Point Clouds of Varieties and Persistent Homology

Mikael Vejdemo-Johansson (St. Andrews)
Jon Hauenstein (Texas A&M)
David Eklund (KTH)
Martina Scolamiero (KTH)
Chris Peterson (Colorado State)
Primoz Skraba (Jožef Stefan Institute)

24 October 2011

Running problem

Given a system of polynomial equations, what is the shape of the set of solutions?

In particular, several applications care about the *topology* of the solution set.

Example

Positions of a robotic arm are described by polynomial equations.

- If the shape of positions is disconnected, the robot arm cannot move between all its positions without dismantling
- If the shape of positions has non-trivial loops, this influences motion planning problems.

Mathematical framework

Algebraic Geometry

Provides tools for analyzing the shape of solutions.

Algebraic Topology

Provides tools for understanding connectivity, loops, and higher-dimensional analogues.

Turns out to be hard to determine topological features using algebraic geometry data. There are Gröbner basis methods, but very high complexity and only feasible in very low dimensions.

Point cloud computation

Algebraic geometry

Numerical methods generate points on varieties. Repeat many times yields point clouds on varieties.

- Homotopy continuation
- Gradient descent methods

Algebraic topology

Persistence techniques estimate topological features from point clouds.

- Persistent Betti numbers
- Representatives

Homotopy continuation

Problem

Given a system of polynomial equations $\{f_k(\bar{x}) = 0\}$, compute all isolated points in the solution set.

Solution

Start out with a system $\{g_k(\bar{x}) = 0\}$ where each $g_k = \bar{x}^{\deg f_k}$. We can write down solutions for the system g_k directly.
For small steps in t, solve the system $\{tf_k + (1-t)g_k = 0\}$ using the previous solutions as starting guesses.

Turns out you can *almost always* do this without running into problems along the way.

Homology

A core technique from algebraic topology is *homology*. It translates certain geometric properties into linear algebra:

$$\begin{aligned} \text{triangulation} &\Leftrightarrow \text{vector space} \\ \text{boundaries} &\Leftrightarrow \text{linear map} \\ \text{closed curves} &\Leftrightarrow \text{nullspace} \\ \text{equivalence classes of holes/bubbles} &\Leftrightarrow \text{nullspace}/\text{image} \end{aligned}$$

These definitions generalize to higher dimensions. We write H_k for the quotient nullspace/image; and β_k for dim H_k.

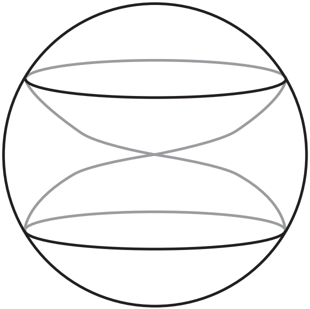
Worked example: Cyclo-octane configurations

Working to replicate an analysis in *Topology of cyclo-octane energy landscape* (Martin, Thompson, Coutsias, Watson; J. Chem. Phys. **132**, 234115 (2010)), we consider cyclo-octane as a linkage. Martin-Thompson-Coutsias-Watson established the topology of this to be a sphere and a Klein bottle, fused along two circles. They also computed the Betti numbers to be $\beta_0 = 1$, $\beta_1 = 1$, and $\beta_2 = 2$.

Requiring rest-state distances between atoms, and rest-state planar angles for carbon-carbon bonds, the resulting linkage has only rotational joints at each carbon atom.

We sampled 34k points from the resulting variety, and are using this as a test-case for a systematic method for the computation.

Worked example: Cyclo-octane configurations



- Sphere
- Klein bottle
- Intersect in disjoint, unlinked pair of circles.

Outline

1. Point clouds

2. Recovery methods

Generating point clouds on varieties

Homotopy methods yield points of varieties by numerical solution of the defining equations.

Explicit implementation in `Bertini`, by Dan Bates, Jon Hauenstein, Andrew Sommese and Charles Wampler.

By intersecting variety with random hyperplanes, sample points on complementary dimensional components can be guaranteed.

Most important fact here

We can generate point clouds from varieties.

First failures

Our first attempts at explicit computation failed for several reasons:

- Sampling method concentrated in high-curvature regions.
- Computation on large point-clouds a memory hog.

These different failure modes have inspired further work.

Outline

Improved sampling

The most effective recovery was an improved sampling technique.

Current technique

1. Generate $\sim 1\,000$ points with old sampling technique.
2. Compute PCA of this point sample.
3. Generate a grid with density weighted by the PCA coefficients; map the grid by the PCA reconstruction matrix.
4. Seeding with grid points, use numerics to find points on the variety.
5. For the point cloud pick the solutions that are close to their seed points.

This yields an ε-dense point cloud. A witness complex with 150 landmarks and $25\,000$ data points recovered the Betti numbers of the cyclo-octane data set.

Interleaved persistence

Before we were able to fix the sampling techniques, we considered ways to make the persistence computation consume less memory.

Based on the implementation in `jPlex`, one approach suggested itself:

`jPlex`

1. Generate the Vietoris-Rips complex.
2. Perform the Persistence algorithm

The entire simplex stream is generated before a single persistence interval is computed.

Improvement

Interleave the computation: generate new simplices at the addition of any new edge of the Rips graph.
Consume simplices for the persistence computation immediately on generation.

Parallelizing persistence

Finally, we could split up the memory consumption between different agents.

Mayer-Vietoris

Compute homology on U, V and on $U \cap V$.
Merge to give homology on $U \cup V$.

Spectral sequences

There is a spectral sequence that computes homology of $\bigcup_i U_i$.

In the variety case, a decomposition suggests itself.

Decompose the variety

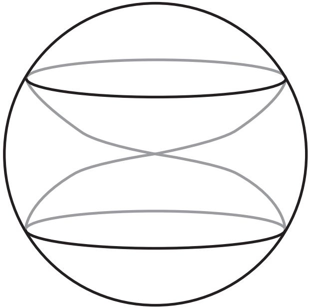
We can easily compute both $V(f)$ and $V_{Sing}(f)$.

If the singularities have codimension 1, they decompose the smooth points into several components.

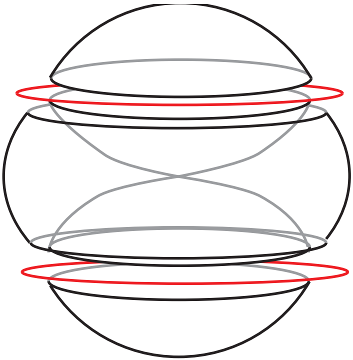
- Compute points on $V(f)$ and on $V_{Sing}(f)$. Let $C = \{x \in V(f), d(x, V_{Sing}(f)) < \varepsilon\}$.
- Cluster the points in $V(f) \setminus C$.
- For each cluster X_i from above:
    - Cluster $C \cup X_i$.
    - Form patches from clusters of $C \cup X_i$ that do not include into C.
- Compute homology on each patch, and each patch intersection. Patch intersections are all contained in C by construction.

This data suffices to reconstruct $H_*(V(f))$.

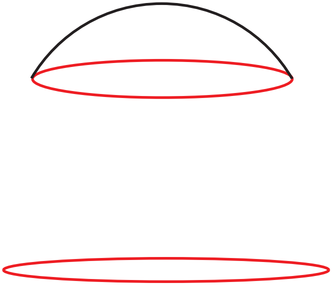
Decomposing the space



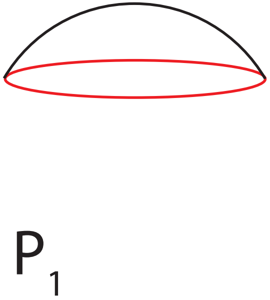
Decomposing the space



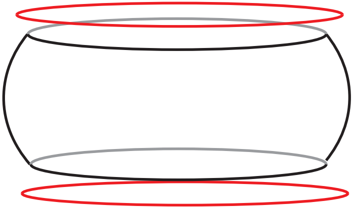
Decomposing the space



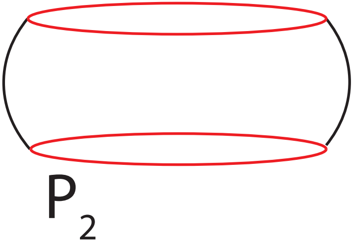
Decomposing the space



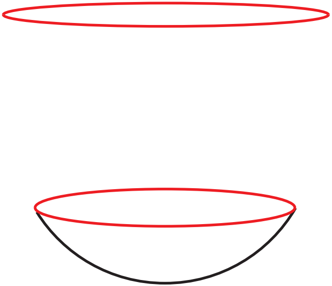
Decomposing the space



Decomposing the space

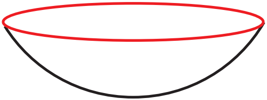


Decomposing the space

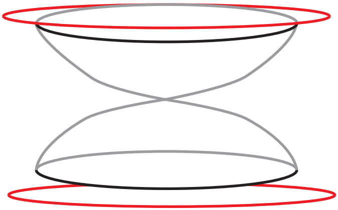


Decomposing the space

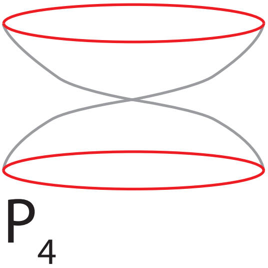
P_3



Decomposing the space



Decomposing the space



Mayer-Vietoris spectral sequence

E^0:

$$\begin{array}{ccccc}
\bigoplus_{i,j,k} C_2 P_i \cap P_j \cap P_k & & \bigoplus_{i,j} C_2 P_i \cap P_j & & \bigoplus_i C_2 P_i \\
\downarrow & & \downarrow & & \downarrow \\
\bigoplus_{i,j,k} C_1 P_i \cap P_j \cap P_k & & \bigoplus_{i,j} C_1 P_i \cap P_j & & \bigoplus_i C_1 P_i \\
\downarrow & & \downarrow & & \downarrow \\
\bigoplus_{i,j,k} C_0 P_i \cap P_j \cap P_k & & \bigoplus_{i,j} C_0 P_i \cap P_j & & \bigoplus_i C_0 P_i
\end{array}$$

Mayer-Vietoris spectral sequence

E^1:

$$\bigoplus_{i,j,k} H_2 P_i \cap P_j \cap P_k \longrightarrow \bigoplus_{i,j} H_2 P_i \cap P_j \longrightarrow \bigoplus_i H_2 P_i$$

$$\bigoplus_{i,j,k} H_1 P_i \cap P_j \cap P_k \longrightarrow \bigoplus_{i,j} H_1 P_i \cap P_j \longrightarrow \bigoplus_i H_1 P_i$$

$$\bigoplus_{i,j,k} H_0 P_i \cap P_j \cap P_k \longrightarrow \bigoplus_{i,j} H_0 P_i \cap P_j \longrightarrow \bigoplus_i H_0 P_i$$

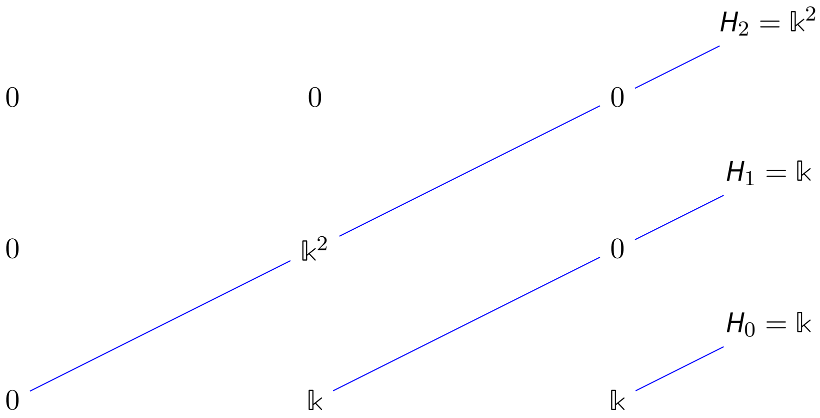
Mayer-Vietoris spectral sequence

E^2:

0	0	0
0	$\Bbbk^2$	0
0	$\Bbbk$	$\Bbbk$

Mayer-Vietoris spectral sequence

E^2:



Questions?

In summary:

- Numerical algebraic geometry produces point clouds from varieties.
- Persistent homology produces Betti number estimates from point clouds.
- Computations end up being large and difficult.
- Classical topology techniques might save the day.