Sampling real algebraic varieties for topological data analysis

Emilie Dufresne
Department of Mathematics
University of York
York, UK
emilie.dufresne@york.ac.uk

Parker B. Edwards
Department of Mathematics
University of Florida
Gainesville, FL, USA
pedwards@ufl.edu

Heather A. Harrington
Mathematical Institute
University of Oxford
Oxford, UK
harrington@maths.ox.ac.uk

Jonathan D. Hauenstein
Department of Applied and Computational Mathematics and Statistics
University of Notre Dame
Notre Dame, IN, USA
hauenstein@nd.edu

***Abstract*—Topological data analysis (TDA) provides tools for computing geometric and topological information about spaces from a finite sample of points. We present an 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 showcase it with several examples.**

***Index Terms*—topological data analysis, real algebraic varieties, dense samples, numerical algebraic geometry, minimal distance**

I. Introduction

The geometry and topology of real algebraic varieties is a challenging problem in applications modelled by polynomial systems. For kinematics problems, geometric insight about configuration spaces can lead to physical insights (e.g., [40]), while the geometry of varieties provide information about biochemical systems (e.g., [32]). Here, we present a new algorithm fulfilling a key step in applying topological data analysis methods (TDA), particularly persistent homology [55] (PH), to real algebraic varieties. The algorithm takes as input a list of polynomials defining a real algebraic variety and outputs a sample of points on the variety tailored for input to PH.

A. Prior work

PH computes topological features closely related to a variety's Betti numbers. Several analyses and proposed algorithms [4], [22], [51], offer theoretical complexity guarantees for variants of the problem of computing Betti numbers given a list of defining polynomials as input; however, implementations are not available. Other approaches similar to PH take as input a sample of points from a variety with output that can be used to estimate Betti numbers. Extensive effort has produced a large number of surface reconstruction algorithms, particularly for nonsingular surfaces embedded in $\mathbb{R}^3$ such as [2], [10], [24], [29], [41]. None of these methods apply to general real varieties, while PH does. There is a probablistic algorithm for computing Betti numbers from uniform random point samples [47]. Given a sample of points from a real variety, one may alternatively compute other features. For a large enough sample of "general" points, [46] studies the "Betti diagram" of a projective variety. Given set of general points, one "learns" the equations defining the algebraic variety [13].

The algorithm in [12] produces samples from the uniform distribution on a variety. Among deterministic sampling approaches, subdivision and reduction sampling methods [45], [53] most closely resemble our algorithm. 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. (2) Adjusting current implementations to reduce the number of sample points is not straightforward. Computational resource requirements for PH scale up quickly with more input points.

Our approach for sampling varieties is based on numerical algebraic geometry, with the books [5], [54] 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. An 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.

B. Organization

The paper is organized as follows: we recall TDA theory and computations in Section II and numerical algebraic geometry in Section III. Section IV details the sampling algorithm,

proves its correctness, and discusses the geometric heuristics for sample minimization. In Section V, we illustrate our sampling algorithm with TDA on several examples.

II. Topological data analysis

Topological data analysis is a field of research encompassing theory and algorithms which adapt the theory of topology and geometry to analyze the "shape" of data. The goal of our sampling algorithm is to produce input for TDA algorithms.

We apply 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 [19], [26], [48], [49]. 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 $\mathbb{R}^N$. For some variants, input in the form of a matrix of pairwise distances suffices.
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*.

Steps 2 and 3 present competing requirements any sampling approach tailored for PH must balance. On the one hand, PH computation costs rise quickly the more points are added to the input sample. On the other hand, the algorithm must return sufficiently many points to construct a "good" sample of a viarety. See e.g. [26], [49] for detailed discussions of PH, and [34] for an introduction to homology.

A. Building simplicial complexes from data

Definition II.1. Let $\hat{X}$ be a finite subset of a metric space Y and ϵ be a real number. The *Čech complex* for $\hat{X}$ with parameter ϵ, $C_\epsilon(\hat{X})$, is a simplicial complex such that:

- If $\epsilon < 0$, $C_\epsilon(\hat{X})$ is defined directly to be $\varnothing$.
- The vertex set of $C_\epsilon(\hat{X})$ is $\hat{X}$.
- A set $x \subseteq \hat{X}$ belongs to $C_\epsilon(\hat{X})$ 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 $\hat{X}$ with parameter ϵ, denoted $R_\epsilon(\hat{X})$, is a simplicial complex that fulfills an alternative version of condition 3 above:

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

Calculating Čech complexes for reasonably sized point clouds presents computational issues since higher order intersections of balls must be checked and individual simplices must be stored. Vietoris-Rips complexes estimate Čech complexes, and can be constructed more easily since only pairs of balls need checking. 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 II.2 (de Silva and Ghrist [23])**.** *If $\hat{X}$ is a finite set of points in $\mathbb{R}^N$ and $\epsilon > 0$ there is a chain of inclusions*

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

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

B. Persistent homology

Consider a point cloud $\hat{X}$ sampled evenly from nearby some underlying space $X \subseteq \mathbb{R}^N$. PH methods consider all of the homology groups $H_p(C_\epsilon(X))$ simultaneously. Persistent homology theory provides an algebraic framework for tracking homology features as the parameter value ϵ changes. We summarize the categorical approach introduced in [15].

Definition II.3. Let k be a field ($\frac{\mathbb{Z}}{2}$ in all subsequent examples). 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 (finite dimensional) 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 $\epsilon \leqslant \epsilon' \leqslant \epsilon''$, $F(\epsilon \leqslant \epsilon'') = F(\epsilon' \leqslant \epsilon'') \circ F(\epsilon \leqslant \epsilon')$

Definition II.4. A point $\epsilon \in \mathbb{R}$ is *regular* for a persistence module F if there exists an open 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 II.5. For any finite point cloud $\hat{X} \subseteq \mathbb{R}^N$ and real numbers $0 \leqslant \epsilon \leqslant \epsilon'$, it follows directly from the definition that $C_\epsilon(\hat{X}) \subseteq C_{\epsilon'}(\hat{X})$. Fixing $p \geqslant 0$ and applying H_p results in a sequence of vector spaces and $\frac{\mathbb{Z}}{2}$-linear maps $H_p(C_\epsilon(\hat{X})) \hookrightarrow H_p(C_{\epsilon'}(\hat{X}))$ induced by inclusion. The assignment $\epsilon \mapsto H_p(C_\epsilon(\hat{X}))$ along with these linear maps defines a tame persistence module $H_pC_\bullet(\hat{X})$, which we will denote by HC. An analagous persistence module exists for the Vietoris-Rips complex denoted by HR. ◁

Definition II.6. The *rank function* of a tame module F assigns $x \leqslant y \mapsto \operatorname{rank} F(x \leqslant y)$ for every $x \leqslant y \in \mathbb{R}$. The *persistence diagram* of F is the multiset DF of points (x, y) with $x \leqslant y \in \mathbb{R} \cup \{-\infty, \infty\}$ where $\operatorname{rank}(x \leqslant y)$ is the number of points in DF above and to the left of (x, y).

Theorem II.7 (Fundamental Theorem of Persistent Homology)**.** *Let F and G be tame persistence modules. F and G are isomorphic if and only if "decorated" versions (see [49]) of DF and DG are equal.*

The original algebraic version of Theorem II.7 for PH appears in [55], and a categorical version in [15]. Each point (x, y) in a module's persistence diagram can be viewed as describing the range of parameter values through which a single independent feature in the module persists. See Fig. 1.

C. Computational considerations

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). In the worst case, the computational complexity for computing the persistence of $H_pR_\bullet(\hat{X}) = HR$ scales with the maximum number of $p+1$ simplices in R_ϵ attained at any parameter value ϵ. More precisely: if $\hat{X}$ contains m points, calculating the full persistence diagram for HR in the worst case has time complexity $O(\binom{m}{p+2}^\omega)$ where ω is the best known exponent for matrix multiplication [43]. The best known upper bound is $\omega < 2.3728639$ [38].

The Persistence Algorithm has been significantly optimized since its original formulation (for instance: [8], [20], [42]). Despite improvement in optimizations and implementations, limiting both the size of the point cloud m and the homology dimension p is often necessary to make PH computations feasible (see [48]). 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.

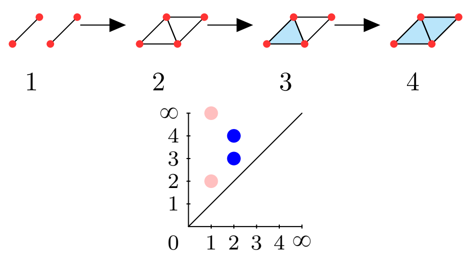

Fig. 1: The top figure shows a filtered complex as it changes with parameter value. The bottom figure depicts the persistence diagram. Blue points represent 0 dimensional homology and pink points represent 1 dimensional homology.

D. Homology inference

Suppose that $\hat{X} \subseteq \mathbb{R}^N$ 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 $\hat{X}$ recovers the homology of the X provided $\hat{X}$ 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 II.8. 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 1. Definition II.8 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.

Definition II.9. Let $X \subseteq \mathbb{R}^N$ be a compact metric space. The homological feature size of X, $\mathrm{hfs}(X)$, is the infimum of all positive critical values over all dimensions k of the persistence module $\epsilon \mapsto H_k(X^\epsilon)$ (negative ϵ are assigned $\varnothing$).

Remark 2. The homological feature size of a space X was introduced in [21], and is bounded below by the space's *reach* [1] and the space's *weak feature size* [17]. More precisely, $0 \leqslant \mathrm{reach}(X) \leqslant \mathrm{wfs}(X) \leqslant \mathrm{hfs}(X)$. The weak feature size of real semialgebraic sets is known to be positive ([30] §5.3), and so the homological feature size is positive as well. Compact real semialgebraic sets are absolute neighborhood retracts (see e.g. Theorem 3 of [39] and Corollary 3.5 of [33]), and in particular this implies $H_*(X) \cong H_*(X^\kappa)$ for any compact semialgebraic set and $\kappa < \mathrm{hfs}(X)$.

Theorem II.10 (Homology Inference Theorem, [18], [21]). *Let $\hat{X}, X \subseteq \mathbb{R}^N$, with X compact semialgebraic and $\hat{X}$ a finite (δ, ϵ)-sample of X where $0 \leqslant \delta \leqslant \epsilon$ and $\mathrm{hfs}(X) > 2(\epsilon + \delta)$. Letting $HC = H_pC_\bullet(\hat{X})$, the dimension of $H_p(X)$ is the number of points in $D(HC)$ 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 $X \hookrightarrow \hat{X}^\epsilon \hookrightarrow X^{\epsilon+\delta} \hookrightarrow \hat{X}^{2\epsilon+\delta} \hookrightarrow X^{2(\epsilon+\delta)}$. The Nerve Theorem (e.g. §4G.3 [34]) implies that $HC(a) \cong H_p(\hat{X}^a)$ 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

$$H_p(X) \longrightarrow HC(\epsilon) \longrightarrow H_p(X) \longrightarrow HC(2\epsilon + \delta) \longrightarrow H_p(X), \quad h: HC(\epsilon) \to HC(2\epsilon+\delta)$$

where the maps from $H_p(X)$ to itself are isomorphisms. Since there is an isomorphism from $H_p(X)$ to itself which factors through h, $\dim H_p(X) \leqslant \mathrm{rank}(h)$. h also factors through a map with domain $H_p(X)$, so $\mathrm{rank}(h) \leqslant \dim H_p(X)$. □

Corollary 1. *Let $HC, X, \hat{X}, \epsilon,$ and δ be as in Theorem II.10. 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 $HR = H_pR_\bullet(\hat{X})$ is a lower bound for $\dim H_p(X)$.*

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

$$HR(a) \longrightarrow HC(\epsilon) \longrightarrow HC(2\epsilon + \delta) \longrightarrow HR(4\epsilon + 2\delta), \quad h: HR(a) \to HR(4\epsilon + 2\delta).$$

It follows that $\mathrm{rank}(h) \leqslant \mathrm{rank}(HC(\epsilon \leqslant 2\epsilon + \delta))$. Theorem II.10 shows that the rank of $HC(\epsilon \leqslant 2\epsilon + \delta)$ is $\dim H_p(X)$. □

III. 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 $V_{\mathbb{R}}$ [52]. We summarize the use of numerical algebraic geometry to perform this computation based on [36] (see also [3], [25], [50]) with Section IV 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. This assumption simplifies formulating the critical point conditions of the minimization problem below.

Example III.1. The affine cone over the twisted cubic curve is the irreducible surface ($d = 2$)

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

defined by $x_2^2 - x_3x_1 = x_2x_3 - x_4x_1 = x_2x_4 - x_3^2 = 0$. 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 [52] is to compute a global minimizer of

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

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. Consider the homotopy

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

The following is immediate from coefficient-parameter continuation [44] showing that generic choices of parameter values (y, β) leads to a well-constructed homotopy $H_{y,\beta}$.

Proposition III.2. *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 $t \in (0, 1]$.*

The number of points in S is equal to the Euclidean distance degree of $\mathcal{V}(f - \beta)$ [25]. The set S can be computed using standard homotopy continuation as described in [36]. 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, tracking the paths starting at the points in S at $t = 1$ yields a finite set of solutions of $G_y = 0$ containing the global minimizer of (III.1) as shown in the following from [36, Thm. 5].

Theorem III.3. *Suppose that $y \in \mathbb{R}^N$ and $\beta \in \mathbb{C}^{N-d}$ such that the three items in Proposition III.2 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 (III.1). 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 (III.1) is identified by simply minimizing over these finitely many points.

IV. Generating Samples

This section presents an algorithm integrating Theorem III.3 with geometric tools to produce provably dense samples of real algebraic varieties. The input and output 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 $\hat{X} \subseteq \mathbb{R}^N$ that form a (δ, ϵ)-sample of $X \cap R$.

Proposition III.2 and Theorem III.3 provide a computationally tractable approach to finding very accurate estimated solutions of the optimization problem (III.1) 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 III.3 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 III.3, 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 $\texttt{MinDistance}(y)$ solves the problem (III.1) 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 $\texttt{MinDistance}(y)$ 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 $\texttt{MinDistance}(y)$. 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 3. 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 $\texttt{SplitBox}$ 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 $\texttt{SplitBox}$ into a tree structure.

Definition IV.1. For any box $A \subseteq \mathbb{R}^N$, let T_A be the tree with root A whose nodes are boxes in $\mathbb{R}^N$. The children of any box C in T_A are the elements of $\texttt{SplitBox}(C)$. The elements of $\texttt{SplitBox(C)}$ have parent node C.

For technical reasons, repeated applications of $\texttt{SplitBox}$ 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 $\texttt{SplitBox}$ 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 $\texttt{SplitBox}$ the sampling algorithm conducts a breadth first search of T_R while iteratively building the output sample.

Theorem IV.3. *Algorithm IV.2 terminates and outputs a (δ, ϵ)-sample of $X \cap R$.*

Before proving Theorem IV.3, we consider the following.

Algorithm IV.2 SAMPLING ALGORITHM

Input: Polynomial equations $f_1, \ldots, f_{N-d} \in \mathbb{R}[x_1, \ldots, x_N]$, a box $R = [a_1, b_1] \times \cdots \times [a_N, b_N]$, sampling density $\epsilon > 0$, and estimation error $0 \leqslant \delta \leqslant \epsilon$.
Output: A list of points which form a (δ, ϵ)-sample of $V_{\mathbb{R}}(f_1, \ldots, f_{N-d}) \cap R$.

1: Initialize an empty spatial database COVEREDREGIONS which can store and retrieve information 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 $\texttt{MinDistance(y)}$ where y is the center point of M, returning a set of sample points S with minimum distance d from y to any point in S.
6: Add regions $B_{d-\delta}(y)$ and $B_\epsilon(s)$ for each $s \in S$ to COVEREDREGIONS. Add each $s \in S$ to SAMPLEOUTPUT.
7: **end if**
8: **if** $M \subseteq B$ for any region B contained in COVEREDREGIONS **then**
9: Mark M and all nodes in the subtree rooted at M "done" and stop searching the subtree rooted at M.
10: **end if**
11: **if** All unsearched boxes in T_R are marked "done" **then**
12: End loop.
13: **end if**
14: **end for**
15: **return** SAMPLEOUTPUT

Lemma IV.4. *(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 4-6 of Algorithm IV.2, 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 as in Definition IV.1, 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 IV.3. (Termination): Let $\alpha = \frac{\epsilon - \delta}{\sqrt{N}}$. By our assumption on $\texttt{SplitBox}$ 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 4-6 of the algorithm will run if M is searched. Part (1) of Lemma IV.4 shows that line 9 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 IV.2 searches before terminating. By construction, T'_R fulfills the conditions of Lemma IV.4 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 III.3 and the definition of $\texttt{MinDistance}$ it follows that $X \cap (\cup_{y \in Y} B_{d-\delta}(y)) = \varnothing$. Similarly to Remark 3, 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 $\texttt{MinDistance}$. 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 $\texttt{MinDistance}$ step in Algorithm IV.2's core loop consumes significantly more time than any other individual step, so an optimal run of the Algorithm makes as few calls to $\texttt{MinDistance}$ as possible. Resource demands for processing the Algorithm's output with data analysis methods scale with the number of points in the sample. 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 $\texttt{MinDistance}$ calls and output sizes. These heuristics include:

- *Dynamic box splitting* - Instead of splitting along the longest side of a box B with $\texttt{SplitBox}$, 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 $\texttt{MinDistance}$ 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 $\texttt{MinDistance}$ 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.

V. EXAMPLES

Algorithm IV.2 has been implemented and used to produce dense samples of varieties for further processing via persistent homology. 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, regions of the persistence diagrams are highlighted according to Corollary 1. 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 $2(\epsilon + \delta)$.

A. 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 IV.2 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 Fig. 2 is derived from Corollary 1 assuming the homological feature size of the torus is at least $2(0.14 + 10^{-7})$. All points above and to the left of $(0.221, 0.56)$ are shaded.

B. Quartic surfaces

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

$$V_1 = \mathrm{V}_{\mathbb{R}}\left(\begin{array}{l} 4x^4 + 7y^4 + 3z^4 - 3 - 8x^3 \\ \quad + 2x^2y - 4x^2 - 8xy^2 - 5xy \\ \quad + 8x - 6y^3 + 8y^2 + 4y \end{array}\right),$$

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

Both quartic equations define pure 2-dimensional varieties. Figure 3 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.

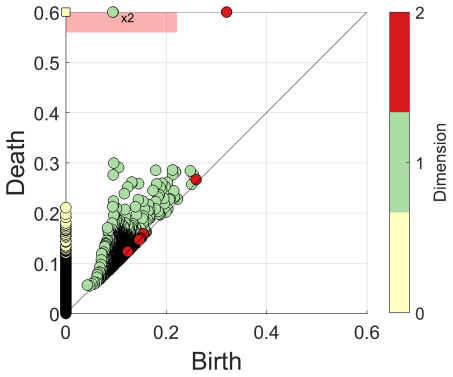

Fig. 2: PH results derived from sampling the Clifford torus. The sampling density is $(10^{-7}, 0.14)$. The estimated Betti numbers are $\beta_0 = 1$, $\beta_1 = 2$, and $\beta_2 = 1$.

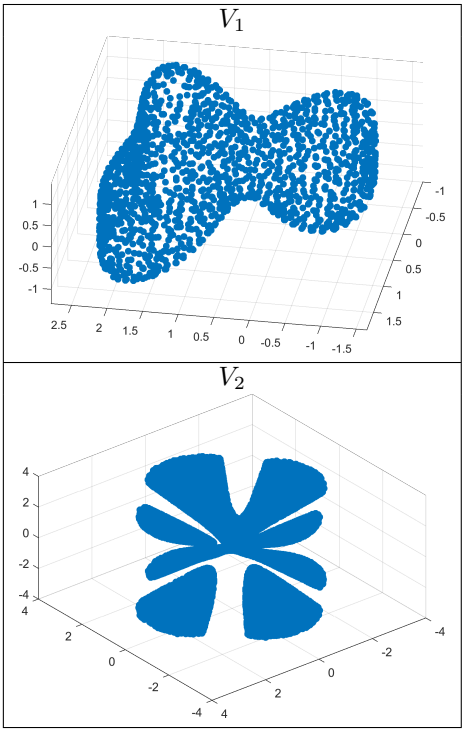

Fig. 3: Quartic surfaces sampled using Algorithm IV.2.

Samples produced for V_1 and V_2 contain 1,511 and 13,904 points respectively. The persistent homology results in Fig. 4 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 Fig. 5 are 5 connected components.

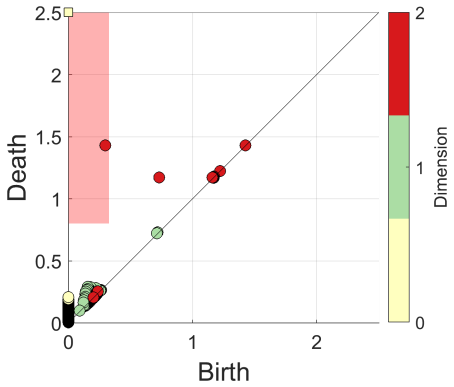

Fig. 4: PH results for V_1. The sampling density is $(10^{-7}, 0.20)$ and the estimated Betti numbers are $\beta_0 = 1, \beta_1 = 0, \beta_1 = 1$.

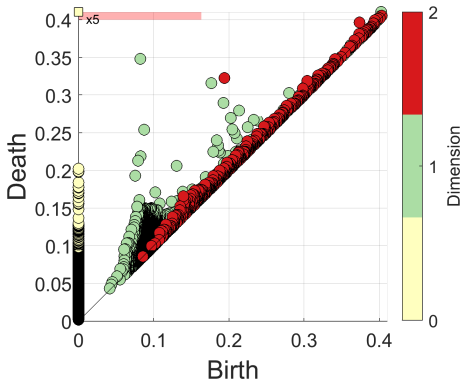

Fig. 5: PH results for V_2 thresholded to a parameter value of 0.405. The sampling density is $(10^{-7}, 0.10)$ and the estimated Betti numbers are $\beta_0 = 5, \beta_1 = 0$, and $\beta_2 = 0$.

C. 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.

A description of the polynomials defining V_p is presented in [11, §6.2.2] which is modelled as 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, namely:

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

A $(10^{-7}, 1.12)$ sample of V_p was produced by first obtaining a $(10^{-7}, 1.0)$ sample using Algorithm IV.2. 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 Fig. 6. The points far from the diagonal on the left hand side capture the theoretically expected homology for the configuration space.

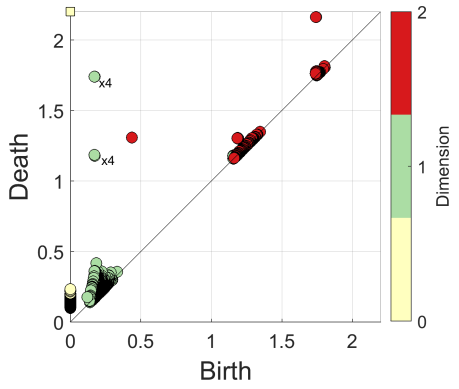

Fig. 6: Persistence diagram computed by sampling the configuration space of deformable pentagonal linkages. The sampling density is $(10^{-7}, 1.12)$.

VI. 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 from the stratification structure of singular varieties into systematic TDA based analysis. Running the PH pipeline on individual strata after identifying them via stratification methods for samples (for instance: [9]) or algebraic methods (detailed in [35]) would result in an even more detailed summary of the variety. Another direction is to apply persistent homology of ellipsoids rather than ϵ-balls [13].

Our work also raises the natural question of computationally estimating a lower bound on the weak feature size of varieties. Future work will explore how to exploit the algebraic description for this purpose. Finally, it would be worthwhile to investigate the noise induced from sampling via homotopy continuation in the context of off-set varieties [37].

Acknowledgements

ED and HAH were funded by the John Fell Oxford University Press (OUP) Research Fund. ED was also funded via an Anne McLaren fellowship from the University of Nottingham. PBE thanks P Bubenik for template scripts and B Sturmfels for the quartic equations. HAH was funded by EPSRC EP/K041096/1, EP/R005125/1, EP/R018472/1 and a Royal Society URF. 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] N. Amenta and M. Bern. Surface reconstruction by voronoi filtering. *Discrete & Computational Geometry*, 22(4):481–504, 1999.
[2] N. Amenta, M. Bern, and M. 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] P. Aubry, F. Rouillier, and M. Safey El Din. Real solving for positive dimensional systems. *J. Symbolic Comput.*, 34(6):543–560, 2002.
[4] S. 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] D.J. Bates, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler. *Numerically solving polynomial systems with Bertini*, volume 25. SIAM, 2013.
[6] U. Bauer. Ripser. https://github.com/Ripser/ripser, 2016.
[7] U. Bauer, M. Kerber, and J. Reininghaus. Dipha (a distributed persistent homology algorithm). Software available at https://github.com/DIPHA/dipha.
[8] U. Bauer, M. Kerber, and J. Reininghaus. Clear and compress: Computing persistent homology in chunks. In *Topological methods in data analysis and visualization III*, pages 103–117. Springer, 2014.
[9] P. Bendich, B. Wang, and S. 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] M. Berger, A. Tagliasacchi, L. Seversky, P. Alliez, J. Levine, A. Sharf, and C. Silva. State of the art in surface reconstruction from point clouds. In *EUROGRAPHICS star reports*, volume 1, pages 161–185, 2014.
[11] D.A. Brake, D.J. Bates, W. Hao, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler. Algorithm 976: Bertini_real: Numerical decomposition of real algebraic curves and surfaces. *ACM Trans. Math. Softw.*, 44(1):10, 2017.
[12] P. Breiding and O. Marigliano. Sampling from the uniform distribution on an algebraic manifold. arXiv preprint arXiv:1810.06271, 2018.
[13] P. Breiding, S. Kališnik, B. Sturmfels, and M. Weinstein. Learning algebraic varieties from samples. *Revista Matematica Complutense*, 31(3):545–593, 2018.
[14] P. Bubenik, V. De Silva, and J. Scott. Metrics for generalized persistence modules. *Foundations of Computational Mathematics*, 15(6):1501–1531, 2015.
[15] P. Bubenik and J.A. Scott. Categorification of persistent homology. *Disc. & Comput. Geom.*, 51(3):600–627, 2014.
[16] G. Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, 2009.
[17] F. Chazal, D. Cohen-Steiner, and A. Lieutier. A sampling theory for compact sets in euclidean space. *Discrete & Computational Geometry*, 41(3):461–479, 2009.
[18] F. Chazal and A. 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] F. Chazal and B. Michel. An introduction to topological data analysis: fundamental and practical aspects for data scientists. *arXiv preprint arXiv:1710.04019*, 2017.
[20] C. Chen and M. Kerber. Persistent homology computation with a twist. In *Proceedings 27th European Workshop on Computational Geometry*, volume 11, 2011.
[21] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37(1):103–120, 2007.
[22] F. Cucker, T. Krick, and M. Shub. Computing the homology of real projective sets. *Foundations of Computational Mathematics*, pages 1–42, 2016.
[23] V. De Silva and R. Ghrist. Coverage in sensor networks via persistent homology. *Algebraic & Geometric Topology*, 7(1):339–358, 2007.
[24] T. Krishna Dey, X. Ge, Q. Que, I. Safa, L. Wang, and Y. Wang. Feature-preserving reconstruction of singular surfaces. In *Computer Graphics Forum*, volume 31, pages 1787–1796. Wiley Online Library, 2012.
[25] J. Draisma, E. Horobeţ, G. Ottaviani, B. Sturmfels, and R. 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] H. Edelsbrunner and J. Harer. *Computational topology: an introduction*. AMS, 2010.
[27] P.B. Edwards. Topological data analysis for real algebraic varieties. Master's thesis, University of Oxford, 2016.
[28] M. Farber and D. Schütz. Homology of planar polygon spaces. *Geometriae Dedicata*, 125(1):75–92, 2007.
[29] D. Freedman. An incremental algorithm for reconstruction of surfaces of arbitrary codimension. *Computational Geometry*, 36(2):106–116, 2007.
[30] J.H.G. Fu. Tubular neighborhoods in Euclidean spaces. *Duke Mathematical Journal*, 52(4):1025–1046, 1985.
[31] R. Ghrist. Barcodes: the persistent topology of data. *Bulletin of the American Math. Society*, 45(1):61–75, 2008.
[32] E. Gross, H.A. Harrington, Z. Rosen, and B. Sturmfels. Algebraic Systems Biology: A Case Study for the Wnt Pathway. *Bulletin of Mathematical Biology*, 78(1):21–51, 2016.
[33] O. Hanner. Some theorems on absolute neighborhood retracts. *Ark. Mat.*, 1:389–408, 1951.
[34] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.
[35] J.D. Hauenstein and C.W. Wampler. Isosingular sets and deflation. *Foundations of Computational Mathematics*, 13(3):371–403, 2013.
[36] J.D. Hauenstein. Numerically computing real points on algebraic sets. *Acta Appl. Math.*, 125:105–119, 2013.
[37] E. Horobet and M. Weinstein. Offset hypersurfaces and persistent homology of algebraic varieties. *arXiv preprint arXiv:1803.07281*, 2018.
[38] F. Le Gall. Powers of tensors and fast matrix multiplication. In *ISSAC 2014—Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*, pages 296–303. ACM, New York, 2014.
[39] S. Lojasiewicz. Triangulation of semi-analytic sets. *Ann. Scuola Norm. Sup. Pisa Cl. Sci. (3)*, 18:449–474, 1964.
[40] S. Martin, A. Thompson, E.A. Coutsias, and J.-P. Watson. Topology of cyclo-octane energy landscape. *The journal of chemical physics*, 132(23):234115, 2010.
[41] S. Martin and J.-P. Watson. Non-manifold surface reconstruction from high-dimensional point cloud data. *Computational Geometry*, 44(8):427–441, 2011.
[42] R. Mendoza-Smith and J. Tanner. Parallel multi-scale reduction of persistent homology filtrations, 2017.
[43] N. Milosavljević, D. Morozov, and P. Skraba. Zigzag persistent homology in matrix multiplication time. In *Proceedings of the twenty-seventh annual symposium on computational geometry*, pp. 216–225, ACM, 2011.
[44] A.P. Morgan and A.J. Sommese. Coefficient-parameter polynomial continuation. *Appl. Math. Comput.*, 29(2, part II):123–160, 1989.
[45] B. Mourrain and J.P. Pavone. Subdivision methods for solving polynomial equations. *Journal of Symbolic Computation*, 44(3):292–306, 2009.
[46] M. Mustaţă. Graded betti numbers of general finite subsets of points on projective varieties. *Le Matematiche*, 53(3):53–81, 1998.
[47] P. Niyogi, S. Smale, and S. Weinberger. Finding the Homology of Submanifolds with High Confidence from Random Samples. *Discrete & Computational Geometry*, 39(1-3):419–441, 2008.
[48] N. Otter, M.A. Porter, U. Tillmann, P. Grindrod, and H.A. Harrington. A roadmap for the computation of persistent homology. *EPJ Data Science*, 6(1):17, 2017.
[49] S.Y. Oudot. *Persistence theory: from quiver representations to data analysis*, volume 209. American Mathematical Society, 2015.
[50] F. Rouillier, M.-F. Roy, and M. 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.
[51] P. Scheiblechner. On the complexity of deciding connectedness and computing betti numbers of a complex algebraic variety. *Journal of Complexity*, 23(3):359–379, 2007.
[52] A. Seidenberg. A new decision method for elementary algebra. *Ann. of Math. (2)*, 60:365–374, 1954.
[53] E.C. Sherbrooke and N.M. Patrikalakis. Computation of the solutions of nonlinear polynomial systems. *Computer Aided Geometric Design*, 10(5):379–405, 1993.
[54] A. Sommese and C. Wampler. *The Numerical solution of systems of polynomials arising in engineering and science*, volume 99. World Scientific, 2005.
[55] A. Zomorodian and G. Carlsson. Computing persistent homology. *Disc. & Comput. Geom.*, 33(2):249–274, 2005.