Sampling real algebraic varieties for topological data analysis

Emilie Dufresne[*] Parker B. Edwards[†] Heather A. Harrington[‡]
Jonathan D. Hauenstein[§]

October 18, 2018

Abstract

Topological data analysis (TDA) provides a growing body of tools for computing geometric and topological information about spaces from a finite sample of points. We present a new adaptive algorithm for finding provably dense samples of points on real algebraic varieties given a set of defining polynomials. The algorithm utilizes methods from numerical algebraic geometry to give formal guarantees about the density of the sampling and it also employs geometric heuristics to reduce the size of the sample. As TDA methods consume significant computational resources that scale poorly in the number of sample points, our sampling minimization makes applying TDA methods more feasible. We provide a software package that implements the algorithm and also demonstrate the implementation with several examples.

Keywords. Topological data analysis, real algebraic varieties, dense samples, numerical algebraic geometry, minimal distance

1 Introduction

Understanding the geometry and topology of real algebraic varieties is a ubiquitous and challenging problem in applications modelled by polynomial systems. For kinematics problems, geometric insight about configuration spaces can lead to physical insight about the system being modelled (e.g., [37]), while the geometry of varieties can encode information about the dynamics of biochemical systems (e.g., [32]). In this paper, we present a new algorithm fulfilling a key step in applying topological data analysis methods (TDA), particularly persistent homology [53], to real algebraic varieties. We aim to provide and demonstrate a computationally feasible pipeline for applying TDA to real algebraic varieties while maintaining theoretical guarantees.

The most closely related problem to computing persistent homology (PH) of real algebraic varieties is the computation of its Betti numbers. There are two main approaches to this problem: symbolic methods which process polynomial equations directly, and surface reconstruction methods which estimate Betti numbers by constructing spaces from point samples.

[*] School of Mathematical Sciences, University of Nottingham, University Park, Nottingham, NG7 2RD (emilie.dufresne@nottingham.ac.uk).

[†] Department of Mathematics, PO Box 118105, University of Florida, Gainesville, FL 32611 (pedwards@ufl.edu)

[‡] Mathematical Institute, Radcliffe Observatory Quarter, Woodstock Road, Oxford OX2 6GG, United Kingdom (harrington@maths.ox.ac.uk)

[§] Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN 46556 (hauenstein@nd.edu, www.nd.edu/~jhauenst).

The complexity of symbolically computing Betti numbers of (complex) projective varieties has been studied [48] as well as numerically stable homology computations of real projective varieties [22] and real semialgebraic sets [4]. Using random sampling of manifolds, an algorithm is provided in [44] for "learning" the homology with complexity bounds in terms of a condition number relating to the curvature and closeness to self-intersections. Alternatively, one can intersect a projective variety with a general linear space and obtain points. For a large enough set of "general" points obtained this way, [43] studies the Betti diagram, which provides information about the projective variety. Although an algorithm for obtaining such general points is not given in that paper, the recent paper [12] provides a uniform sampling method for algebraic manifolds using slicing. For a given set of general points, recent work has "learned" the equations defining the algebraic variety and confirmed the expected homology via persistent homology computations [13].

Extensive effort has also produced a large number of surface reconstruction algorithms, particularly for nonsingular surfaces embedded in $\mathbb{R}^3$. The survey [10] and articles [2, 24, 29, 38] provide a representative, though by no means exhaustive, list of examples. The general format of surface reconstruction algorithms is to take as input a point cloud sampled from an underlying surface or space, and output a simplicial complex (or richer structure) which geometrically estimates the underlying space. Betti numbers computed from the simplicial complex serve as estimates of the numbers for the underlying space. The inherent limitations on the underlying space required by these methods exclude their use as general tools for analyzing real algebraic varieties. Indeed, all of the methods listed above either provide no theoretical guarantees on the correctness of the reconstruction, or require that the underlying space is some combination of nonsingular, embedded in $\mathbb{R}^3$, and/or intrinsically 2-dimensional.

We focus on using the persistent homology pipeline to analyze real varieties. Like surface reconstruction methods, the pipeline accepts as input a finite set of points sampled from a space and outputs estimated homological information about the space. More precisely, suppose that we are given defining polynomials for a pure dimensional algebraic set $V \subseteq \mathbb{C}^N$. The compact set $V_\mathbb{R}$ resulting from intersecting V with a hypercube in $\mathbb{R}^N$ is the one we wish to sample and analyze. PH captures richer homological information than just Betti numbers, and the theoretical guarantees of PH apply to potentially singular varieties embedded in any dimension. As a trade off, the computational resources required to compute PH quickly become large when more points are added to the sample (e.g., [45]). Both the theoretical framework for the PH pipeline and its computational costs drive the requirements of a suitable sampling algorithm. Among existing sampling approaches, subdivision and reduction sampling methods [42, 50] are the most obvious candidates. In their most general format, these methods can take the polynomials defining a real semialgebraic set as input and output a dense sample of points. For PH computations, they exhibit two drawbacks:

1. Sample points in the output need not be especially close to the underlying variety. Input samples with points close to the underlying variety significantly improve the accuracy of PH results.

2. Adjusting current implementations to reduce the number of sample points in the output is not straightforward. Computational resource requirements for PH scale up quickly with more sample points.

Our alternative approach for sampling varieties is based on numerical algebraic geometry, with the books [5, 51] providing a general overview. The algorithm addresses the first point above by constructing provably dense samples with points very close to the underlying variety. The theoretical version of the algorithm can be readily adjusted to incorporate geometric heuristics which significantly reduce the number of points in the final output thereby addressing the second point.

The remainder of the paper is organized as follows: In Section 2, we recall the TDA theory and computational considerations which determine our sampling algorithm's requirements. In Section 3, we explain the tools from numerical algebraic geometry used in our approach. Section 4 details the sampling algorithm, proves its correctness, and discusses the geometric heuristics for sample minimization. Finally, we present examples in Section 5 to illustrate how our sampling algorithm can be used in conjunction with TDA to calculate topological information for several varieties.

2 Topological data analysis

Topological data analysis is a very active field of research broadly encompassing theory and algorithms which adapt the theoretical tools of topology and geometry to analyze the "shape" of data. We concentrate on applying the "persistent homology pipeline" popularized by Carlsson in [16] and summarized by Ghrist in [31]. Broader overviews of other TDA methods can be found in the articles [19, 45] and textbooks [26, 46]. The PH pipeline follows these steps:

1. Input data is expected to be in the form of a *point cloud* consisting of finitely many points in $\mathbb{R}^n$ together with their pairwise distances.

2. A collection of shapes, simplicial complexes, are constructed out of the input data. The complexes encode the shape of the data at different distance scales.

3. Algebraic topological features of the simplicial complexes produced in Step 2 are calculated, compared, and ultimately assembled into a single output summary using the algebraic theory of *persistent homology*.

In this section, we briefly recall simplicial complexes and the basics of homology (the textbook [33] provides detailed information on homology theory). We then summarize the theoretical and computational elements in each step of the PH pipeline that pertain to applying the pipeline to real algebraic varieties.

2.1 Homology groups

Algebraic topology studies methods for assigning algebraic structures to topological spaces in such a way that the algebraic structures encode topological information about the space. For a natural number $p \geqslant 0$, the p-th homology group of a space captures information about the number of p-dimensional holes in the space. We initially restrict focus to spaces called simplicial complexes which are more amenable to computation.

Definition 2.1. An (abstract) *simplicial complex* is a finite set Ω of non-empty subsets of $\mathbb{N}$ such that $\omega \in \Omega$ implies that every subset of ω is an element in Ω. If Ω is an abstract simplicial complex, the elements of the set $\Omega_0 = \cup_{\omega \in \Omega} \omega$ are called the *vertices* of Ω. The *dimension* of Ω is one less than the size of the largest set in Ω. For simplicial complexes Ω and Ω', a simplicial map from Ω to Ω' is a map $f : \Omega_0 \to \Omega'_0$ where $\omega \in \Omega$ implies $f(\omega) \in \Omega'$.

This purely combinatorial definition corresponds geometrically to forming spaces by gluing together points, lines, triangles, tetrahedra, and higher dimensional equivalents. Note that a simplicial complex Ω defines a subspace $|\Omega|$ of some Euclidean space ($|\Omega|$ is a *geometric realization* of Ω). The vertices Ω_0 correspond to geometric vertices in $|\Omega|$, the 2-element subsets in Ω to lines, the 3-element subsets to triangles, etc., as shown in Figure 1. The homology groups (with $\mathbb{Z}/2$ coefficients) for the complex Ω are built in 3 steps.

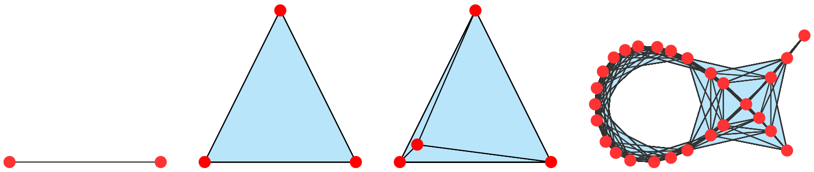


Figure 1: From left to right: Geometric realizations of a 1-simplex, 2-simplex, 3-simplex (the interior of the tetrahedron is included), and a general simplicial complex.

First, encode the complex Ω algebraically.

Definition 2.2. Take $\Omega_p \subseteq \Omega$ for an integer $p \geqslant 0$ to contain the sets in Ω with size $p+1$. Members of Ω_p are *p-simplices* of Ω. The p-th chain group of Ω, denoted $C_p(\Omega)$, is the group of formal sums $\sum_{k=1}^{n} b_k\omega_k$ where $\omega_k \in \Omega_p$ and $b_k \in \mathbb{Z}/2$ for all k. Set $C_{-1} = 0$.

Next, define a simplicial analog to the geometric operation of taking the boundary of a space.

Definition 2.3. The p-th boundary operator is a homomorphism $\partial_p : C_p(\Omega) \to C_{p-1}(\Omega)$ for each $p \geqslant 0$. Define ∂_0 to be the zero map $\partial_0 : C_0(\Omega) \to 0$. For $p > 0$ and any basis element $\omega \in C_p(\Omega)$, define $\partial_p(\omega) = \sum_{\{\omega' \in \Omega_{p-1} | \omega' \subseteq \omega\}} \omega'$. Extending the function linearly from its action on the basis elements of $C_p(\Omega)$ defines ∂_p on the entirety of $C_p(\Omega)$. Elements in the kernel of ∂_p are called *cycles* and the kernel is denoted $\ker(\partial_p) = Z_p(\Omega)$. Elements in the image of ∂_{p+1} are called *boundaries* and this group is denoted $\mathrm{Im}(\partial_{p+1}) = B_p(\Omega)$.

Finally, capture algebraically, for all dimensions, the geometric intuition that a 1-dimensional loop has no boundary, and encloses a void only if it is not the boundary of a 2-dimensional region.

Definition 2.4. It can be shown that $\partial_p \circ \partial_{p+1} = 0$ for all $p \geqslant 0$, so that $B_p \subseteq Z_p$. The *p-th homology group* of Ω, $H_p(\Omega)$, is the quotient group $Z_p(\Omega)/B_p(\Omega)$. $H_p(\Omega)$ is a finite dimensional vector space, and its rank is called the *p-th Betti number* of Ω, $\beta_p(\Omega)$.

The elements of the homology groups informally represent loops and higher dimensional equivalents in a space, with $\beta_p(\Omega)$ counting the number of p-dimensional holes. A basis element of $H_0(\Omega)$ for a complex Ω represents a single connected component of $|\Omega|$, a basis element of $H_1(\Omega)$ represents a set of loops which can all be deformed within the space $|\Omega|$ into a loop which encloses the same 2-dimensional void, and basis elements of $H_2(\Omega)$ account for 3-dimensional voids.

Homology groups behave nicely with respect to simplicial maps, a property called *functoriality*. Consider simplicial complexes Ω and Ω'. Functoriality implies that for any simplicial map $f : \Omega \to \Omega'$ and $p \geqslant 0$ there exists an $\mathbb{Z}/2$-linear map $H_p(f) : H_p(\Omega) \to H_p(\Omega')$. If Ω'' is a third simplicial complex and $g : \Omega' \to \Omega''$ another simplicial map, then

$$H_p(g \circ f) = H_p(g) \circ H_p(f).$$

We can construct homology groups from topological spaces directly without using simplicial complexes. This more general *singular homology* construction (H_p^{sing}) implies the existence of a $\mathbb{Z}/2$-linear map $H_p^{\text{sing}}(X) \to H_p^{\text{sing}}(Y)$ induced by any continuous function $f : X \to Y$. H_p^{sing} also retains nice behavior with respect to composition of continuous functions. Standard results in algebraic topology show that these two different notions of homology agree where they are both defined. We will not distinguish between singular and simplicial homology subsequently.

2.2 Building simplicial complexes from data

Let $\mathcal{X} \subseteq \mathbb{R}^n$ be a finite point cloud serving as input to the PH pipeline. A natural way to estimate shape information from $\mathcal{X}$ is to build a simplicial complex from $\mathcal{X}$ based on the distances between its points.

Definition 2.5. Let $\mathcal{X}$ be a finite subset of a metric space Y and ϵ be a real number. The *Čech complex* for $\mathcal{X}$ with parameter ϵ, $C_\epsilon(\mathcal{X})$, is a simplicial complex such that:

- If $\epsilon < 0$, $C_\epsilon(\mathcal{X})$ is defined directly to be $\varnothing$.
- The vertex set of $C_\epsilon(\mathcal{X})$ is $\mathcal{X}$.
- A set $x \subseteq \mathcal{X}$ belongs to $C_\epsilon(\mathcal{X})$ if there exists a point $y \in Y$ such that distance between y and any point in X is at most ϵ.

The *Vietoris-Rips complex* for $\mathcal{X}$ with parameter ϵ, denoted $R_\epsilon(\mathcal{X})$, is a simplicial complex that fulfills an alternative version of condition 3 above:

* A set $x \subseteq \mathcal{X}$ belongs to $R_\epsilon(\mathcal{X})$ if the distance between any two points in x is at most ϵ.

The Čech complex is closely connected with the geometric operation of "thickening" a finite point cloud $\mathcal{X} \subseteq \mathbb{R}^n$. Given a parameter $\epsilon \geqslant 0$ we can replace the space $\mathcal{X}$ with the union of closed balls of radius ϵ in $\mathbb{R}^n$ centered at points in $\mathcal{X}$. The Nerve Theorem (see e.g. §4G.3 [33]) guarantees that the singular homology of this version of $\mathcal{X}$ which has been thickened by ϵ is isomorphic to the simplicial homology of $C_\epsilon(\mathcal{X})$. Calculating Čech complexes for reasonably sized point clouds presents computational issues, as a large number of intersections of balls around points must be checked. Vietoris-Rips complexes estimate Čech complexes, and can be constructed more easily since only distances between pairs of points must be checked. See §3.2 of [26] for more computational details about constructing these complexes. The following interleaving result precisely describes the manner in which the Vietoris-Rips complexes estimate Čech complexes.

Theorem 2.6 (de Silva and Ghrist [23])**.** *If $\mathcal{X}$ is a finite set of points in $\mathbb{R}^n$ and $\epsilon > 0$ there is a chain of inclusions*

$$C_{\frac{\epsilon'}{2}}(\mathcal{X}) \subseteq R_{\epsilon'}(\mathcal{X}) \subseteq C_\epsilon(\mathcal{X}) \subseteq R_{2\epsilon}(\mathcal{X})$$

whenever $\frac{\epsilon}{\epsilon'} \geqslant \frac{1}{2}\sqrt{\frac{2n}{n+1}}$.

2.3 Persistent homology

Given a point cloud $\mathcal{X}$ sampled evenly from nearby some underlying space $X \subseteq \mathbb{R}^n$ as input, we could ask which parameter ϵ produces homology $H_p(C_\epsilon(\mathcal{X}))$ that most closely matches the homology $H_p(X)$ of the underlying space. TDA methods sidestep this question, and instead consider all of the homology groups $H_p(C_\epsilon(X))$ simultaneously. Persistent homology provides an algebraic framework for tracking homology features as the parameter value changes. We summarize the categorical approach to persistent homology introduced in [15]. The central objects of study in this framework are called *persistence modules*.

Definition 2.7. Let k be a field. A *persistence module* is a functor $F : (\mathbb{R}, \leqslant) \to \mathbf{Vect}_k$ from the poset $(\mathbb{R}, \leqslant)$ to the category $\mathbf{Vect}_k$ consisting of vector spaces over k with linear maps between them. Explicitly, F is determined by:

- A k-vector space $F(\epsilon)$ for every $\epsilon \in \mathbb{R}$
- A linear map $F(\epsilon \leqslant \epsilon') : F(\epsilon) \to F(\epsilon')$ for every pair of real numbers $\epsilon \leqslant \epsilon'$ such that:
  - $F(\epsilon \leqslant \epsilon)$ is the identity map from $F(\epsilon)$ to itself
  - Given real numbers $\epsilon \leqslant \epsilon' \leqslant \epsilon''$, $F(\epsilon \leqslant \epsilon'') = F(\epsilon' \leqslant \epsilon'') \circ F(\epsilon \leqslant \epsilon')$

If F and G are both persistence modules, their direct sum $F \oplus G$ is a persistence module where $(F \oplus G)(\epsilon) = F(\epsilon) \oplus G(\epsilon)$ and similarly $(F \oplus G)(\epsilon \leqslant \epsilon') = F(\epsilon \leqslant \epsilon') \oplus G(\epsilon \leqslant \epsilon')$.
F is (naturally) isomorphic to G, $F \cong G$, if for all real numbers $\epsilon \leqslant \epsilon'$ there exist isomorphisms from $F(\epsilon) \to G(\epsilon)$ and $F(\epsilon') \to G(\epsilon')$ such that the following diagram commutes:

$$\begin{tikzcd}
F(\epsilon) \arrow[r, "F(\epsilon\leqslant\epsilon')"] \arrow[d] & F(\epsilon') \arrow[d] \\
G(\epsilon) \arrow[r, "G(\epsilon\leqslant\epsilon')"] & G(\epsilon')
\end{tikzcd}$$

Definition 2.8. A point $\epsilon \in \mathbb{R}$ is *regular* for a persistence module F if there exists an interval $I \subseteq \mathbb{R}$ where $\epsilon \in I$ and $F(a \leqslant b)$ is an isomorphism for all pairs $a \leqslant b \in I$. Otherwise ϵ is *critical*. A functor is *tame* if it has finitely many critical values.

Example 2.9. For any finite point cloud $\mathcal{X} \subseteq \mathbb{R}^n$ and real numbers $0 \leqslant \epsilon \leqslant \epsilon'$, it follows directly from the definition that $C_\epsilon(\mathcal{X}) \subseteq C_{\epsilon'}(\mathcal{X})$. Regarding the subset inclusion as an inclusion map and fixing $p \geqslant 0$ results in a sequence of vector spaces and $\frac{\mathbb{Z}}{2}$-linear maps $H_p(C_\epsilon(\mathcal{X})) \hookrightarrow H_p(C_{\epsilon'}(\mathcal{X}))$ from the functoriality of H_p. The assignment $\epsilon \mapsto H_p(C_\epsilon(\mathcal{X}))$ along with these linear maps induced by inclusion defines a tame persistence module $H_pC_\bullet(\mathcal{X})$, which we will denote by HC. An analagous persistence module exists for the Vietoris-Rips complex, which we will denote by VR. $\lhd$

Example 2.10. Let I be an interval of the form $[a, b)$, $(a, b]$, or (a, b) where $a, b \in \bar{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$ and let k be a field. The persistence module χ_I maps $\epsilon \in \mathbb{R}$ to the vector space k if $\epsilon \in I$, and maps ϵ to the trivial vector space 0 otherwise. For any real numbers $\epsilon \leqslant \epsilon'$, define $\chi_I(\epsilon \leqslant \epsilon')$ to be the identity map if both $\epsilon, \epsilon' \in I$, and the trivial map otherwise. $\lhd$

Theorem 2.11 (Fundamental Theorem of Persistent Homology). *Let J be a tame persistence module. Then there is a finite set*[1] *$\mathcal{B}_J$ of intervals on the real line, $\mathcal{B}_J = \{I_1, \ldots, I_l\}$, such that*

$$J \cong \chi_{I_1} \oplus \cdots \oplus \chi_{I_l}$$

[1] More precisely, this set is a *multiset*, which is a set where individual elements can occur more than once.

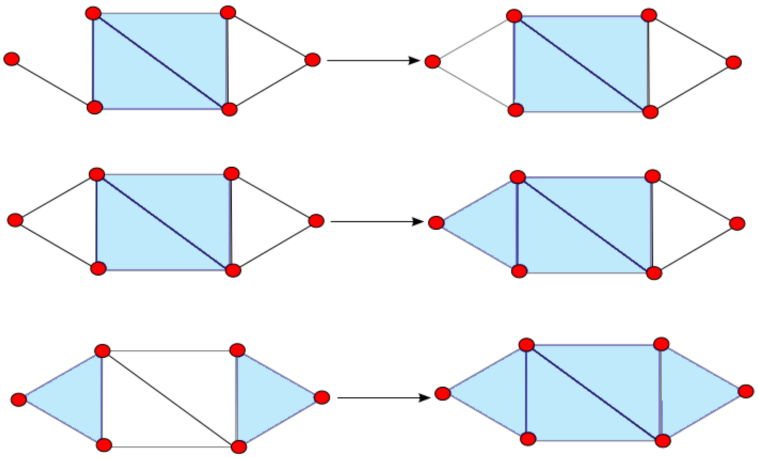


Figure 2: Different events in the life of a homology feature. In the top row a feature is born, in the middle two features merge, and in the bottom a feature dies.

and this decomposition is unique up to reordering of the intervals.

Let J be any tame persistence module and B_J its corresponding set of intervals as in the Fundamental Theorem. The algebraic features encoded in the persistence module J can be visualized (see Figure 3) as a *barcode* or a *persistence diagram*. The set $\mathcal{B}_J$ is called the *barcode* associated to J. The *persistence diagram* of J, denoted DJ, is the set of points $(a, b) \in \bar{\mathbb{R}}^2$ where a is the left endpoint of an interval $I \in \mathcal{B}_J$, b is the right endpoint of I and $a, b \in \bar{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$. Note that the points DJ which have a large straight-line distance to the diagonal correspond to long bars in the barcode. DJ also contains all points of the form (c, c) for all $c \in \bar{\mathbb{R}}$.

The original algebraic version of Theorem 2.11 for persistent homology appears in [53], and a categorical version in [15]. Each interval in the barcode of a tame functor can be viewed as describing the range of parameter values through which a single independent feature in the module persists. For a module like $H_p C_\bullet(\mathcal{X}) = HC$, an interval $[a, b)$ in the barcode corresponds to a p-dimensional void that first appears at parameter value a and is "filled in" by $(p+1)$-dimensional simplices at parameter value b. See Figure 2.

Persistence diagrams for modules arising from the homology of finite simplicial complexes can be computed via the Persistence Algorithm (see e.g. [21] VII.1). Note that intervals in the barcodes for such modules always have the form $[a, b)$ for some $a \leqslant b \in \bar{\mathbb{R}}$. Persistence diagrams therefore contain the same amount of information as barcodes for these modules. In the worst case, the computational complexity for computing the persistence of $H_p R_\bullet(\mathcal{X}) = VR$ scales with the maximum number of $p+1$ simplices in R_ϵ attained at any parameter value ϵ. More precisely: if $\mathcal{X}$ contains m points, calculating the full persistence diagram for VR in the worst case has time complexity $O(\binom{m}{p+2}^\omega)$ where $\omega = 2.376$ is the best known exponent for matrix multiplication [40].

The Persistence Algorithm has been significantly optimized since its original formulation (for instance: [8, 20, 39]). Despite improvement in optimizations and implementations, limiting both the size of the point cloud m and the homology dimension p is often necessary in practice to make persistent homology computations feasible (see [45]). Many applications restrict to computing PH only in dimensions $p \leqslant 2$. Since memory consumption grows rapidly as the number of points m increases, this necessitates keeping the size of point samples as low as possible.

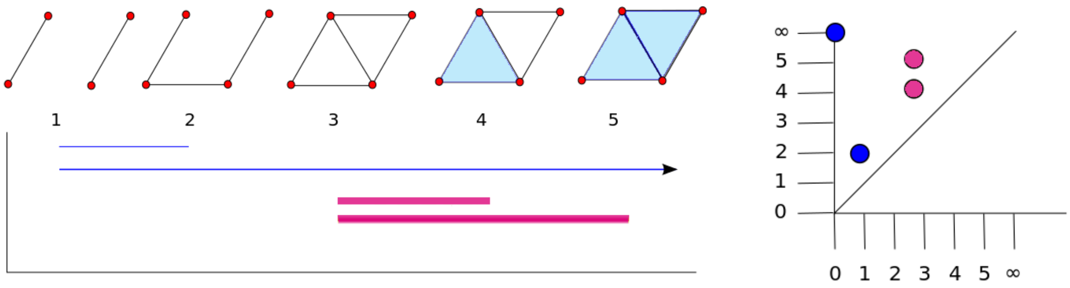


Figure 3: The persistence diagram and barcode of a filtered complex. The left figure shows the complex as it changes with parameter value, along with the corresponding functor's barcode. The right figure depicts the persistence diagram equivalent to the barcode. Blue bars and points represent 0 dimensional homology, whereas pink bars and points represent 1 dimensional homology. An arrow on a bar indicates that the homology feature corresponding to the bar "lives forever"- the corresponding interval is of the form $[a, \infty)$.

2.4 Homology inference

Suppose that $\mathcal{X} \subseteq \mathbb{R}^n$ is a finite point cloud sampled from nearby the compact topological space $X \subseteq \mathbb{R}^n$. A key property of persistent homology, first observed and proven in [21], is that persistent homology computed from $\mathcal{X}$ recovers the homology of the X provided X is a "dense enough" sample. To make this notion precise, recall that any compact topological space such as X defines the distance-to-X function $d_X : \mathbb{R}^n \to \mathbb{R}$. The function is given by $d_X(y) = \min_{x \in X} d(x, y)$ for any $y \in \mathbb{R}^n$. Given any real number $\epsilon \geqslant 0$, define $X^\epsilon = d_X^{-1}(-\infty, \epsilon]$. The space X^ϵ is formed from X by taking the union of all closed balls of radius ϵ in $\mathbb{R}^n$ centered at points of X.

Definition 2.12. Let $A, B \subseteq \mathbb{R}^n$ be compact and $0 \leqslant \delta \leqslant \epsilon \in \mathbb{R}$. The set A is a (δ, ϵ)-*sample* of B if $A \subseteq B^\delta$ and $B \subseteq A^\epsilon$.

Remark 2.13. Definition 2.12 is a specific instance of an *interleaving* between generalized persistence modules as defined in [14]. It is also generalization of the Hausdorff distance between subsets of metric space. Given compact $A, B \subseteq \mathbb{R}^n$, the smallest ϵ such that A and B are (ϵ, ϵ)-samples of one another is the Hausdorff distance between A and B.

Definition 2.14. Let $X \subseteq \mathbb{R}^n$ be a compact metric space. The homological feature size of X, $\text{hfs}(X)$, is the smallest critical value over all dimensions k of the persistence module $\epsilon \mapsto H_k(X^\epsilon)$ (negative ϵ are assigned $\varnothing$). Equivalently $\text{hfs}(X)$ is the smallest number ϵ such that thickening the space X by ϵ changes its homology.

Remark 2.15. Homological feature size of a space was introduced in [21], and is bounded below by the space's *local feature size* [1] and *weak feature size* [17]. It follows from the definition that $\text{hfs}(X) \geqslant 0$ for any space. The weak feature size of real semialgebraic sets is known to be positive ([30] §5.3), and so the homological feature size of real algebraic varieties is positive as well.

Theorem 2.16 (Homology Inference Theorem, [18, 21])**.** *Let $\mathcal{X}, X \subseteq \mathbb{R}^n$, with X compact and $\mathcal{X}$ a finite (δ, ϵ)-sample of X where $0 \leqslant \delta \leqslant \epsilon$ and $\text{hfs}(X) > 2(\epsilon + \delta)$. Letting $HC = H_p C_\bullet(\mathcal{X})$, the dimension of $H_p(X)$ is the number of points in $D(HC)$, the persistence diagram of the functor HC, above and to the left of the point $(\epsilon, 2\epsilon + \delta) \in \bar{\mathbb{R}}^2$.*

Proof. From the definition of (δ, ϵ)-sample we have inclusions $X \hookrightarrow \mathcal{X}^{\epsilon} \hookrightarrow X^{\epsilon+\delta} \hookrightarrow \mathcal{X}^{2\epsilon+\delta} \hookrightarrow X^{2(\epsilon+\delta)}$. The Nerve Theorem implies that $HC(a) \cong H_p(\mathcal{X}^a)$ for all $a \in \mathbb{R}$. Applying homology to the sequence and using the assumption on the homology when thickening X, we obtain the commutative diagram

$$\begin{tikzcd}
H_p(X) \arrow[r] & HC(\epsilon) \arrow[r] \arrow[rr, bend right, "h"'] & H_p(X) \arrow[r] & HC(2\epsilon+\delta) \arrow[r] & H_p(X)
\end{tikzcd}$$

where the maps from $H_p(X)$ to itself are isomorphisms. Consider the map h. Since there is an isomorphism from $H_p(X)$ to itself which factors through h, $\dim H_p(X) \leqslant \operatorname{rank}(h)$. We also have that h factors through a map with domain $H_p(X)$, so that $\operatorname{rank}(h) \leqslant \dim H_p(X)$. Thus $\operatorname{rank}(h) = \dim H_p(X)$. The Theorem follows upon noting that $\operatorname{rank}(h)$ counts the number of intervals in the barcode for HC with left endpoint at most ϵ, and right end point greater than $2\epsilon+\delta$. These intervals correspond to points above and to the left of $(\epsilon, 2\epsilon+\delta)$ in the persistence diagram. □

Corollary 2.17. *Let $HC, X, \mathcal{X}, \epsilon$, and δ be as in Theorem 2.16. Then the number of points above and to the left of $\left(2\epsilon\sqrt{\frac{n+1}{2n}}, 4\epsilon + 2\delta\right)$ in the persistence diagram for $VR = H_pR_\bullet(\mathcal{X})$ is a lower bound for $\dim H_p(X)$ and upper bound for $\operatorname{rank}\left(HC\left(\epsilon\sqrt{\frac{n+1}{2n}} \leqslant (2\epsilon+\delta)\sqrt{\frac{2n}{n+1}}\right)\right)$.*

Proof. Let $a = 2\epsilon\sqrt{\frac{n+1}{2n}}$. By Theorem 2.6, we have the following commutative diagram of linear maps

$$\begin{tikzcd}
VR(a) \arrow[r] \arrow[rrr, bend right, "h"'] & HC(\epsilon) \arrow[r] & HC(2\epsilon+\delta) \arrow[r] & VR(4\epsilon+2\delta).
\end{tikzcd}$$

It follows that $\operatorname{rank}(h) \leqslant \operatorname{rank}(HC(\epsilon \leqslant 2\epsilon+\delta))$ because h factors through $HC(\epsilon \leqslant 2\epsilon+\delta)$. Theorem 2.16 shows that the rank of $HC(\epsilon \leqslant 2\epsilon+\delta)$ is $\dim H_p(X)$. As in the proof of Theorem 2.16, the rank of h is precisely the number of points in the persistence diagram of VR which are above and to the left of $(a, 4\epsilon+2\delta)$. The proof for the upper bound is similar, and uses the sequence of maps $HC(\frac{a}{2}) \to VR(a) \to VR(4\epsilon+2\delta) \to HC((2\epsilon+\delta)\sqrt{\frac{2n}{n+1}})$. □

3 Sampling using numerical algebraic geometry

An algebraic variety $V \subset \mathbb{C}^N$ is the solution set of a system of polynomial equations. The real points of V, $V_\mathbb{R} = V \cap \mathbb{R}^N \subset \mathbb{R}^N$, is a real algebraic variety. One approach to compute a point on $V_\mathbb{R}$ is by computing a point $x \in V_\mathbb{R}$ which is a global minimizer of the distance function between a given test point $y \in \mathbb{R}^N$ and $V_\mathbb{R}$ [49]. We summarize the use of numerical algebraic geometry to perform this computation based on [35] (see also [3, 25, 47]) with Section 4 relying on this to generate a provably dense sampling of $V_\mathbb{R}$.

Suppose that $f_1, \ldots, f_{N-d} \in \mathbb{R}[x_1, \ldots, x_N]$ and let $V \subset \mathbb{C}^N$ be the union of d-dimensional irreducible components of the solution set of $f = \{f_1, \ldots, f_{N-d}\} = 0$. That is, V is a pure d-dimensional algebraic variety with corresponding real algebraic variety $V_\mathbb{R} = V \cap \mathbb{R}^N$. We note that there is no loss of generality since one can utilize randomization if more than $N-d$ polynomials are provided as shown in the following example.

Example 3.1. The affine cone over the twisted cubic curve is the irreducible surface ($d = 2$)

$$V = \{(s^3, s^2t, st^2, t^3) \mid s, t \in \mathbb{C}\} \subset \mathbb{C}^4$$

which is defined by $g_1 = g_2 = g_3 = 0$ where

$$g(x) = \begin{bmatrix} x_2^2 - x_3x_1 \\ x_2x_3 - x_4x_1 \\ x_2x_4 - x_3^2 \end{bmatrix}.$$

Since $N = 4$, we can randomize down to $N - d = 2$ equations, say $f_1 = f_2 = 0$ where

$$f(x) = \begin{bmatrix} x_2^2 - x_3x_1 + 2(x_2x_4 - x_3^2) \\ x_2x_3 - x_4x_1 - 3(x_2x_4 - x_3^2) \end{bmatrix}.$$

In particular, V is one of the two irreducible components of the solution set defined by $f_1 = f_2 = 0$ with the other being the plane defined by $3x_1 + 7x_2 - 4x_4 = x_1 - 7x_3 - 6x_4 = 0$. $\lhd$

Given a test point $y \in \mathbb{R}^N$, the approach of Seidenberg [49] is to compute a global minimizer of

$$\min \left\{ \sum_{i=1}^{N} (x_i - y_i)^2 \;\middle|\; x \in V_{\mathbb{R}} \right\} \tag{3.2}$$

which is accomplished by solving the Fritz John optimality conditions, namely solving

$$G_y(x, \lambda) = \begin{bmatrix} f(x) \\ \lambda_0(x - y) + \sum_{i=1}^{N-d} \lambda_i \nabla f_i(x) \end{bmatrix}$$

on $\mathbb{C}^N \times \mathbb{P}^{N-d}$, where $\nabla f_i(x)$ is the gradient of $f_i(x)$ with respect to x. The polynomial system G_y is a so-called square system consisting of $(N - d) + N$ polynomials on $\mathbb{C}^N \times \mathbb{P}^{N-d}$ so that it is

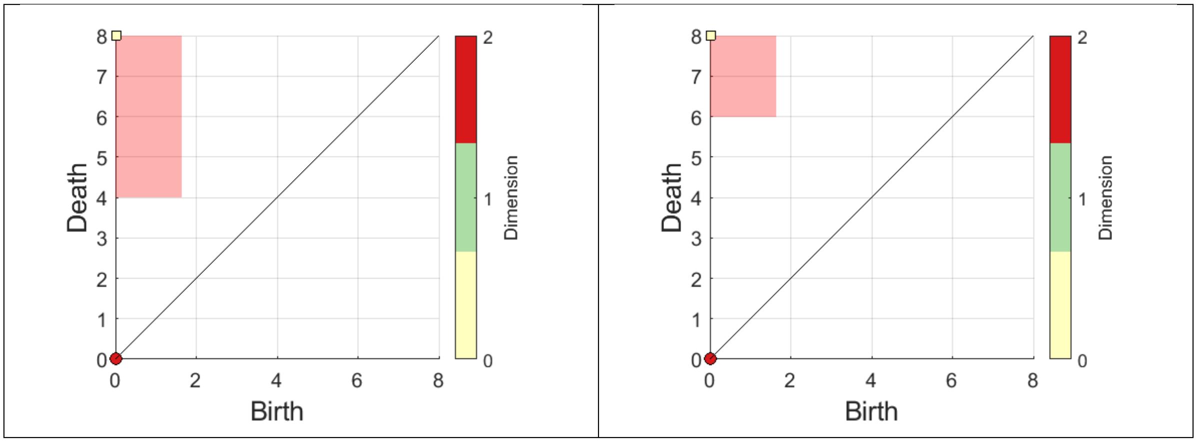


Figure 4: Diagrams displaying the results of Corollary 2.17. The diagram on the left is for a $(0, 1)$-sample of an underlying space with homological feature size at least 2. Any points falling into the pink region in the diagram on the left correspond to features in the underlying space. The diagram on the right is the same, but is for a $(1, 1)$-sample of an underlying space with homological feature size at least 4.

amenable to solving via homotopy continuation. In particular, for $\beta \in \mathbb{C}^{N-d}$, consider

$$H_{y,\beta}(x,\lambda,t) = \left[\begin{array}{c} f(x) - t\beta \\ \lambda_0(x-y) + \sum_{i=1}^{N-d} \lambda_i \nabla f_i(x) \end{array} \right].$$

The following is immediate from coefficient-parameter continuation [41] showing that generic choices of parameter values (y,β) leads to a well-constructed homotopy $H_{y,\beta}$.

Proposition 3.3. *There exists a nonempty Zariski dense open subset $U \subset \mathbb{C}^N \times \mathbb{C}^{N-d}$ such that if $(y,\beta) \in U$, then*

1. *the set $S \subset \mathbb{C}^N \times \mathbb{P}^{N-d}$ consisting of all solutions to $H_{y,\beta}(x,\lambda,1) = 0$ is finite and each is a nonsingular solution;*

2. *the number of points in S is equal to the maximum number, as $y' \in \mathbb{C}^N$ and $\beta' \in \mathbb{C}^{N-d}$ both vary, of isolated solutions of $H_{y',\beta'}(x,\lambda,1) = 0$;*

3. *the solution paths defined by the homotopy $H_{y,\beta}(x,\lambda,t) = 0$ starting at the points in S at $t = 1$ are smooth for $t \in (0,1]$.*

Since $G_y(x,\lambda) = H_{y,\beta}(x,\lambda,0)$, the endpoints of solution paths defined by $H_{y,\beta}(x,\lambda,t) = 0$ contained in $\mathbb{C}^N \times \mathbb{P}^{N-d}$ are solutions of $G_y = 0$. Hence, by tracking the finitely many paths starting at the points in S at $t = 1$, one obtains a finite set of solutions of $G_y = 0$, one of which corresponds with the global minimizer of (3.2) as shown in the following from [35, Thm. 5].

Theorem 3.4. *Suppose that $y \in \mathbb{R}^N$ and $\beta \in \mathbb{C}^{N-d}$ such that the three items in Proposition 3.3 hold. Let E be the set of endpoints contained in $\mathbb{C}^N \times \mathbb{P}^{N-d}$ of the homotopy paths starting at the points of S at $t = 1$ defined by $H_{y,\beta}(x,\lambda,t) = 0$ and $\pi_1(x,\lambda) = x$. Then, $\pi_1(E) \cap V_{\mathbb{R}}$ contains finitely many points, one of which is a global minimizer of (3.2). Hence, $V_{\mathbb{R}} = \varnothing$ if and only if $\pi_1(E) \cap V_{\mathbb{R}} = \varnothing$.*

Since $\pi(E) \cap V_{\mathbb{R}}$ consists of finitely many points, a global minimizer of (3.2) is identified by simply minimizing over these finitely many points.

4 Generating samples

This section presents an algorithm integrating Theorem 3.4 with geometric tools to produce provably dense samples of real algebraic varieties. The input and output of the algorithm are as follows.

Input:

- Polynomial equations $f_1, \ldots, f_{N-d} \in \mathbb{R}[x_1, \ldots, x_N]$ defining a pure d-dimensional real algebraic variety $X = \mathrm{V}_{\mathbb{R}}(f_1, \ldots, f_{N-d})$.

- A compact region $R \subseteq \mathbb{R}^N$ of the form $R = [a_1, b_1] \times \cdots \times [a_N, b_N]$. We call any regions of this form boxes.

- A sampling density $\epsilon > 0$.

- An estimation error δ with $0 \leqslant \delta \leqslant \epsilon$.

Output: A (finite) set of points $\mathcal{X} \subseteq \mathbb{R}^N$ that form a (δ, ϵ)-sample of $X \cap R$.

Proposition 3.3 and Theorem 3.4 provide a computationally tractable approach to finding very accurate estimated solutions of the optimization problem (3.2) for generic $y \in \mathbb{R}^N$. Following the terminology of these two results, we can define a subroutine `MinDistance` which takes a point $y \in \mathbb{R}^N$ as input, and outputs a set S consisting of one point s_q with $d(q, s_q) \leqslant \delta$ for every point $q \in \pi_1(E) \cap V$. The subroutine follows these steps on input y:

1. Choose a parameter $\beta \in \mathbb{C}^{N-d}$ such that Theorem 3.4 holds for the pair (y, β), which exists for generic y, and construct the homotopy $H_{y,\beta}$ using the polynomial system f defining X.

2. Track the homotopy paths of $H_{y,\beta}$, which are guaranteed to exist by Theorem 3.4, to obtain the elements of $\pi_1(E) \cap V_{\mathbb{R}}$ up to numerical error δ.

The smallest distance from y to a point in `MinDistance`(y) solves the problem (3.2) up to error δ. Repeatedly solving the minimum distance problem this way yields enough information to construct a provably dense sampling of X. Neglecting estimation error momentarily, the sampling algorithm's core consists of a short loop which computes the desired sample points iteratively. Denoting the open ball of radius r centered at y by $B_r(y)$ for any $r > 0$, this short loop is:

1. Choose an appropriate new "test point" $y \in \mathbb{R}^N$.

2. Run `MinDistance`(y) and place the returned points into the set of output points. Each sample point s covers a region $B_\epsilon(s)$ of points in X that are within distance ϵ of s. Let $d = d_X(y)$ be the minimum distance from y to X which can be calculated from the points returned by `MinDistance`(y). Thus, the region $B_d(y)$ does not contain any points of X. Store information about $B_d(y)$ and $B_\epsilon(s)$ (for all returned sample points s) for later use.

3. Check to see if the union all of the regions of the form $B_\epsilon(s)$ and $B_d(y)$ found in previous iterations of Step 2 contains R. If so, stop and output the sample points which have been collected. Otherwise, return to Step 1.

Remark 4.1. The stopping condition in Step 3 above guarantees that the outputted sample is a dense sample of $X \cap R$. Suppose that $R \subseteq \mathcal{B} \cup \mathcal{C}$, where $\mathcal{B} = \cup_{s \in S} B_\epsilon(s)$ for some subset S of X, and $\mathcal{C} \cap X = \varnothing$. Then for any $x \in X \cap R$, it follows that $x \in \mathcal{B}$, so $d(x, s_0) < \epsilon$ for some $s_0 \in S$. Thus, $d_S(x) < \epsilon$.

The full sampling algorithm tracks the spatial information for Steps 1 and 3 above by recursively dividing the region R into smaller boxes as necessary. Let `SplitBox` be a subroutine which takes as input a box $A = [c_1, d_1] \times \cdots \times [c_N, d_N] \subseteq \mathbb{R}^N$. It returns a pairwise disjoint set of smaller boxes $\{A_1, \ldots, A_k\}$ such that $A = \cup_{i=1}^k A_i$. We can arrange repeated applications of `SplitBox` into a tree structure.

Definition 4.2. Let T_R be a tree with root R whose nodes are boxes in $\mathbb{R}^N$. The children of R in T_R are the elements of `SplitBox`(R). Suppose that all the $(n-1)$-children of R in T_R have been defined where $n > 1$. Then the n-children of R are the elements of `SplitBox(C)` for every $(n-1)$-child C of R. The elements of `SplitBox(C)` have parent node C.

For technical reasons, repeated applications of `SplitBox` must eventually split an input region $A = [c_1, d_1] \times \cdots \times [c_N, d_N]$ into arbitrarily small pieces. Put precisely, given any $\gamma > 0$ and input region A, there is some n such that all n-children of A in T_A have maximum side length at most γ.

As an example, consider a version of `SplitBox` that when applied to A returns the two boxes $[c_1, d_1] \times \cdots \times [c_j, \frac{c_j+d_j}{2}] \times \cdots \times [c_N, d_N]$ and $[c_1, d_1] \times \cdots \times [\frac{c_j+d_j}{2}, d_j] \times \cdots \times [c_N, d_N]$ where $|d_j - c_j|$ is the maximum side length for the box A. Using `SplitBox` the sampling algorithm conducts a breadth first search of T_R while iteratively building the output sample.

Algorithm 4.3 Sampling algorithm

```
 1: Initialize an empty spatial database CoveredRegions which can store and retrieve informa-
    tion about subregions of $\mathbb{R}^N$
 2: Initialize an empty list SampleOutput of points in $\mathbb{R}^N$
 3: for each node $M$ in $T_R$ not marked “done”, iterated via breadth first search do
 4:     if The maximum side length of $M$ is at most $\frac{\epsilon-\delta}{\sqrt{N}}$ or $M$ does not intersect any region stored
    in CoveredRegions then
 5:         Run MinDistance(y) where $y$ is the center point of $M$, returning a set of sample points
 6:         $S$ with minimum distance $d$ from $y$ to any point in $S$. Add regions $B_{d-\delta}(y)$ and
 7:         $B_\epsilon(s)$ for each $s \in S$ to CoveredRegions. Add each $s \in S$ to SampleOutput.
 8:     end if
 9:     if $M \subseteq B$ for any region $B$ contained in CoveredRegions then
10:         Mark $M$ and all nodes in the subtree rooted at $M$ “done” and stop
11:         searching the subtree rooted at $M$.
12:     end if
13:     if All unsearched boxes in $T_R$ are marked “done” then
14:         End loop.
15:     end if
16: end for
17: return SampleOutput
```

Theorem 4.4. *Algorithm 4.3 terminates and outputs a (δ, ϵ)-sample of $X \cap R$.*

Before proving Theorem 4.4, we consider the following.

Lemma 4.5. *(1) Let A be a box in $\mathbb{R}^N$ with maximum side length less than $\frac{\epsilon-\delta}{\sqrt{N}}$. If y, S, and d take values as in lines 5-7 of Algorithm 4.3, then either $A \subseteq B_\epsilon(s)$ where $s \in S$ minimizes the distance to y, or $A \subseteq B_{d-\delta}(y)$. (2) Let T_A be a tree of boxes in $\mathbb{R}^N$ with root A constructed via `SplitBox` in the same manner as T_R, and let T'_A be a finite subtree of T_A such that if a node M is in T'_A and is not a leaf, all the first children of M in T_A are contained in T'_A. If $\mathcal{L} = \{L_1, \ldots, L_k\}$ are the leaf nodes of T'_A, the equality $A = \cup_{i=1}^k C_i$ follows.*

Proof. (1): Let γ be the maximum side length of A and suppose that $y = (y_1, \ldots, y_N)^T$. Without loss of generality we can replace A with the hypercube $\Pi_{i=1}^N [y_i - \frac{\gamma}{2}, y_i + \frac{\gamma}{2}]$ since A is a subset of the latter box. A has diagonal length $\Delta = \gamma\sqrt{N}$, which by assumption is less than $\epsilon - \delta$. Let $a \in A$ be an arbitrary point, and note that the maximum distance from a to y is half the length of the diagonal Δ. Suppose that $d = d(y, s) \leqslant \frac{\Delta}{2} + \delta$. Then for any point in $a \in A$, it follows that $d(a, s) \leqslant d(y, s) + d(y, a) \leqslant \Delta + \delta < \epsilon - \delta + \delta = \epsilon$. Therefore $A \subseteq B_\epsilon(s)$. Otherwise, suppose $d = d(y, s) > \frac{\Delta}{2} + \delta$. Then $d - \delta > \frac{\Delta}{2}$. Since the maximum value of $d(y, a)$ is $\frac{\Delta}{2}$, it follows that $a \in B_{d-\delta}(y)$, so $A \subseteq B_{d-\delta}(y)$.

(2): We proceed by induction on the maximum depth of the tree T'_A. Suppose that the depth of T'_A is 0. Then T'_A is a tree that consists of one leaf node, the box A, and (2) holds trivially. Suppose that (2) holds for any box B, corresponding tree T_B, and subtree T'_B with depth at most $k-1$ where $k \geqslant 1$. Then if T'_A has depth k, T'_A contains all the nodes $\texttt{SplitBox}(A) = \{A_1, \ldots, A_j\}$ by assumption. Note that T'_A is the union of the root A along with finite subtrees fulfilling the conditions of (2) rooted at $A_1, \ldots, A_j$, and that the set $\mathcal{L}$ of leaf nodes of T'_A is the union $\mathcal{L}_1 \cup \cdots \cup \mathcal{L}_j$ where $\mathcal{L}_i$ is the set of leaf nodes of the subtree rooted at A_i. By the induction assumption, $\cup_{L\in\mathcal{L}_i} L = A_i$. Therefore $A = \cup_{i=1}^{j} A_i = \cup_{i=1}^{j} \cup_{L\in\mathcal{L}_i} L = \cup_{L\in\mathcal{L}} L$ as desired. □

Proof of Theorem 4.4. (Termination): Let $\alpha = \frac{\epsilon-\delta}{\sqrt{N}}$. By our assumption on `SplitBox` there is an n such that the n-children of R in T_R have side length less than α. Therefore if M is any n-child of R, lines 5-7 of the algorithm will run if M is searched. Part (1) of Lemma 4.5 shows that lines 10-11 will run on M subsequently. Therefore the algorithm's breadth first search terminates at maximum depth n.

(Completeness): Let T'_R be the subtree of T_R which Algorithm 4.3 searches before terminating. By construction, T'_R fulfills the conditions of Lemma 4.5 part (2). If $\mathcal{L}$ is the set of leaf nodes in T'_R, then $R = \cup_{L\in\mathcal{L}} L$ follows. Let S be SampleOutput which was returned by the algorithm and Y be the set of center points of balls with form $B_{d-\delta}(y)$ in CoveredRegions. By construction any element $L \in \mathcal{L}$ has $L \subseteq B_\epsilon(s)$ for some $s \in S$ or $L \subseteq B_{d-\delta}(y)$ for some $y \in Y$. By Theorem 3.4 and the definition of `MinDistance` it follows that $X \cap (\cup_{y\in Y} B_{d-\delta}(y)) = \varnothing$. Similarly to Remark 4.1, we have that $X \cap R \subseteq \cup_{s\in S} B_\epsilon(s)$. We also have $d_X(s) \leqslant \delta$ for all $s \in S$ by definition of `MinDistance`. Thus S is a (δ, ϵ)-sample of $X \cap R$.

□

In practice, there are two opposing resource demands the algorithm needs to balance. The `MinDistance` step in Algorithm 4.3's core loop consumes significantly more time than any other individual step, so an optimal run of the Algorithm makes as few calls to `MinDistance` as possible. Resource demands for processing the Algorithm's output with data analysis methods scale with the number of points in the sample. Also, with more points in the sample more resources are consumed accessing and storing information in the spatial database used throughout the Algorithm. An optimal output sample therefore contains as few points as possible while being provably dense. We can adjust the Algorithm's components, integrating geometric heuristics both to reduce `MinDistance` calls and output sizes. These heuristics include:

- *Dynamic box splitting* - Instead of splitting along the longest side of a box B with `SplitBox`, split B so that the largest intersection (by Lebesgue measure) of B with a region stored in CoveredRegions is a box in the output.

- *Dynamic sampling* - Refuse to add points to the output sample if their distance to the nearest point already in SampleOutput is less than some threshold.

- *Heuristic tree searching* - Place priority on first searching and applying `MinDistance` to the largest boxes (by Lebesgue measure) at each level of depth in the search tree. Larger boxes represent larger regions which potentially do not intersect X, and so a single run of `MinDistance` has the potential to lead to the exclusion of a much larger box $B_{d-\delta}(y)$.

See [27] for an extended discussion of both the heuristics and implementation.

5 Examples

Algorithm 4.3 has been implemented and used to produce dense samples of varieties for further processing via persistent homology. The implementation is publicly available as the Python package `tdasampling` on PyPi and the package source code is available at https://github.com/P-Edwards/tdasampling. Data, algorithm parameters, plots, and other scripts for the examples are available at https://github.com/P-Edwards/sampling-varieties-data. Vietoris-Rips persistent homology calculations were performed using the package Ripser [6]. Plots of persistence diagrams were produced using a modified version of a plotting script included with DIPHA [7].

In the following examples, the persistence diagrams are decorated as in Figure 4. Points in the highlighted region of an example's diagram correspond to homological features in the underlying variety, assuming the diagram was produced from a (δ, ϵ)-sample of a variety with homological feature size at least $2(\epsilon + \delta)$.

5.1 Clifford torus

The Clifford torus T is an embedding of the product of two circles, $S^1 \times S^1$, into $\mathbb{R}^4$. It is also a pure 2-dimensional algebraic variety defined by two equations in four variables:

$$T = \mathrm{V}_{\mathbb{R}}(x_1^2 + y_1^2 - \frac{1}{2}, x_2^2 + y_2^2 - \frac{1}{2}).$$

Since T is a torus, its Betti numbers are known theoretically to be $\beta_0 = 1, \beta_1 = 2$, and $\beta_2 = 1$. Note that T is compact as it is contained in the closed ball $\overline{B_1(0)}$ in $\mathbb{R}^4$. A sample of T was obtained by using Algorithm 4.3 to produce a $(10^{-7}, 0.14)$ sample of T (the bounding box used was $[-1, 1]^4$). The sample contains 5,689 points.

Vietoris-Rips persistent homology thresholded to a parameter value of 0.60 was subsequently calculated for the sample. The points in the persistence diagram represent features born before 0.60, and the points on the top edge of the diagram represent features that do not die at 0.60 or earlier. The shaded region in the diagram is derived from Corollary 2.17 assuming the homological feature size of the torus is at least $2(0.14 + 10^{-7})$. Recall from the Corollary that the number of points above and to the left of the point $\left(2\epsilon\sqrt{\frac{4+1}{(2)(4)}}, 4\epsilon + 2\delta\right)$, where the sample is a (δ, ϵ) sample of T, is a lower bound on T's Betti numbers. In this case, $\left(2\epsilon\sqrt{\frac{4+1}{(2)(4)}}, 4\epsilon + 2\delta\right) \approx (0.221, 0.56)$. In Figure 5, the shaded region consists of all points above and to the left of $(0.221, 0.56)$, the region's bottom right corner. The persistence diagram in Figure 5 mirrors the expected theoretical results. A connected component and two 1-dimensional homology features appear in the shaded region, and one long-lived 2-dimensional homology feature also appears in the diagram.

5.2 Quartic surfaces

Restricting to the box $[-3, 3] \times [-3, 3] \times [-3, 3]$, we next consider the real algebraic varieties

$$V_1 = \mathrm{V}_{\mathbb{R}}(4x^4 + 7y^4 + 3z^4 - 3 - 8x^3 + 2x^2y - 4x^2 - 8xy^2 - 5xy + 8x - 6y^3 + 8y^2 + 4y)$$

$$V_2 = \mathrm{V}_{\mathbb{R}}\begin{pmatrix} 144x^4 + 144y^4 - 225(x^2 + y^2)z^2 + 350x^2y^2 \\ + 81z^4 + x^3 + 7x^2y + 3x^2 + 3xy^2 - 4x - 5y^3 + 5y^2 + 5y \end{pmatrix}.$$

Both quartic equations define pure 2-dimensional varieties. Figure 6 displays visualizations of both V_1 and V_2 using the gathered samples allowing for a qualitative analysis. In particular, V_1 appears to be a sphere up to homotopy, with two distinct sphere-like features.

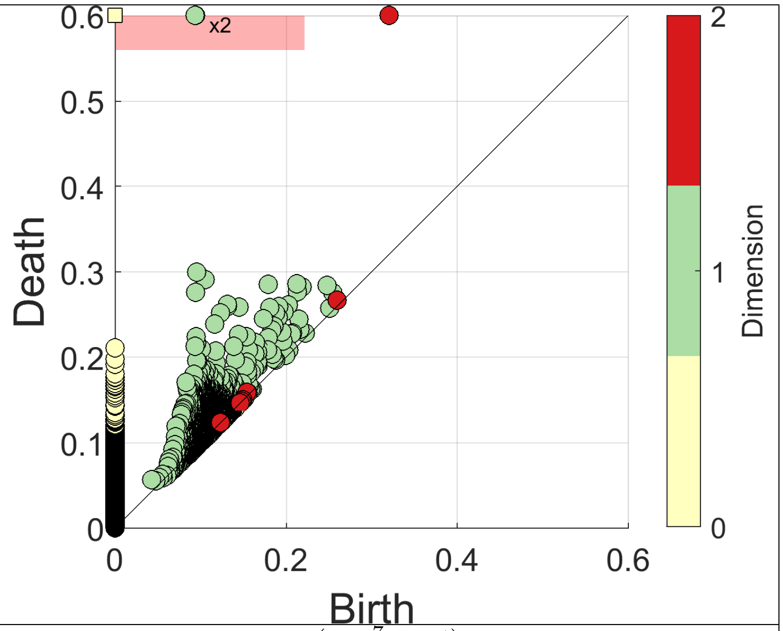
Persistence diagram	
Sampling density	$(10^{-7}, 0.14)$
Estimated Betti numbers	$\beta_0 = 1, \beta_1 = 2, \beta_2 = 0$

Figure 5: Persistent homology results dervied from sampling the Clifford torus. Points in the shaded region of the persistence diagram provably correspond to homology features in the underlying space.

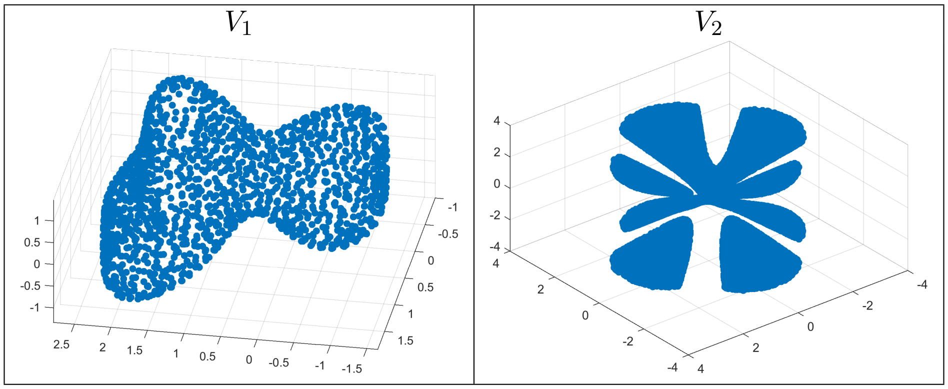


Figure 6: Quartic surfaces sampled using Algorithm 4.3.

Samples produced for V_1 and V_2 contain 1,511 and 13,904 points respectively. The persistent homology results in Figure 7 show that V_1 has homology features corresponding to a 2-sphere. The persistence diagram for V_1 shows how the persistence diagram also captures geometric information about V_1 beyond just its Betti numbers. A 2-dimensional point which is relatively far away from the diagonal but not in the shaded region appears in the persistence diagram for V_1, and corresponds to the smaller of the two sphere-like features. The only homology features confirmed for V_2 in Figure 8 are 5 connected components.

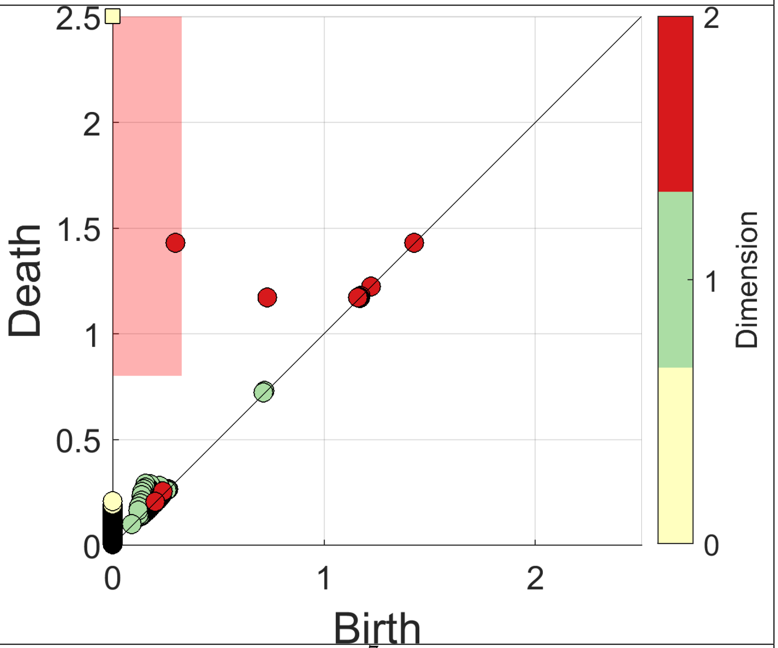
Variety	V_1
Persistence diagram	
Sampling density	$(10^{-7}, 0.20)$
Estimated Betti numbers	$\beta_0 = 1, \beta_1 = 0, \beta_2 = 1$

Figure 7: Persistent homology results derived from sampling the variety V_1.

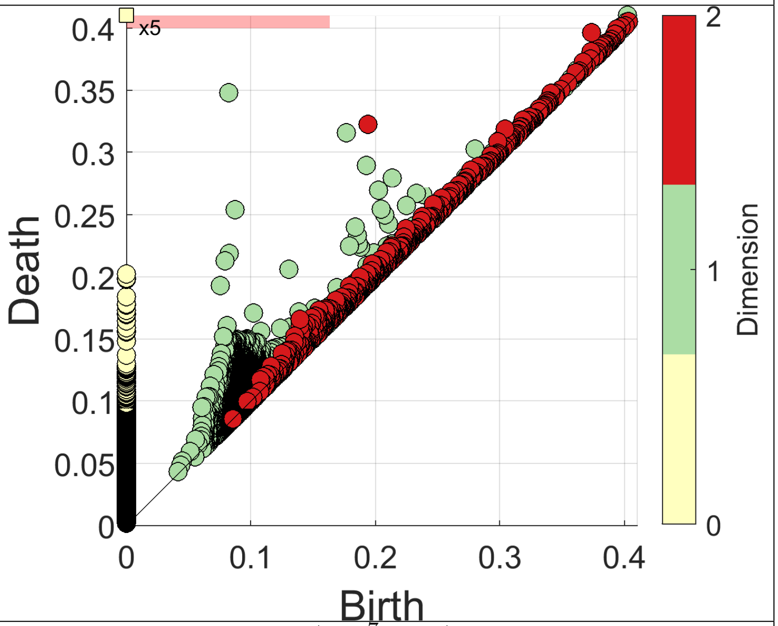
Variety	V_2
Persistence diagram	
Sampling density	$(10^{-7}, 0.10)$
Estimated Betti numbers	$\beta_0 = 5, \beta_1 = 0, \beta_2 = 0$

Figure 8: Persistent homology results derived from sampling the variety V_2. The calculation for V_2 has been thresholded to a parameter value of 0.405.

5.3 Deformable pentagonal linkages

For a more elaborate example, we also analyze a kinematics inspired polynomial system. Consider a regular pentagon in the plane consisting of links with unit length, and with one of the links fixed to lie along the x-axis with leftmost point at $(0,0)$. The space V_p of all possible configurations of this regular pentagon is a real algebraic variety. Farber and Schütz study this type of configuration space in [28], as well as provide an overview of its study. A specialization of their results shows that β_0 of V_p is 1, β_1 is 8, and β_2 is 1.

We describe the polynomials defining V_p as explained in [11]. Number the links in order around the loop with link 0 as the fixed link, and let θ_i for $i = 0, \ldots, 4$ be the absolute rotation of the i-th link in the plane. Note that $\theta_0 = 0$ because link 0 is fixed, and θ_4 is totally determined by θ_i for $i = 1, 2, 3$ since one of link 4's endpoints is $(0,0)$ and the other is an endpoint of link 3. A 3-tuple $(\theta_1, \theta_2, \theta_3)$ defines a valid regular pentagon only if the free endpoint of link 3 is distance 1 from $(0,0)$. Letting $s_i = \sin(\theta_i)$ and $c_i = \cos(\theta_i)$ for $i = 1, \ldots, 3$, we have the polynomial condition $(s_1 + s_2 + s_3)^2 + (1 + c_1 + c_2 + c_3)^2 = 1$. To enforce that s_i and c_i are sine-cosine pairs for $i = 1, \ldots, 3$, we require the equations $s_i^2 + c_i^2 = 1$. Assembling these constraints together, the configuration space for deformable regular pentagons is modelled by a compact pure 2-dimensional real algebraic variety in the six variables s_1, s_2, s_3 and c_1, c_2, c_3 with four equations:

$$V_p = V_{\mathbb{R}} \begin{pmatrix} s_1^2 + c_1^2 - 1, \\ s_2^2 + c_2^2 - 1, \\ s_3^2 + c_3^2 - 1, \\ (s_1 + s_2 + s_3)^2 + (1 + c_1 + c_2 + c_3)^2 - 1 \end{pmatrix}.$$

A $(10^{-7}, 1.12)$ sample of V_p was produced by first obtaining a $(10^{-7}, 1.0)$ sample using Algorithm 4.3. This sample was then sub-sampled by iteratively choosing a point in the sample, removing all other points within .12 of the chosen point, and repeating this loop until all points in the subsample had no other points within distance .12. The sample contains 3,548 points, and persistent homology calculations were thresholded to distance value 2.2. The persistent homology results are summarized in Figure 9. The points far from the diagonal on the left hand side capture the theoretically expected homology for the configuration space. Though 2 features in dimension 2 (voids) appear on the right hand side of the persistent diagram, those features persist for a shorter period of time than the features on the left.

6 Conclusion and Future Work

The sampling algorithm presented in this paper is a first step towards systematizing the use of the TDA for obtaining geometric and topological information from algebraic varieties, including those that arise in applications. Our use of numerical algebraic geometry methods in the sampling process is unique among sampling approaches, and enables our algorithm to simultaneously satisfy both theoretical and practical constraints for applying TDA. The examples we provide in Section 5 illustrate how using the PH pipeline approach allows for the extraction of detailed information beyond Betti numbers on a real algebraic variety.

A step forward would be to derive and incorporate further information about the geometric structure of singular varieties into systematic TDA based analysis. A real algebraic variety X can be stratified into singular regions: $X \supset X_0 \supset X_1 \supset X_2 \supset \ldots \supset X_t$, where X_0 is the singular locus of X, X_1 is the singular locus of X_0, and so on. A classic result of Whitney [52] shows that this

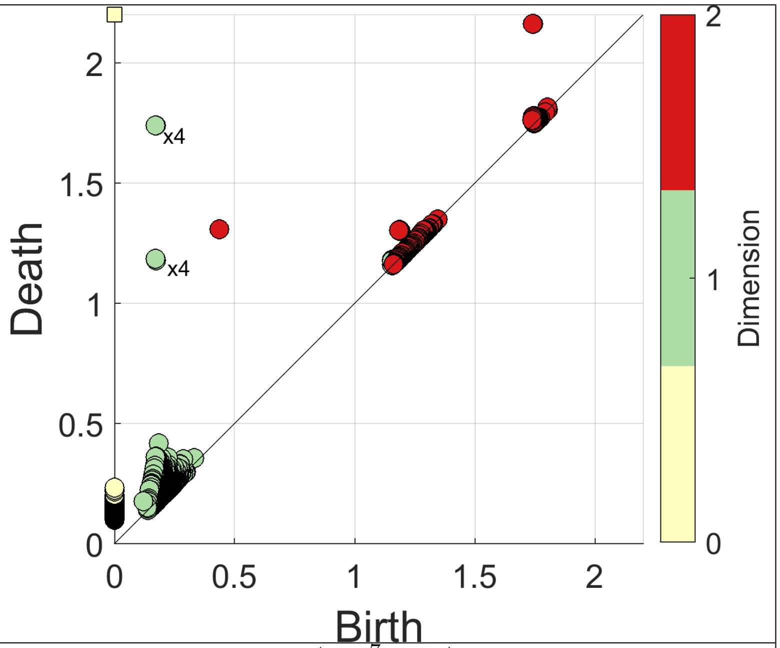
Persistence diagram	
Sampling density	$(10^{-7}, 1.12)$

Figure 9: Persistence diagram computed by sampling the configuration space of deformable pentagonal linkages.

stratification presents X as a stratified manifold. Alternatively, a variety X can also be given an *isosingular stratification* [34], breaking it into strata based on its singularity structure. Running the PH pipeline on individual strata after identifying them via stratification methods for samples (for instance: [9]) or algebraic methods (detailed in [34]) would result in an even more detailed summary of the variety to use for either dimensionality reduction or machine learning. Another direction would be to apply persistent homology of ellipsoids rather than ϵ-balls [13].

For a variety X, our work also raises the natural question of theoretically determining or computationally estimating a lower bound on the weak feature size of X. Future work will explore how to exploit the algebraic description of a variety in computing these quantities. Finally, it would be worthwhile to investigate the noise induced from sampling via homotopy continuation in the context of off-set varieties [36].

Acknowledgements

This paper arises from research done while ED and HAH were funded by the John Fell Oxford University Press (OUP) Research Fund. ED was also funded via a Anne McLaren fellowship from the University of Nottingham. PBE thanks Peter Bubenik for helpful template scripts to generate simplicial complex figures and Bernd Sturmfels for the example quartic equations. HAH acknowledges funding from EPSRC EP/K041096/1, EP/R005125/1, EP/R018472/1 and Royal Society University Research Fellowship. JDH was supported in part by NSF CCF 1812746, Army YIP W911NF-15-1-0219, Sloan Research Fellowship BR2014-110 TR14, and ONR N00014-16-1-2722.

References

[1] Nina Amenta and Marshall Bern. Surface reconstruction by voronoi filtering. *Discrete & Computational Geometry*, 22(4):481–504, 1999.

[2] Nina Amenta, Marshall Bern, and Manolis Kamvysselis. A new voronoi-based surface reconstruction algorithm. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 415–421. ACM, 1998.

[3] Philippe Aubry, Fabrice Rouillier, and Mohab Safey El Din. Real solving for positive dimensional systems. *J. Symbolic Comput.*, 34(6):543–560, 2002.

[4] Saugata Basu. Computing the first few Betti numbers of semi-algebraic sets in single exponential time. *Journal of Symbolic Computation*, 41(10):1125–1154, 2006.

[5] Daniel J Bates, Jonathan D Hauenstein, Andrew J Sommese, and Charles W Wampler. *Numerically solving polynomial systems with Bertini*, volume 25. SIAM, 2013.

[6] Ulrich Bauer. Ripser. https://github.com/Ripser/ripser, 2016.

[7] Ulrich Bauer, Michael Kerber, and Jan Reininghaus. Dipha (a distributed persistent homology algorithm). Software available at https://github.com/DIPHA/dipha.

[8] Ulrich Bauer, Michael Kerber, and Jan Reininghaus. Clear and compress: Computing persistent homology in chunks. In *Topological methods in data analysis and visualization III*, pages 103–117. Springer, 2014.

[9] Paul Bendich, Bei Wang, and Sayan Mukherjee. Local homology transfer and stratification learning. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, pages 1355–1370. Society for Industrial and Applied Mathematics, 2012.

[10] Matthew Berger, Andrea Tagliasacchi, Lee Seversky, Pierre Alliez, Joshua Levine, Andrei Sharf, and Claudio Silva. State of the art in surface reconstruction from point clouds. In *EUROGRAPHICS star reports*, volume 1, pages 161–185, 2014.

[11] Daniel A. Brake, Daniel J. Bates, Wenrui Hao, Jonathan D. Hauenstein, Andrew J. Sommese, and Charles W. Wampler. Algorithm 976: Bertini_real: Numerical decomposition of real algebraic curves and surfaces. *ACM Trans. Math. Softw.*, 44(1):10:1–10:30, July 2017.

[12] Paul Breiding and Orlando Marigliano. Sampling from the uniform distribution on an algebraic manifold. arXiv preprint arXiv:1810.06271, 2018.

[13] Paul Breiding, Sara Kalisnik Verovsek, Bernd Sturmfels, and Madeleine Weinstein. Learning algebraic varieties from samples. *Revista Matemtica Complutense*, 31(3):545–593, 2018.

[14] Peter Bubenik, Vin De Silva, and Jonathan Scott. Metrics for generalized persistence modules. *Foundations of Computational Mathematics*, 15(6):1501–1531, 2015.

[15] Peter Bubenik and Jonathan A Scott. Categorification of persistent homology. *Discrete & Computational Geometry*, 51(3):600–627, 2014.

[16] Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, 2009.

[17] Frédéric Chazal, David Cohen-Steiner, and André Lieutier. A sampling theory for compact sets in euclidean space. *Discrete & Computational Geometry*, 41(3):461–479, 2009.

[18] Frédéric Chazal and André Lieutier. Weak feature size and persistent homology: computing homology of solids in $\mathbb{R}^n$ from noisy data samples. In *Proceedings of the twenty-first annual symposium on Computational geometry*, pages 255–262. ACM, 2005.

[19] Frédéric Chazal and Bertrand Michel. An introduction to topological data analysis: fundamental and practical aspects for data scientists. *arXiv preprint arXiv:1710.04019*, 2017.

[20] Chao Chen and Michael Kerber. Persistent homology computation with a twist. In *Proceedings 27th European Workshop on Computational Geometry*, volume 11, 2011.

[21] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37(1):103–120, 2007.

[22] Felipe Cucker, Teresa Krick, and Michael Shub. Computing the homology of real projective sets. *Foundations of Computational Mathematics*, pages 1–42, 2016.

[23] Vin De Silva and Robert Ghrist. Coverage in sensor networks via persistent homology. *Algebraic & Geometric Topology*, 7(1):339–358, 2007.

[24] Tamal Krishna Dey, Xiaoyin Ge, Qichao Que, Issam Safa, L Wang, and Yusu Wang. Feature-preserving reconstruction of singular surfaces. In *Computer Graphics Forum*, volume 31, pages 1787–1796. Wiley Online Library, 2012.

[25] Jan Draisma, Emil Horobeţ, Giorgio Ottaviani, Bernd Sturmfels, and Rekha Thomas. The Euclidean distance degree. In *SNC 2014—Proceedings of the 2014 Symposium on Symbolic-Numeric Computation*, pages 9–16. ACM, New York, 2014.

[26] Herbert Edelsbrunner and John Harer. *Computational topology: an introduction.* American Mathematical Soc., 2010.

[27] Parker B. Edwards. Topological data analysis for real algebraic varieties. Master's thesis, University of Oxford, 2016.

[28] Michael Farber and Dirk Schütz. Homology of planar polygon spaces. *Geometriae Dedicata*, 125(1):75–92, 2007.

[29] Daniel Freedman. An incremental algorithm for reconstruction of surfaces of arbitrary codimension. *Computational Geometry*, 36(2):106–116, 2007.

[30] Joseph Howland Guthrie Fu. Tubular neighborhoods in Euclidean spaces. *Duke Mathematical Journal*, 52(4):1025–1046, 1985.

[31] Robert Ghrist. Barcodes: the persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1):61–75, 2008.

[32] Elizabeth Gross, Heather A. Harrington, Zvi Rosen, and Bernd Sturmfels. Algebraic Systems Biology: A Case Study for the Wnt Pathway. *Bulletin of Mathematical Biology*, 78(1):21–51, 2016.

[33] Allen Hatcher. *Algebraic Topology.* Cambridge University Press, 2002.

[34] Jonathan D Hauenstein and Charles W Wampler. Isosingular sets and deflation. *Foundations of Computational Mathematics*, 13(3):371–403, 2013.

[35] Jonthan D. Hauenstein. Numerically computing real points on algebraic sets. *Acta Appl. Math.*, 125:105–119, 2013.

[36] Emil Horobet and Madeleine Weinstein. Offset hypersurfaces and persistent homology of algebraic varieties. *arXiv preprint arXiv:1803.07281*, 2018.

[37] Shawn Martin, Aidan Thompson, Evangelos A Coutsias, and Jean-Paul Watson. Topology of cyclo-octane energy landscape. *The journal of chemical physics*, 132(23):234115, 2010.

[38] Shawn Martin and Jean-Paul Watson. Non-manifold surface reconstruction from high-dimensional point cloud data. *Computational Geometry*, 44(8):427–441, 2011.

[39] Rodrigo Mendoza-Smith and Jared Tanner. Parallel multi-scale reduction of persistent homology filtrations, 2017.

[40] Nikola Milosavljević, Dmitriy Morozov, and Primoz Skraba. Zigzag persistent homology in matrix multiplication time. In *Proceedings of the twenty-seventh annual symposium on Computational geometry*, pages 216–225. ACM, 2011.

[41] Alexander P. Morgan and Andrew J. Sommese. Coefficient-parameter polynomial continuation. *Appl. Math. Comput.*, 29(2, part II):123–160, 1989.

[42] Bernard Mourrain and Jean Pascal Pavone. Subdivision methods for solving polynomial equations. *Journal of Symbolic Computation*, 44(3):292–306, 2009.

[43] Mircea Mustaţă. Graded betti numbers of general finite subsets of points on projective varieties. *Le Matematiche*, 53(3):53–81, 1998.

[44] Partha Niyogi, Stephen Smale, and Shmuel Weinberger. Finding the Homology of Submanifolds with High Confidence from Random Samples. *Discrete & Computational Geometry*, 39(1-3):419–441, 2008.

[45] Nina Otter, Mason A Porter, Ulrike Tillmann, Peter Grindrod, and Heather A Harrington. A roadmap for the computation of persistent homology. *EPJ Data Science*, 6(1):17, 2017.

[46] Steve Y Oudot. *Persistence theory: from quiver representations to data analysis*, volume 209. American Mathematical Society, 2015.

[47] Fabrice Rouillier, Marie-Francoise Roy, and Mohab Safey El Din. Finding at least one point in each connected component of a real algebraic set defined by a single equation. *J. Complexity*, 16(4):716–750, 2000.

[48] Peter Scheiblechner. On the complexity of deciding connectedness and computing betti numbers of a complex algebraic variety. *Journal of Complexity*, 23(3):359–379, 2007.

[49] Abraham Seidenberg. A new decision method for elementary algebra. *Ann. of Math. (2)*, 60:365–374, 1954.

[50] Evan C Sherbrooke and Nicholas M Patrikalakis. Computation of the solutions of nonlinear polynomial systems. *Computer Aided Geometric Design*, 10(5):379–405, 1993.

[51] Andrew Sommese and Charles Wampler. *The Numerical solution of systems of polynomials arising in engineering and science*, volume 99. World Scientific, 2005.

[52] Hassler Whitney. Elementary Structure of Real Algebraic Varieties. *The Annals of Mathematics*, 66(3):545, 1957.

[53] Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33(2):249–274, 2005.