 24-dimensional cyclooctane conformation space dataset containing 6,400 samples. Topological data analysis, using the two-step pipeline in Figure 6, only required 24.47 seconds on a desktop machine. By comparison, geometric reconstruction of the surface using a specialized algorithm is numerically challenging and requires domain knowledge, such as the intrinsic dimension of the unknown space and its types of non-manifold structure [**50**].

6. Reduced Representations

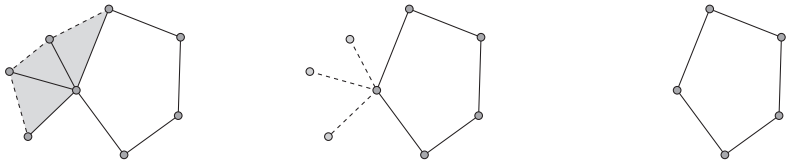
So far, this chapter has been organized around the full implementation of the two-step pipeline in Figure 6. The division of topological analysis into two steps, while useful, stems from a geometric point of view and is somewhat artificial. From an algebraic point of view, the two steps are very much interrelated. Homology is defined on a chain complex, as described in Section 4.3. In our pipeline, the chain complex is always derived from a cell complex built in step one, but this derivation is not a requirement. If we can obtain a chain complex without building a cell complex explicitly, we may still compute homology. Such an approach is desirable given the massive size of the cell complexes that we are now able to build with the methods in Section 3. For example, the simplicial complex representing the cyclooctane dataset has more than three million simplices defined on only 6,400 points.

In this section, we attempt to reduce the size of our representations by combining the two steps of the pipeline. We begin by describing reduction methods that maintain the category of a cell complex, yielding complexes with fewer cells. For even further reduction, we switch category to the simplicial set, a combinatorial structure that allows for collapsed simplices. We next use simplicial sets to define tidy sets, a method for computing homology of any clique complex without its full construction. We end the section by analyzing the cyclooctane dataset using tidy sets.

6.1. Reductions. The traditional approach to dealing with massive complexes has been to reduce the size of the complex before computing homology. This approach is somewhat justified since the reduction algorithm in Section 4.3 has supercubical complexity in the size of the complex over integers, retaining quadratic space and cubic time complexity over fields [**68**]. It is a reasonable to search for heuristics that reduce the size of the complex, while preserving its topology. To be useful, the heuristics must be simple and fast.

There have been a number of reduction techniques proposed for different categories of complexes. For simplicial complexes, the earliest, and perhaps simplest method, is elementary contraction, proposed by Whitehead in defining the simple homotopy type [**75**]. Let K be a simplicial complex. A simplex $\sigma \in K$ has a *free* face $\tau \subseteq \sigma$ if τ has no other cofaces in K. The pair (σ, τ), then, is a *free pair*. An *elementary contraction* removes a free pair (σ, τ) from a complex K. Since an elementary contraction is a *deformation retraction*, a type of homotopy equivalence, the resulting smaller complex $K - \{\sigma, \tau\}$ has the same homotopy type as K [**38**]. But while deformation retractions are continuous, elementary contractions are combinatorial, involving only the deletion of simplices. Elementary contractions may be extended to other cell complexes, such as cubical complexes. Another heuristic is the recent *LC-reduction* [**16**] that produces not only a homotopic, but isomorphic complex [**51**].

Example 6.1 (contraction). Figure 19 displays six elementary contractions in two steps for the complex in (a). In the first step, the three highlighted triangles and their dashed edges form free pairs and are removed in (b). In the second step, the three dashed edges and their highlighted vertices form free pairs and are removed. The final complex in (c) has the same homotopy type as the original complex in (a). It is minimal with respect to elementary contraction.



(a) (b) (c)

FIGURE 19. Elementary contractions. The three highlighted triangles and dashed edges (a) form free pairs and are removed in (b). Then, the dashed edges and highlighted vertices form free pairs and are removed, resulting in a minimal complex (c), homotopy equivalent to (a).

The key property of both heuristics is that the reduced complex remains in the initial category. For instance, elementary contractions on a simplicial complex always yield a simplicial complex. Due to the popularity of simplicial complexes in computational topology, the techniques have been used widely. For *cubical complexes*, the CHOMP project has examined a large number of heuristics over the years, resulting in an array of homology engines [**15, 63**].

A stronger reduction is to collapse a cell into a point, as all cells are contractible by design. Such collapses, however, may change the category type of the structure.

EXAMPLE 6.2 (collapse). Figure 20 shows that collapsing edge bc in triangle abc (a) results in a 2-gon ad (b), which is not a simplicial complex as its two edges are both named ad.

(a) 3-gon abc (b) 2-gon ad

FIGURE 20. The simplicial collapse of edge bc of the triangle abc (a) yields a 2-gon ad (b) that is no longer a simplicial complex.

6.2. Simplicial Sets. To allow for simplicial collapses, we move into the category of simplicial sets. Intuitively, a simplicial set models a well-behaved topological space. One such space is a simplicial complex within which any simplex may be collapsed, and any subset of vertices may be identified.

Let K be a simplicial complex with n-simplices K_n. We define the *simplicial set* X corresponding to K to be a collection of sets $\{X_n\}_n$ together with maps:

$$d_i \colon X_n \to X_{n-1},$$
$$s_i \colon X_n \to X_{n+1},$$

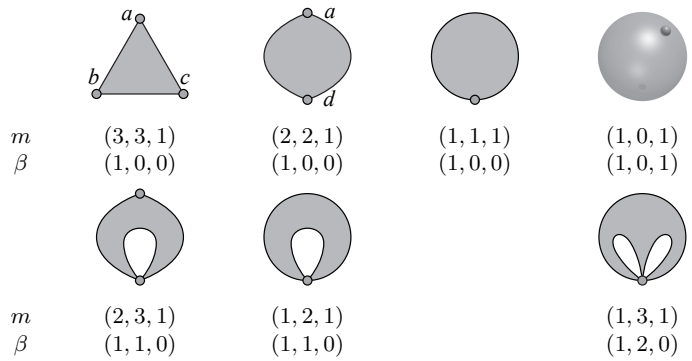


FIGURE 21. The 7 possible 2-simplices in a simplicial set. The triangle *abc* is the only one allowed in a complex. The rest have collapsed edges (top row), identified vertices (bottom row), or both. The vector m counts the non-degenerate simplices and the vector β holds the Betti numbers.

where d_i is the *ith face operator* and s_i is the *ith degeneracy operator* defined as follows:

$$d_i([v_0, \ldots, v_n]) = [v_0, \ldots, \hat{v_i}, \ldots, v_n],$$
$$s_i([v_0, \ldots, v_n]) = [v_0, \ldots, v_i, v_i, \ldots, v_n].$$

That is, the ith face operator d_i deletes the ith vertex, and the ith degeneracy operator s_i repeats it. We now define X_n inductively using the degeneracy operators:

$$X_0 = K_0,$$
$$X_n = K_n \cup \cup_i^n s_i(X_{n-1}), \quad n > 0.$$

It is easy to verify that $\{X\}_n$ together with these operators satisfy the axioms for a simplicial set [**53**]. A simplex $\sigma \in X$ such that $\sigma = s_i(\tau)$ for some $\tau \in X$ is *degenerate* and $\sigma \notin K$. Otherwise, σ is *non-degenerate* and $\sigma \in K$.

EXAMPLE 6.3 (triangle). Figure 21 gives the seven possible 2-simplices in a simplicial set, in contrast to the only possible 2-simplex in a simplicial complex, namely the triangle *abc* on the top left. We now represent the 2-gon from Example 6.2 as well as spaces with different topological types, such as the 2-sphere on the top right corner. For each simplex, the vectors m and β hold the number of non-degenerate simplices and the Betti numbers, respectively. As a simplicial complex K, *abc* has

$$K_0 = \{a, b, c\},$$
$$K_1 = \{ab, bc, ac\},$$
$$K_2 = \{abc\}.$$

As a simplicial set X, abc has

$$\begin{aligned}
X_0 &= \{a, b, c\},\\
X_1 &= \{ab, bc, ac, aa, bb, cc\},\\
X_2 &= \{abc, aab, abb, bbc, bcc, aac, acc, aaa, bbb, ccc\},
\end{aligned}$$

where the set X_n is K_n augmented with degenerate simplices, such as abb, a triangle with one collapsed edge.

Since simplicial sets are capable of representing collapsed simplices, we now incorporate this reduction into the model. Given a simplicial set X and an n-simplex $\sigma \in X$, the *collapse of* σ identifies σ to a single point, giving us a new simplicial set $X' = X/\sigma$. To construct X', we introduce a new vertex v and replace σ, its faces, and its degeneracies, with appropriate degeneracies of v. We first gather σ's non-degenerate k-faces inductively for $k \geq 0$:

$$\bar{F}_k(\sigma) = \begin{cases} \emptyset, & \text{if } k > n,\\ \{\sigma\}, & \text{if } k = n,\\ \cup_{i=0}^{k+1} d_i(\bar{F}_{k+1}(\sigma)), & \text{if } k < n. \end{cases}$$

By adding the degenerate faces, we get all the faces of σ:

$$F_k(\sigma) = \begin{cases} \bar{F}_0(\sigma), & \text{if } k = 0,\\ \bar{F}_k(\sigma) \cup \bigcup_{i=0}^{k-1} s_i(F_{k-1}(\sigma)), & \text{if } k > 0. \end{cases}$$

We replace these faces with degeneracies of v:

$$X'_k = (X_k - F_k(\sigma)) \cup \{s_0^k(v)\},$$

where s_0^k denotes applying the degeneracy operator k times. To complete the definition of X' as a simplicial set, we now define the operators for any $\tau \in X'$:

$$d'_i(\tau) = \begin{cases} d_i(\tau), & \text{if } d_i(\tau) \notin F_{i-1}(\sigma),\\ s_0^{i-1}(v), & \text{otherwise.} \end{cases}$$

$$s'_i(\tau) = \begin{cases} s_i(\tau), & \text{if } s_i(\tau) \notin F_{i+1}(\sigma),\\ s_0^{i+1}(v), & \text{otherwise.} \end{cases}$$

Example 6.4 (2-gon). In Example 6.3, we listed the n-simplices of the triangle abc in Figure 21 as a simplicial set X. We now collapse edge bc to a new vertex d to get the 2-gon $X' = ad$ in the figure. We have

$$\begin{aligned}
\bar{F}_2(bc) &= \emptyset,\\
\bar{F}_1(bc) &= \{bc\},\\
\bar{F}_0(bc) &= F_0(bc) = \{b, c\},\\
F_1(bc) &= \{bc, bb, cc\},\\
F_2(bc) &= \{bbc, bcc, bbb, ccc\},\\
X'_0 &= \{a, d\},\\
X'_1 &= \{ab, ac, aa, dd\},\\
X'_2 &= \{abc, aab, abb, aac, acc, aaa, ddd\}.
\end{aligned}$$

The operators follow easily, e.g. $d'_0(abc) = dd$.

To extend simplicial homology from Section 4.3 to simplicial sets, we just need a chain complex. Let X be a simplicial set. The *nth chain group* $C_n(X)$ of X is the free Abelian group on K's set of oriented, non-degenerate, n-simplices. The *boundary homomorphism* $\partial_n \colon C_n \to C_{n-1}$ is the linear extension of

$$\partial_n = \sum_{i=0}^{n} (-1)^i d_i,$$

where d_i are the face operators and a degenerate face is treated as 0. The boundary homomorphism connects the chain groups into a chain complex, and homology follows.

Example 6.5 (collapsed boundary). The face operators for our collapsed set in Example 6.4 give us the correct boundary. For instance, we have $d_0(abc) = dd$, $d_1(abc) = ac$, and $d_2(abc) = ab$, giving us $\partial_2(abc) = -ac + ab$, as dd is degenerate. Taking another boundary, we have

$$\partial_1\partial_2(abc) = \partial_1(-ac) + \partial_1(ab) = -(d-a) + (d-a) = 0.$$

After the collapse, the simplex abc represents the 2-gon, and its boundary is still a 1-cycle.

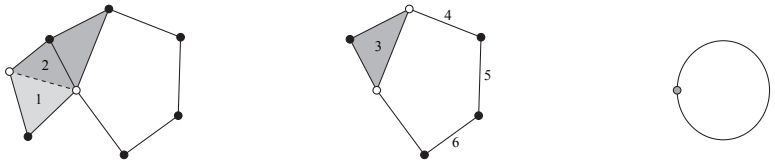
We may now collapse simplices in a simplicial complex to get a smaller simplicial set to represent the unknown space of our point set. In practice, however, computing homology of geometric complex with the persistence algorithm exhibits linear time behavior [**77**]. For the cyclooctane dataset, constructing the complex of more than 3 million simplices takes 11.12 seconds in Section 3.6, while computing homology takes 13.35 seconds in Section 4.4. For larger complexes, we spend most of the time building and storing the complex, not computing its homology. We need reduction *during* construction, not *after*.

6.3. Tidy Sets. Tidy sets are clique complexes that are reduced during construction. Recall from Sections 3.3 and 3.4 that the VR and witness complexes are both clique complexes and are popular in topological analysis. Clique complexes, therefore, present an excellent model for reduction using simplicial sets.

Generally, we construct skeletons of clique complexes using *bottom-up* algorithms, as we did for the cyclooctane dataset in Section 3.6. Alternatively, we may compute the maximal cliques directly as they become maximal simplices in the clique complex [**79**]. Maximal simplices are a minimal description for a simplicial complex as their closure under the subset operation enumerates the full complex. We would need the full description for computing homology, but we may also reduce the complex via *top-down* reduction first.

Let Q and C be disjoint sets of maximal sets, and $\mathcal{X}(Q, C)$ be the simplicial set having the sets in Q as maximal simplices and the sets in C as collapsed maximal simplices. We use the tuple (Q, C) to denote $\mathcal{X}(Q, C)$. Initially, a clique complex is the simplicial set $K = \mathcal{X}(Q, \emptyset)$. A simplicial set X is *acyclic* if $H_0(X) \cong \mathbb{Z}$ and $H_n(X) \cong \{0\}$ for $n > 0$. Contractible spaces, such as simplices in a simplicial complex, are acyclic.

Given a clique complex represented as a tuple (Q, C), we perform two types of reductions that we will describe informally. A *leaf* is a simplex in a simplicial complex that has an acyclic intersection with the rest of the complex, the intersection being its "stem". The notion of a leaf may also be extended naturally to simplicial



(a) Cliques (b) Trimmed (c) Tidy Set

FIGURE 22. Starting with the cliques (a), we trim cliques 1 and 2, and thin cliques 3 through 6 (b) to get the tidy set (c), which is homotopy equivalent to the clique complex (a).

sets. Our first reduction is removing leaves, which preserves homology while keeping a simplicial complex in its category. For cubical complexes, removing leaves is called *shaving* for full-dimensional cubes [**62**]. Our second reduction is collapsing acyclic simplices, which also preserves homology, but may change the category to a simplicial set. Given tuple (Q, C) with $\sigma \in Q$, we have two reductions that we may perform easily.

$$\begin{array}{lll} \textit{trim:} & (Q,C) \mapsto (Q - \{\sigma\}, C) & \sigma, \text{ a leaf} \\ \textit{thin:} & (Q,C) \mapsto (Q - \{\sigma\}, C \cup \{\sigma\}) & \sigma, \text{ acyclic} \end{array}$$

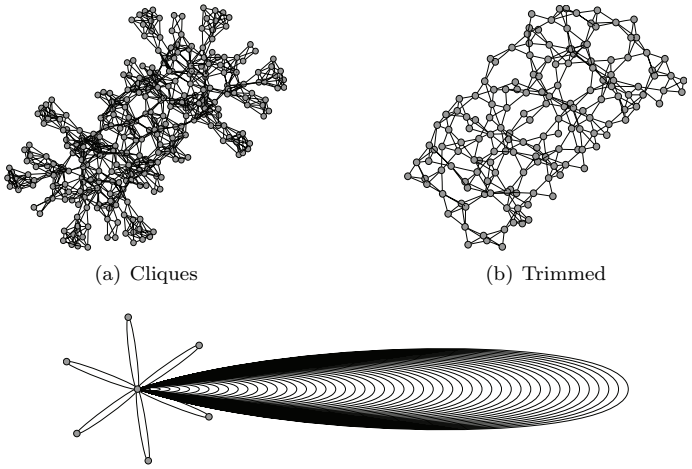
Thinning is closely related to the construction of acyclic subspaces for homology computation [**58**]. A *tidy set* is a trimmed, then thinned, simplicial complex [**80**]. A tidy set is minimal with respect to trimming and thinning.

EXAMPLE 6.6. Figure 22 illustrates the construction of the tidy set for a small complex. We start with the set of maximal cliques (a), rendered as maximal simplices. The intersection of clique 1 with the rest of the cliques is the dashed edge with white vertices. Since this intersection is acyclic, clique 1 is a leaf and is removed. Clique 2 is similarly trimmed. Clique 3 (b), however, intersects the remaining cliques in the two white vertices, so it is not a leaf and cannot be trimmed. Instead, we thin cliques 3 through 6 in order to get the tidy set (c), a loop with one vertex and one edge. The tidy set has the homotopy type of the complex in (a). Compare with elementary contractions in Figure 19.

EXAMPLE 6.7. Figure 23 shows projections of the 1-skeletons of three homologous structures: A clique complex defined by 331 maximal cliques (a), its trimmed complex (b) with 88 remaining cliques, and its tidy set with 23 uncollapsed cliques (c).

We have algorithms for computing tidy sets, based on greedy trimming, and thinning in both simplicial complex and set categories [**80**]. Testing whether a simplex is acyclic or a leaf involves homology, and much of the algorithmic design involves postponing homology computation as long as it is possible.

6.4. Tidy Analysis. Having described tidy sets as an alternative to our two-step pipeline, we now analyze our cyclooctane dataset $\mathcal{S}$ once again. We begin with the neighborhood graph built in Section 3.6. Recall that the maximum scale is $\hat{\epsilon} = 0.4$ and the graph has 76,657 edges.



(a) Cliques (b) Trimmed

(c) Tidy Set

FIGURE 23. A clique complex (a) is trimmed (b) and thinned (c), resulting in its tidy set. We project 1-skeletons only.

We enumerate the maximal cliques in the neighborhood graph in 1.17 seconds using the IK-GX algorithm [**13**]. There are 23,279 maximal cliques, with the average size of a clique being about 8.68 and the maximum being 16, implying that the full VR complex is 15-dimensional.

Since we only built a 4-skeleton in Section 3.6, we first construct the 4-skeleton of the tidy set for comparison. From the maximal cliques, the tidy set algorithm trims 20,246 (87%) and collapses another 1,860 (8%). The remaining 1,173 cliques (5%) give rise to a 4-dimensional tidy set with 155,202 simplices, as compared to the 4-dimensional VR complex with more than 3 million simplices constructed earlier. The entire construction process, including clique enumeration, is 11.73 seconds. In another 0.36 seconds, we compute homology using the persistence algorithm. We find $\beta_0 = 1$, $\beta_1 = 1$, $\beta_2 = 2$, matching our earlier analysis results in Sections 4.3 and Section 5.5. But the tidy set is about 5% of the size of the VR complex, and our total analysis time drops from 24.47 to 12.09 seconds.

The reduction in size motivates us to construct the full tidy set instead of a low-dimensional skeleton. The largest uncollapsed clique in the tidy set has size 12, and we construct the full 11-dimensional tidy set in 11.99 seconds with 202,406 simplices. That is, we only require 0.26 seconds to construct an additional 47,204 simplices. By comparison, the 15-dimensional VR complex has more than 13 million simplices, requiring a computer with large memory for its construction, and even larger memory for its homology computation. Since our tidy set is 66 times smaller, we compute homology groups in all 11 dimensions in only another 0.52 seconds. We find $\beta_0 = 1$, $\beta_1 = 1$, $\beta_2 = 2$, as before, but also $\beta_n = 0$ for all $3 \leq n \leq 11$. The triviality of homology in all higher dimensions is a strong indication that the cyclooctane dataset has intrinsic dimension 2, which indeed is the case [**50**]. Although our only assumption is that the unknown space is topological, our analysis is yielding information about the intrinsic dimension of the dataset.

Due to the size of simplicial complexes, topological analysis has been limited to low dimensional features, such as components [**17**], tunnels [**8**], and voids [**43**]. The tidy set is the first method that enables topological analysis in higher dimensions. On the other hand, this method does not yield filtrations, so we cannot analyze tidy sets at multiple scales using persistent homology. There are several approaches, however, for multiscale analysis using tidy sets, such as zigzag persistence.

7. Conclusion

With its focus on qualitative information, topological data analysis is the first step toward a robust understanding of data. In this chapter, we looked at current multiscale structures and invariants for computing the topology of data. Swift advances in technology allow us to acquire high-resolution data, transmit it through fast networks, and store it on distributed cloud infrastructure. We are engulfed in heterogeneous scientific data without theory or algorithms for its analysis. To extract information from these massive datasets, we need multiscale, nonmonotonic, probabilistic models to represent their structure, as well as randomized and streaming algorithms for their analysis.

Acknowledgments

The author thanks Shawn Martin for generously providing the cyclooctane dataset $\mathcal{S}$ as well as its reconstructed surface.

References

[1] F. Aurenhammer, *Power diagrams: Properties, algorithms and applications*, SIAM Journal on Computing **16** (1987), 78–96.
[2] J.-D. Boissonnat, O. Devillers, and S. Hornus, *Incremental construction of the Delaunay triangulation and the Delaunay graph in medium dimension*, Proc. ACM Symposium on Computational Geometry, 2009, pp. 208–216.
[3] R. Bott and L. W. Tu, *Differential forms in algebraic topology*, Springer-Verlag, New York, NY, 1982.
[4] W. M. Brown, S. Martin, S. N. Pollack, E. A. Coutsias, and J.-P. Watson, *Algorithmic dimensionality reduction for molecular structure analysis*, Journal of Chemical Physics **129** (2008), no. 064118.
[5] G. Carlsson, *Topology and data*, Bulletin of the American Mathematical Society (New Series) **46** (2009), no. 2, 255–308.
[6] G. Carlsson and V. de Silva, *Zigzag persistence*, Foundations of Computational Mathematics **10** (2010), 367–405.
[7] G. Carlsson, V. de Silva, and D. Morozov, *Zigzag persistent homology and real-valued functions*, Proc. ACM Symposium on Computational Geometry, 2009, pp. 247–256.
[8] G. Carlsson, T. Ishkhanov, V. de Silva, and A. Zomorodian, *On the local behavior of spaces of natural images*, International Journal of Computer Vision **76** (2008), no. 1, 1–12.
[9] G. Carlsson, G. Singh, and A. Zomorodian, *Computing multidimensional persistence*, Journal of Computational Geometry **1** (2010), no. 1, 72–100.
[10] G. Carlsson and A. Zomorodian, *The theory of multidimensional persistence*, Discrete & Computational Geometry **42** (2009), no. 1, 71–93.
[11] G. Carlsson, A. Zomorodian, A. Collins, and L. J. Guibas, *Persistence barcodes for shapes*, International Journal of Shape Modeling **11** (2005), no. 2, 149–187.
[12] F. Cazals, J. Giesen, M. Pauly, and A. Zomorodian, *The conformal alpha shape filtration*, The Visual Computer **22** (2006), no. 8, 531–540.
[13] F. Cazals and C. Karande, *Reporting maximal cliques: new insights into an old problem*, Research Report 5642, INRIA, 2005.
[14] CGAL, *Computational Geometry Algorithms Library*, `http://www.cgal.org`.
[15] CHOMP, *Computational Homology Project*, 2011, `http://chomp.rutgers.edu/`.

[16] Y. Civan and E. Yalçin, *Linear colorings of simplicial complexes and collapsing*, Journal of Combinatorial Theory Series A **114** (2007), no. 7, 1315–1331.
[17] A. Collins, A. Zomorodian, G. Carlsson, and L. Guibas, *A barcode shape descriptor for curve point cloud data*, Computers & Graphics **28** (2004), no. 6, 881–894.
[18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*, third ed., The MIT Press, Cambridge, MA, 2009.
[19] D. A. Cox, J. Little, and D. O'Shea, *Using algebraic geometry*, second ed., Graduate Texts in Mathematics, vol. 185, Springer-Verlag, New York, 2005.
[20] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational geometry: Algorithms and applications*, third ed., Springer-Verlag, New York, 2008.
[21] V. de Silva and G. Carlsson, *Topological estimation using witness complexes*, Proc. IEEE/Eurographics Symposium on Point-Based Graphics, 2004, pp. 157–166.
[22] H. Derksen and J. Weyman, *Quiver representations*, Notices of the American Mathematical Society **52** (2005), no. 2, 200–206.
[23] T. K. Dey, *Curve and surface reconstruction*, Cambridge Monographs on Applied and Computational Mathematics, vol. 23, Cambridge University Press, New York, NY, 2007.
[24] D. L. Donoho and C. Grimes, *Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data*, Proc Natl Acad Sci USA **100** (2003), no. 10, 5591–5596.
[25] J.-G. Dumas, F. Heckenbach, B. D. Saunders, and V. Welker, *Computing simplicial homology based on efficient Smith normal form algorithms*, Algebra, Geometry, and Software Systems, 2003, pp. 177–207.
[26] D. Dummit and R. Foote, *Abstract algebra*, third ed., John Wiley & Sons, Inc., New York, NY, 2004.
[27] H. Edelsbrunner, D. G. Kirkpatrick, and R. Seidel, *On the shape of a set of points in the plane*, IEEE Transactions on Information Theory **29** (1983), 551–559.
[28] H. Edelsbrunner and E. P. Mücke, *Three-dimensional alpha shapes*, ACM Transactions on Graphics **13** (1994), 43–72.
[29] H. Edelsbrunner and N. R. Shah, *Incremental topological flipping works for regular triangulations*, Proc. ACM Symposium on Computational Geometry, 1992, pp. 43–52.
[30] H. Edelsbrunner and A. Zomorodian, *Computing linking numbers in a filtration*, Homology, Homotopy and Applications **5** (2003), no. 2, 19–37.
[31] First *(Floppy Inclusions and Rigid Substructure Topography)*, `http://flexweb.asu.edu/software/`.
[32] *Folding@Home: Distributed Computing*, `http://folding.stanford.edu/`.
[33] P. Gabriel, *Unzerlegbare Darstellungen I*, manuscripta mathematica **6** (1972), no. 1, 71–103, (German).
[34] P. Gabriel and A. V. Roiter, *Representations of finite-dimensional algebras*, Springer-Verlag, New York, NY, 1997.
[35] R. Ghrist, *Barcodes: the persistent topology of data*, Bulletin of the American Mathematical Society (New Series) **45** (2008), no. 1, 61–75.
[36] R. Ghrist and A. Muhammad, *Coverage and hole-detection in sensor networks via homology*, Proc. International Symposium on Information Processing in Sensor Networks, 2005.
[37] M. Gromov, *Hyperbolic groups*, Essays in Group Theory (S. Gersten, ed.), Springer-Verlag, New York, NY, 1987, pp. 75–263.
[38] A. Hatcher, *Algebraic topology*, Cambridge University Press, New York, NY, 2002, `http://www.math.cornell.edu/~hatcher/AT/ATpage.html`.
[39] D. J. Jacobs, A. J. Rader, L. A. Kuhn, and M. F. Thorpe, *Protein flexibility prediction using graph theory*, Proteins: Structure, Function, and Genetics **44** (2001), 150–165.
[40] I. T. Jolliffe, *Principle component analysis*, second ed., Springer-Verlag, New York, NY, 2002.
[41] V. G. Kac, *Infinite root systems, representations of graphs and invariant theory*, Inventiones Mathematicae **56** (1980), no. 1, 57–92.
[42] T. Kaczynski, K. Mischaikow, and M. Mrozek, *Computational homology*, Springer-Verlag, New York, NY, 2004.
[43] P. M. Kasson, A. Zomorodian, S. Park, N. Singhal, L. J. Guibas, and V. S. Pande, *Persistent voids: a new structural metric for membrane fusion*, Bioinformatics **23** (2007), no. 14, 1753–1759.
[44] N. Katoh and S-i. Tanigawa, *A proof of the molecular conjecture*, Proc. ACM Symposium on Computational Geometry, 2009, pp. 296–305.

[45] F. Klein, *A comparative review of recent researches in geometry*, Bull. New York Math. Soc. **2** (1892–1893), 215–249, Translated by M. W. Haskell.
[46] D. Kozlov, *Combinatorial algebraic topology*, Springer-Verlag, New York, NY, 2008.
[47] T. Mamonova, B. Hespenheide, R. Straub, M. F. Thorpe, and M. Kurnikova, *Protein flexibility using constraints from molecular dynamics simulations*, Physical Biology **2** (2005), S137–S147.
[48] A. A. Markov, *Insolubility of the problem of homeomorphy*, Proc. International Congress of Mathematics, 1958, pp. 14–21.
[49] S. Martin, A. Thompson, E. A. Coutsias, and J.-P. Watson, *Topology of cyclo-octane energy landscape*, Journal of Chemical Physics **132** (2010), no. 234115.
[50] S. Martin and J.-P. Watson, *Non-manifold surface reconstruction from high-dimensional point cloud data*, Computational Geometry: Theory & Applications **44** (2011), no. 8, 427–441.
[51] J. Matoušek, *LC reductions yield isomorphic simplicial complexes*, Contributions to Discrete Mathematics **3** (2008), no. 2, 37–39.
[52] Y. Matsumoto, *An introduction to Morse theory*, Iwanami Series in Modern Mathematics, vol. 208, American Mathematical Society, Providence, RI, 2002.
[53] J. P. May, *Simplicial objects in algebraic topology*, D. Van Nostrand Co., Inc., Princeton, NJ, 1967.
[54] E. W. Mayr, *Some complexity results for polynomial ideals*, Journal of Complexity **13** (1997), no. 3, 303–325.
[55] N. Milosavljević, D. Morozov, and P. Skraba, *Zigzag persistent homology in matrix multiplication time*, Proc. ACM Symposium on Computational Geometry, 2011, pp. 216–225.
[56] D. Morozov, *Dionysus*, `http://www.mrzv.org/software/dionysus/`.
[57] D. M. Mount and S. Arya, Ann*: A library for approximate nearest neighbor searching, version 1.1.1*, `http://www.cs.umd.edu/~mount/ANN/`.
[58] M. Mrozek, P. Pilarczyk, and N. Żelazna, *Homology algorithm based on acyclic subspace*, Computers and Mathematics with Applications **55** (2008), no. 11, 2395–2412.
[59] National Library of Medicine, *The Visible Human Project*, `http://www.nlm.nih.gov/research/visible/visible_human.html/`.
[60] P. Niyogi, S. Smale, and S. Weinberger, *Finding the homology of submanifolds with high confidence from random samples*, Discrete & Computational Geometry **39** (2008), no. 1, 419–441.
[61] NOAA Satellite and Information Service, *National Geophysical Data Center*, `http://www.ngdc.noaa.gov/`.
[62] P. Pilarczyk, *Computer assisted method for proving existence of periodic orbits*, Topological Methods in Nonlinear Analysis **13** (1999), 365–377.
[63] *RedHom*, `http://redhom.ii.uj.edu.pl/`.
[64] J. J. Rotman, *An introduction to algebraic topology*, Springer-Verlag, New York, NY, 1988.
[65] S. Roweis and L. K. Saul, *Nonlinear dimensionality reduction by locally linear embedding*, Science **290** (2000), no. 5500, 2323–2326.
[66] H. Sexton and M. V. Johansson, *JPlex*, `http://comptop.stanford.edu/programs/jplex/`.
[67] *The Stanford 3D Scanning Repository*, `http://www-graphics.stanford.edu/data/3Dscanrep/`.
[68] A. Storjohann, *Near optimal algorithms for computing Smith normal forms of integer matrices*, Proc. International Conference on Symbolic and Algebraic Computation, 1996, pp. 267–274.
[69] ———, *Computing Hermite and Smith normal forms of triangular integer matrices*, Linear Algebra and Its Applications **282** (1998), no. 1–3, 25–45.
[70] T. S. Tay and W. Whiteley, *Recent advances in the generic rigidity of structures*, Structural Topology **9** (1984), 31–38.
[71] G. Turk and M. Levoy, *Zippered polygon meshes from range images*, Proc. SIGGRAPH, 1994, pp. 311–318.
[72] F. Uhlig, *Transform linear algebra*, Prentice Hall, Upper Saddle River, NJ, 2002.
[73] R. Vidal, Y. Ma, and S. Sastry, *Generalized principal component analysis*, IEEE Trans. Pattern Anal. Mach. Intell. **27** (2005), no. 12, 1945–1959.
[74] L. Vietoris, *Über den höheren zusammenhang kompakter Räume und eine Klasse von zusammenhangstreuen Abbildungen*, Mathematische Annalen **97** (1927), no. 1, 454–472.

[75] J. H. C. Whitehead, *Simplicial spaces, nuclei, and m-groups*, Proceedings of the London Mathematical Society **s2-45** (1939), no. 1, 243–327.
[76] C. K. Yap, *Robust geometric computation*, Handbook of Discrete and Computational Geometry (J. E. Goodman and J. O'Rourke, eds.), CRC Press, LLC, Boca Raton, FL, second ed., 2004, pp. 927–952.
[77] A. Zomorodian, *Topology for computing*, paperback ed., Cambridge University Press, New York, NY, 2009.
[78] ———, *Computational topology*, Algorithms and Theory of Computation Handbook (M. Atallah and M. Blanton, eds.), vol. 2, Chapman & Hall/CRC Press, Boca Raton, FL, second ed., 2010.
[79] ———, *Fast construction of the Vietoris-Rips complex*, Computers & Graphics **34** (2010), no. 3, 263 – 271.
[80] ———, *The tidy set: A minimal simplicial set for computing homology of clique complexes*, Proc. ACM Symposium of Computational Geometry, 2010, pp. 257–266.
[81] A. Zomorodian and G. Carlsson, *Computing persistent homology*, Discrete & Computational Geometry **33** (2005), no. 2, 249–274.
[82] ———, *Localized homology*, Computational Geometry: Theory & Applications **41** (2008), no. 3, 126–148.

The D. E. Shaw Group, New York, NY