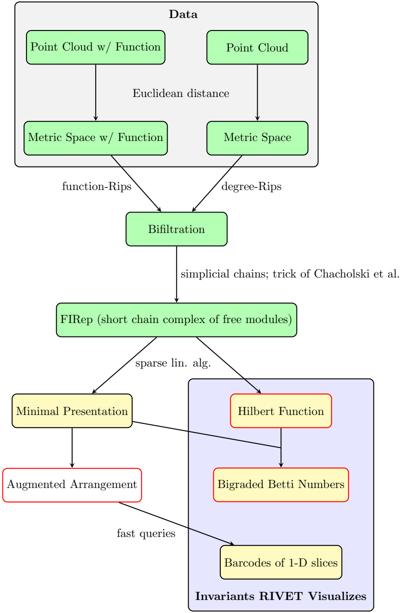
Computing Minimal Presentations of Bipersistence Modules in Cubic Time

Michael Lesnick
joint work w/ Matthew Wright

IMA Tutorial on Multiparameter Persistence, Computation, and Applications,
August 14, 2018

- Implemented in RIVET.
- A descendant of an earlier algorithm by Matthew and me for computing bigraded Betti numbers.
- Yields Hilbert Function (i.e., dimension function) and bigraded Betti numbers as a biproduct.

RIVET Computational Pipeline



Free bipersistence modules:

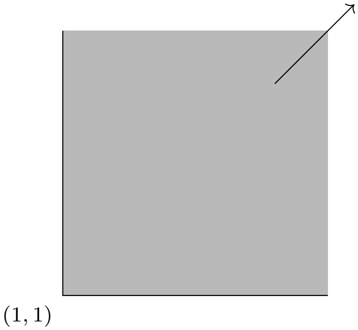
- Needed to define presentations, resolutions, Betti numbers,
- Algebra of these looks a lot like ordinary linear algebra!

For $c \in \mathbb{Z}^2$, define the **quadrant** bipersistence module $\mathcal{Q}^c$ by

$$\mathcal{Q}^c_a = \begin{cases} K & \text{if } a \geq c, \\ 0 & \text{otherwise.} \end{cases} \qquad \mathcal{Q}^c_{a,b} = \begin{cases} \mathrm{Id}_K & \text{if } a \geq c, \\ 0 & \text{otherwise.} \end{cases}$$

$Q^{(1,1)}$:

$$\begin{array}{ccccccccc}
\vdots & & \vdots & & \vdots & & \vdots & & \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \longrightarrow & k & \longrightarrow & k & \longrightarrow & k & \longrightarrow & \cdots \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \longrightarrow & k & \longrightarrow & k & \longrightarrow & k & \longrightarrow & \cdots \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \longrightarrow & k & \longrightarrow & k & \longrightarrow & k & \longrightarrow & \cdots \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \longrightarrow & 0 & \longrightarrow & 0 & \longrightarrow & 0 & \longrightarrow & \cdots
\end{array}$$

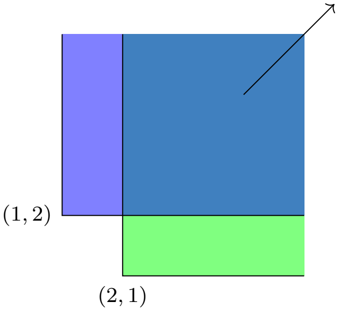


Exercise: For which $y, z \in \mathbb{Z}^2$ does there exists a non-zero morphism $\mathcal{Q}^y \to \mathcal{Q}^z$?

A **free** module is one isomorphic to a direct sum of quadrant modules.

$Q^{(2,1)} \oplus Q^{(1,2)}$:

$$
\begin{array}{ccccccccc}
\vdots & & \vdots & & \vdots & & \vdots & & \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \longrightarrow & k & \longrightarrow & k^2 & \longrightarrow & k^2 & \longrightarrow & \cdots \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \longrightarrow & k & \xrightarrow{\binom{1}{0}} & k^2 & \longrightarrow & k^2 & \longrightarrow & \cdots \\
\uparrow & & \uparrow & & \uparrow{\scriptstyle\binom{0}{1}} & & \uparrow & & \\
0 & \longrightarrow & 0 & \longrightarrow & k & \longrightarrow & k & \longrightarrow & \cdots \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \longrightarrow & 0 & \longrightarrow & 0 & \longrightarrow & 0 & \longrightarrow & \cdots
\end{array}
$$



(Maps between equal vector spaces are identities).

Bases of Free modules

Let M be any bipersistence module, G be a set of vectors in M.

We say G **generates** M if every vector in M is a linear combination of elements of G.

Bases of Free modules

Let M be any bipersistence module, G be a set of vectors in M.

We say G **generates** M if every vector in M is a linear combination of elements of G.

A **basis** for a free module F is a minimal generating set.

$$
\begin{array}{ccccccccc}
\vdots & & \vdots & & \vdots & & \vdots & & \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \to & k & \to & k^2 & \to & k^2 & \to & \cdots \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \to & k & \xrightarrow{\binom{1}{0}} & k^2 & \to & k^2 & \to & \cdots \\
\uparrow & & \uparrow & & \uparrow{\scriptstyle\binom{0}{1}} & & \uparrow & & \\
0 & \to & 0 & \longrightarrow & k & \longrightarrow & k^2 & \to & \cdots \\
\uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
0 & \to & 0 & \longrightarrow & 0 & \longrightarrow & 0 & \to & \cdots
\end{array}
$$

Example: For $Q^{(2,1)} \oplus Q^{(1,2)}$,

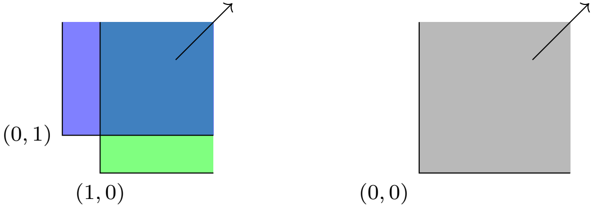
- $\{1^{(2,1)}, 1^{(1,2)}\}$ is a basis.
- $\{1^{(2,1)}, 1^{(1,2)}, (1,1)^{(2,2)}\}$ is a generating set, not a basis.
- $\{1^{(2,1)}\}$ is not a generating set.

Example: Let's consider the map

$$f : \mathcal{Q}^{(1,0)} \oplus \mathcal{Q}^{(0,1)} \to \mathcal{Q}^{(0,0)}$$

which restricts to the inclusion on each summand.

What are its kernel and cokernel?



A **submodule** M of a bipersistence module of N is a collection of subspaces

$$\{M_z \subset N_z\}_{z \in \mathbb{Z}^2}$$

such that $N_{y,z}(M_y) \subset M_z$ for all $y \leq z$.

E.g., **images** and and **kernels** are well defined submodules.

Example (continued): For

$$f : \mathcal{Q}^{(1,0)} \oplus \mathcal{Q}^{(0,1)} \to \mathcal{Q}^{(0,0)}$$

as before,

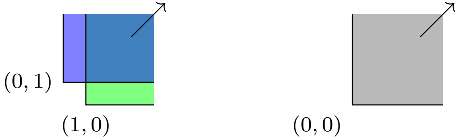
$$\ker(f) \cong Q^{(1,1)}.$$

$(1, -1) \subset M_{(1,1)} = k^2$ is a generator.

Explanation : For $z \geq (1,1)$, $f_z : k^2 \to k$ is given by

$$f_z(x, y) = x + y.$$

For $z \not\geq (1,1)$, $f_z : k \to k$ is the identity.



Kernels of Maps of Free Bipersistence Modules

Essential Fact: The kernel of a morphism of free finitely generated 2-parameter persistence modules is free.

Exercise:

(i) Prove this, using stuff later in the next few slides.

(ii) Give a counterexample for three parameters.

Quotient Persistence Modules

Linear algebra fact: Given vector spaces $V' \subset V$, $W' \subset W$ and a morphism $f : V \to W$ with $f(V') \subset W'$, we get an induced map

$$\bar{f} : V/V' \to W/W'.$$

Quotient Persistence Modules

Linear algebra fact: Given vector spaces $V' \subset V$, $W' \subset W$ and a morphism $f : V \to W$ with $f(V') \subset W'$, we get an induced map

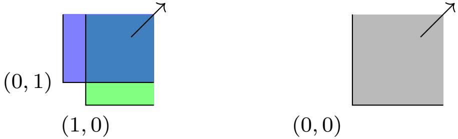
$$\bar{f} : V/V' \to W/W'.$$

For bipersistence modules $M \subset N$, define the **quotient** N/M by taking quotients at each index.

Example (continued): Define the persistence module $\mathbf{k}$ by:

$$\mathbf{k}_z := \begin{cases} k & \text{if } z = (0,0), \\ 0 & \text{otherwise.} \end{cases}$$

For f as before, coker $f \cong \mathbf{k}$.



A **free resolution** of a persistence module M is an exact sequence of free modules

$$\cdots \xrightarrow{\partial_3} F_2 \xrightarrow{\partial_2} F_1 \xrightarrow{\partial_1} F_0$$

with $M \cong \operatorname{coker} \partial_1$.

Theorem: If M is finitely generated, then $\exists$ a **minimal** free resolution $F_\bullet$.

Minimality = Any other is isomorphic to one obtained from $F_\bullet$ by summing with free resolutions of the form

$$\cdots \to 0 \to 0 \to G \xrightarrow{\mathrm{Id_G}} G \to 0 \to 0 \to \cdots \to 0$$

By Krull-Schmidt, the minimal resolution is unique up to iso.

Example: A minimal resolution for $\mathbf{k}$:

$$0 \to \mathcal{Q}^{(1,1)} \hookrightarrow \mathcal{Q}^{(1,0)} \oplus \mathcal{Q}^{(0,1)} \xrightarrow{f} \mathcal{Q}^{(0,0)}.$$

Presentations

A **presentation** of a bipersistence module M is a morphism of free modules $\partial_1 : F_1 \to F_0$ with $M \cong F_0/\operatorname{im}\gamma$.

A **minimal presentation** is one such that any other can be obtained, up to iso, by adding summands of the form:

$$G \xrightarrow{\mathrm{id_G}} G \qquad \text{or} \qquad G \to 0.$$

Bases of Free modules

Recall: A **basis** for a free module F is a minimal generating set.

Let $\xi(F)$ denote the multiset of bigrades of elements in any basis for F.

For M finitely generated and $i \geq 0$, define

$$\xi_i^M := \xi(F_i),$$

where F_i is the i^{th} module in any minimal resolution for M.

We call ξ_i^M the i^{th} **bigraded Betti number** of M.

Example: A minimal resolution for $\mathbf{k}$:

$$0 \to \mathcal{Q}^{(1,1)} \hookrightarrow \mathcal{Q}^{(1,0)} \oplus \mathcal{Q}^{(0,1)} \xrightarrow{f} \mathcal{Q}^{(0,0)}$$

$$\begin{aligned}
\xi_0^{\mathbf{k}} &= \{(0,0)\}, \\
\xi_1^{\mathbf{k}} &= \{(1,0),(0,1)\}, \\
\xi_2^{\mathbf{k}} &= \{(1,1)\}.
\end{aligned}$$

Hilbert′s Syzygy Theorem: For M a fin. gen. bipersistence module, $\xi_i^M = \emptyset$ for $i > 2$.

The Matrix Algebra of Free Modules

Fix ordered bases B, B' for finitely generated free modules M and N of sizes b and b'.

A vector $v \in M_z$ is represented w.r.t. B as an element $[v]_B \in k^b$.

- $[v]_B$ records coefficients in the linear combination of B giving v.

The Matrix Algebra of Free Modules

Fix ordered bases B, B' for finitely generated free modules M and N of sizes b and b'.

A vector $v \in M_z$ is represented w.r.t. B as an element $[v]_B \in k^b$.

- $[v]_B$ records coefficients in the linear combination of B giving v.

A map $\gamma : M \to N$ can be represented as a $b' \times b$ matrix $[\gamma]_{B,B'}$, with each row and each column labelled by an element of $\mathbb{Z}^2$:

- The i^{th} column of $[\gamma]_{B,B'}$ is $[\gamma(B_i)]_{B'}$.
- label the i^{th} column by the bigrade of the i^{th} element of B.
- label the i^{th} row by the bigrade of the i^{th} element of B'.

Fact: $[\gamma]_{B,B'}$ encodes the isomorphism type of γ.

In particular, $[\gamma]_{B,B'}$ encodes $\operatorname{coker} \gamma$ up to iso.

When interpreting γ as a presentation, I'll call $[\gamma]_{B,B'}$ a **concrete presentation**.

computing minimal presentations

Problem Specification:

Input: **FIRep** for a bipersistence module M (Short chain complex of free modules).

Output: Concrete minimal presentation for M (a matrix w/ bigrade labels for both rows and columns).

A **free implicit representation** (**FIRep**) for a bipersistence module M is a chain complex of free modules:

$$C \xrightarrow{f} D \xrightarrow{g} E.$$

with $\ker g / \operatorname{im} f \cong M$.

Given ordered bases B_C, B_D, B_E for C, D, and E, we can represent the FIRep up to isomorphism by the labeled matrices

$$[f]_{B_C,B_D} \quad \text{and} \quad [g]_{B_D,B_E}.$$

In fact, it's enough to keep only the column labels:

- We don't need row labels for $[g]_{B_D,B_E}$ to compute a presentation for M.
- Row labels for $[f]_{B_C,B_D}$ are same as column labels for $[g]_{B_D,B_E}$.

Complexity of our minimal presentation computation:

- $O(n^3)$, where n is the maximum number of columns or rows in either matrix
- $O(n^2)$ storage.

We use a column-sparse implementation which performs very well in practice.

NOTE: Computing minimal presentations is common in commutative algebra.

- Algorithms are not optimized for types of input we encounter in TDA.
- working in the 2-parameter setting with bigraded modules allows for simplificiations.
- To the best of our knowledge, ours is the first $O(n^3)$ algorithm.

See also work of Jacek Skryzalin, which addresses related things.

FIRep from a Bifiltration

Given a bifiltration F, we get a chain complex of bipersistence modules $C_\bullet(F) =$

$$\cdots \xrightarrow{\partial_{i+1}} C_j(F) \xrightarrow{\partial_i} C_{i-1}(F) \xrightarrow{\partial_{i-1}} \cdots \xrightarrow{\partial_1} C_0(F) \to 0$$

with $C_j(F)_z := C_j(F_z)$.

We say a bifiltration F is **1-critical** if each simplex in $\bigcup_{z\in\mathbb{Z}^2} F_z$ has a unique bigrade of appearance.

Otherwise, we say F is **multicritical**.

If F is 1-critical, then each $C_i(F)$ is free with basis the j-simplices, and so

$$C_{i+1}(F) \xrightarrow{\partial^{i+1}} C_j(F) \xrightarrow{\partial^i} C_{i-1}(F)$$

is an FIRep for $H_i(F)$.

Algorithm for computing a (non – minimal) presentation
Given: FIRep

$$C \xrightarrow{f} D \xrightarrow{g} E$$

1. Using $[f]_{B_C,B_D}$ as input, find a minimal ordered set of generators S for $\operatorname{im} f$.
2. Compute a basis B_{ker} for $\ker g$.
3. Express each element of S in B_{ker}-coordinates; put resulting column vectors into a matrix P, with column labels the bigrades of S and row labels the bigrades of B_{ker}.

Remarks:

(i) The algorithms for steps 1 and 2 are similar.

(ii) Step 1 ensures a presentation where extra summand is of the form

$$G \xrightarrow{\mathrm{Id}_G} G$$

To illustrate the ideas, we'll focus on kernel computation.

Computing Kernels in the 1-Parameter Case

Let

- $\gamma : M \to N$ be a map of free monopersistence modules,
- B, B' be ordered bases for M and N, with B in order of increasing grade.

Input: Matrix $R := [\gamma]_{B,B'}$, with columns labeled by grades of B.
Output: Basis for $\ker \gamma$ (represented w.r.t. basis B):

Algorithm:

- Run persistence algorithm on R with a "slave matrix" V.
- For each column j zeroed out in R, add the vector in $M_{\text{label}(j)}$ represented by the j^{th} column of V to the basis for $\ker \gamma$.

Computing Kernels in the 2-Parameter Case

Let

- $f : M \to N$ be a map of free bipersistence modules.
- B, B' be ordered bases for M and N, with B in **colexicographical order** on the bigrades.

Input: Matrix $R := [f]_{B,B'}$, with the columns labeled by grades of B.
Output: Basis for $\ker f$ (represented with respect to the basis B):

Let R^z denote the submatrix consisting of columns with label $\leq z$.

Algorithm: For each z in **lexicographical order**:

1. Run persistence algorithm on R^z.
2. Also perform each column operation on a slave matrix V
3. If column j gets zeroed out, add the vector in M_z represented by the j^{th} column of V to the basis for $\ker f$.

minimizing the presentation

Let P be a concrete presentation with all non-minimal summands are of the form $G \xrightarrow{\mathrm{Id}_G} G$.

To minimize the presentation P:

For each column i:

1. check if the label of the column i is equal to the row-label of the pivot.
2, If so, zero out the row of the pivot by adding column i to columns of higher index.
3. If so, remove both the column i and the row of the pivot.

Remarks:

- This is embarrassingly parallel.
- This is a bigraded, column-sparse variant of a standard procedure for minimizing a presentation.

Future Work

- Most or all of the optimizations of PHAT/Ripser seem to apply to computing minimal presentations.
- None our implemented yet, except for lazy heaps.
- Next step: Clearing.

Future Work

- Most or all of the optimizations of PHAT/Ripser seem to apply to computing minimal presentations.
- None our implemented yet, except for lazy heaps.
- Next step: Clearing.

Thank you!!

Download RIVET and try out our minimal presentations code at http://rivet.online