Spectral Sequences for Applied Topology
Thesis Defense

Ryan H. Lewis

Advisor: Gunnar Carlsson

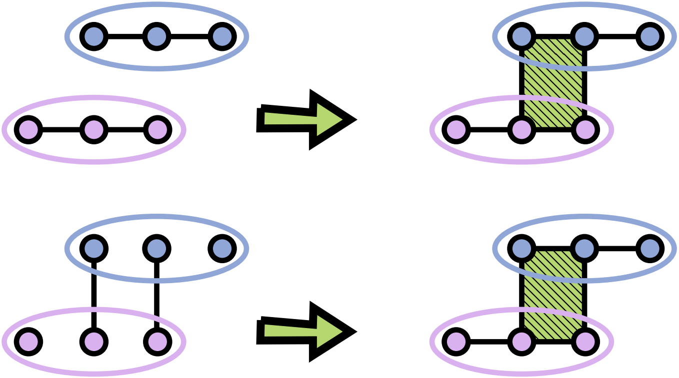
May 19, 2016

The Agenda

Today we will present distributed algorithms for homology and persistent homology, via spectral sequences.

- Motivation
- **Background**
- Persistent Homology an Sequence
- Homology from subcom Vietoris
- A distributed persistenc

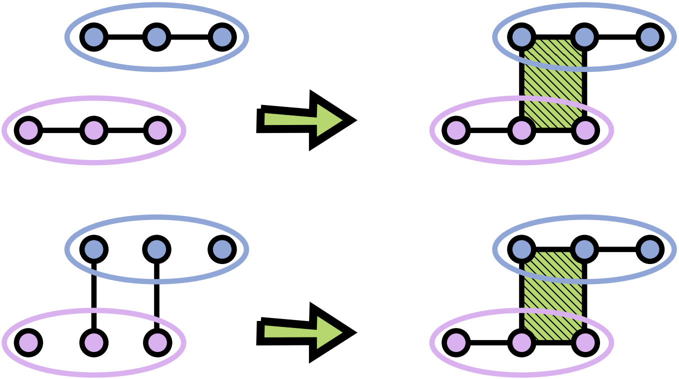
Filtrations



Recall: K^U has cells of the form $\sigma \times \tau$ with $\sigma \in K$ and $\tau \in N$

Difference is about ordering on cells: $\sigma \times \tau < \sigma' \times \tau'$

Filtrations

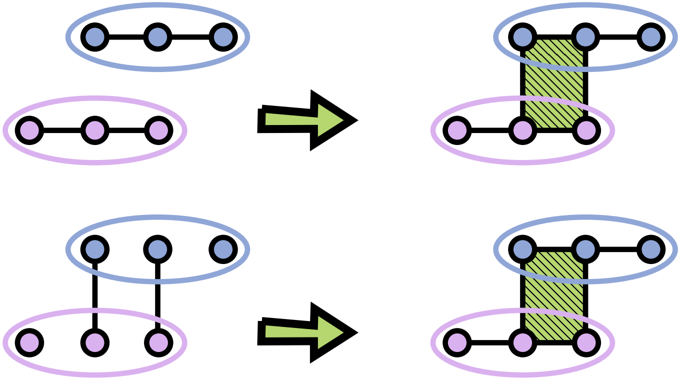


Recall: K^U has cells of the form $\sigma \times \tau$ with $\sigma \in K$ and $\tau \in N$

Difference is about ordering on cells: $\sigma \times \tau < \sigma' \times \tau'$

Picture 1: $\tau < \tau'$ then breaking ties by comparing $\sigma < \sigma'$.

Filtrations



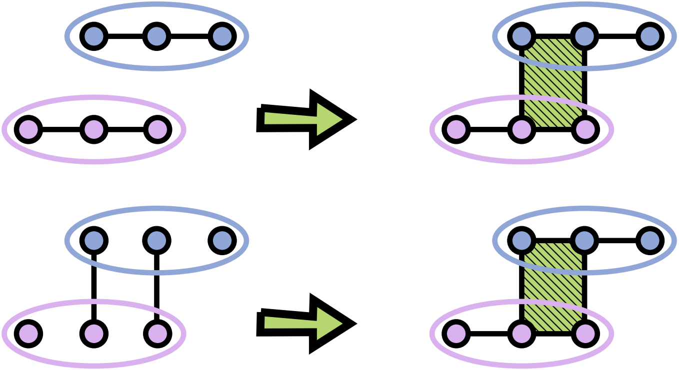
Recall: K^U has cells of the form $\sigma \times \tau$ with $\sigma \in K$ and $\tau \in N$

Difference is about ordering on cells: $\sigma \times \tau < \sigma' \times \tau'$

Picture 1: $\tau < \tau'$ then breaking ties by comparing $\sigma < \sigma'$.

Picture 2: $\sigma < \sigma'$ then breaking ties by comparing $\tau < \tau'$

Filtrations



Recall: K^U has cells of the form $\sigma \times \tau$ with $\sigma \in K$ and $\tau \in N$

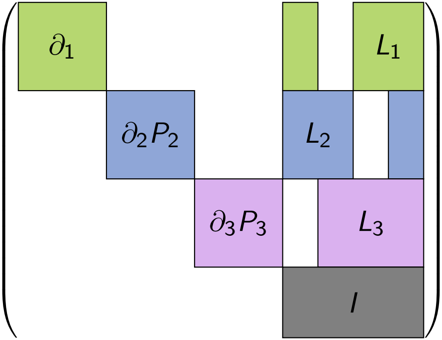
Difference is about ordering on cells: $\sigma \times \tau < \sigma' \times \tau'$

Picture 1: $\tau < \tau'$ then breaking ties by comparing $\sigma < \sigma'$.

Picture 2: $\sigma < \sigma'$ then breaking ties by comparing $\tau < \tau'$

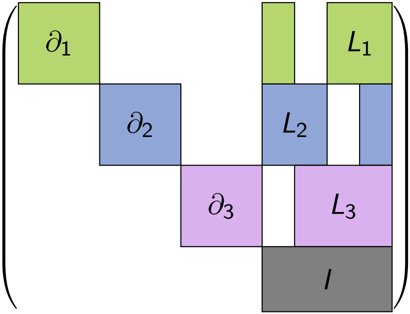
Important Observation: If $\tau = \tau'$ then orders agree!

revisit the blowup



1. ∂_{K^U} in block form, according to incorrect filtration.
2. Reducing the matrix $\Pi' \cdot \partial_{K^U} \cdot \Pi$, where Π permutes between filtrations, results in the correct persistent homology.

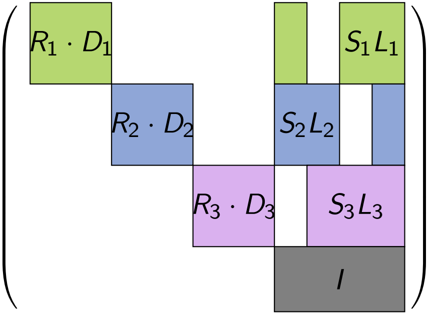
derive persistence with mayer vietoris.



All processors execute these operations with no communication!

Step 1 Reduce all blocks (except I) of a fixed color independently.
$\partial_i = R_i \cdot D_i$

derive persistence with mayer vietoris.

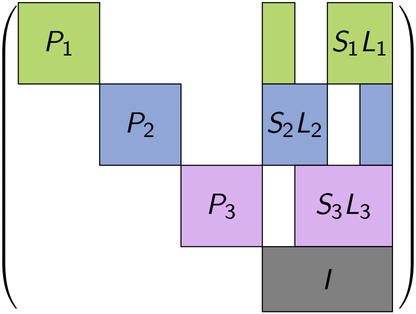


All processors execute these operations with no communication!

Step 1 Reduce all blocks (except I) of a fixed color independently.
$\partial_i = R_i \cdot D_i$

Step 2 Row reduce disjoint union $S^{-1}[R_i \mid L_i] \to [P_i \mid \tilde{L}_i]$

derive persistence with mayer vietoris.



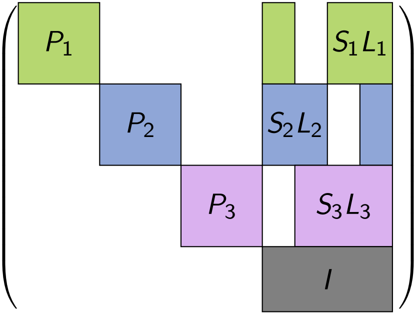
All processors execute these operations with no communication!

Step 1 Reduce all blocks (except I) of a fixed color independently.
$\partial_i = R_i \cdot D_i$

Step 2 Row reduce disjoint union $S^{-1}[R_i \mid L_i] \to [P_i \mid \tilde{L}_i]$

Step 3 Perform all **valid** columns adds from P_i into $\tilde{L}_i$ (perfect parallelism).

derive persistence with mayer vietoris.



All processors execute these operations with no communication!

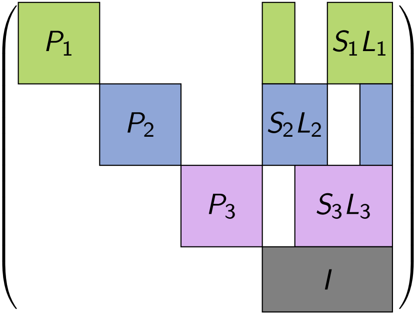
Step 1 Reduce all blocks (except **I**) of a fixed color independently.
$\partial_i = R_i \cdot D_i$

Step 2 Row reduce disjoint union $S^{-1}[R_i \mid L_i] \to [P_i \mid \tilde{L}_i]$

Step 3 Perform all **valid** columns adds from P_i into $\tilde{L}_i$ (perfect parallelism).

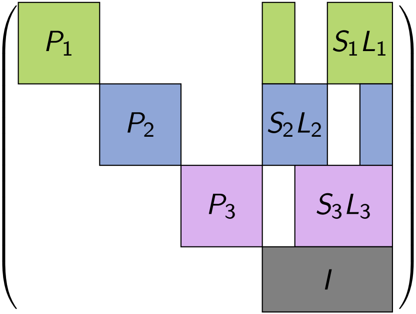
Sofar Not finished yet, but, have not done anything wrong.

derive persistence with mayer vietoris.



Sparsity: Now each P_i has at most 1 nonzero per column.

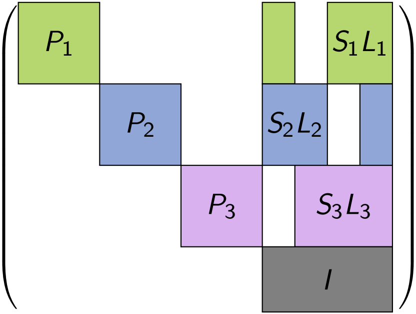
derive persistence with mayer vietoris.



Sparsity: Now each P_i has at most 1 nonzero per column.

Fill in: If $D.U.$ has size m and total size is $m + n$ then fill in is at most $O(mn)$ down from $O((m+n)^2)$

derive persistence with mayer vietoris.

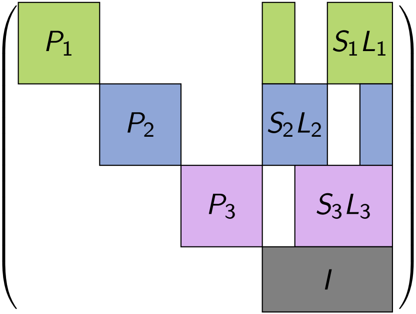


Sparsity: Now each P_i has at most 1 nonzero per column.

Fill in: If $D.U.$ has size m and total size is $m + n$ then fill in is at most $O(mn)$ down from $O((m + n)^2)$

Issue: After permuting, naive reduction, could fill in entire matrix.

derive persistence with mayer vietoris.



Sparsity: Now each P_i has at most 1 nonzero per column.

Fill in: If $D.U.$ has size m and total size is $m+n$ then fill in is at most $O(mn)$ down from $O((m+n)^2)$

Issue: After permuting, naive reduction, could fill in entire matrix.

Solution: Row operations to the rescue!

Cascade

All columns with pivot in row m (after Π):

$$
\begin{pmatrix}
\times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\
\times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\
\times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\
\times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\
\times & \times & \times & \times & \times & \times & \times & \times & \times
\end{pmatrix}
$$

Cascade

All columns with pivot in row m (after Π):

$$\begin{pmatrix} \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & \times & \times & \times & \times & \times & \times & \times & \times \end{pmatrix} \Rightarrow \begin{pmatrix} \times & \times & \times & \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & \times & \times & \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & \times & 0 & \times & \times & \times & \times & \times & \times \\ 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & \times & 0 & \times & \times & \times & \times & \times & \times \\ \times & \times & 0 & \times & \times & \times & \times & \times & \times \\ \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Observation row m is $\alpha \cdot e_m$!

Cascade

All columns with pivot in row m (after Π):

$$\begin{pmatrix} \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & \times & \times & \times & \times & \times & \times & \times & \times \end{pmatrix} \Rightarrow \begin{pmatrix} \times & \times & \times & \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & \times & \times & \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & \times & 0 & \times & \times & \times & \times & \times & \times \\ 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & \times & 0 & \times & \times & \times & \times & \times & \times \\ \times & \times & 0 & \times & \times & \times & \times & \times & \times \\ \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Observation row m is $\alpha \cdot e_m$!

Fix Remove fill by using row m as a pivot row.

Cascade

All columns with pivot in row m (after Π):

$$\begin{pmatrix} \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & \times & \times & \times & \times & \times & \times & \times & \times \end{pmatrix} \Rightarrow \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Observation row m is $\alpha \cdot e_m$!

Fix Remove fill by using row m as a pivot row.

theorem!

Theorem

In K is a complex with m simplices covered by U and K^U has size $m + n$ then the mayer vietoris algorithm uses $O(mn^2)$ time and $O(mn)$ space.

In practice

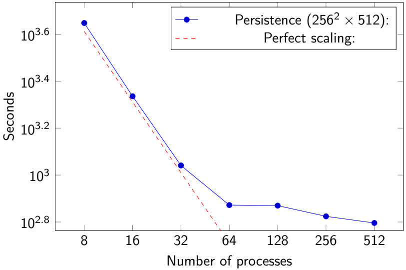


Figure: Times to compute persistence diagram for the $256^2 \times 512$ combustion data set. Credit: Dmitry Morozov.

In practice

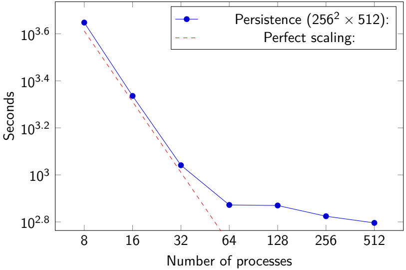


Figure: Times to compute persistence diagram for the $256^2 \times 512$ combustion data set. Credit: Dmitry Morozov.

- An input of size of 1.3×10^6 while quite large, is still considerably smaller than what can be computed today.
- Interesting: memory usage is not closely tracking our space bound.
- Slowdown as number of processes increase matches our intuition, total size of intersection is getting much larger.

Future Directions

1. There is still some room to improve the space complexity of this algorithm, for example, by reducing the factor n in $O(mn)$
2. Algorithm is top heavy, eventually a large matrix is on one machine.
3. We wanted to use M.V. to avoid this! Still some room for more cleverness here.

Mayer Vietoris Spectral Sequence

$$
\begin{tikzcd}
\vdots \arrow[d, "\partial_K"] & \vdots \arrow[d, "\partial_K"] & \vdots \arrow[d, "\partial_K"] & \\
E^0_{2,0} \arrow[d, "\partial_K"] & E^0_{2,1} \arrow[l, "\partial_N"'] \arrow[d, "\partial_K"] & E^0_{2,2} \arrow[l, "\partial_N"'] \arrow[d, "\partial_K"] & \cdots \arrow[l, "\partial_N"'] \\
E^0_{1,0} \arrow[d, "\partial_K"] & E^0_{1,1} \arrow[l, "\partial_N"'] \arrow[d, "\partial_K"] & E^0_{1,2} \arrow[l, "\partial_N"'] \arrow[d, "\partial_K"] & \cdots \arrow[l, "\partial_N"'] \\
E^0_{0,0} & E^0_{0,1} \arrow[l, "\partial_N"'] & E^0_{0,2} \arrow[l, "\partial_N"'] & \cdots \arrow[l, "\partial_N"']
\end{tikzcd}
$$

$$E^0_{p,q} = \langle p\text{-chains in a } q\text{-way intersection} \rangle$$

Mayer Vietoris Spectral Sequence

$$
\begin{tikzcd}
 & \vdots \arrow[d, "\partial_K"] & \vdots \arrow[d, "\partial_K"] & \vdots \arrow[d, "\partial_K"] & \\
C_2\left(K^U\right) & E^0_{2,0} \arrow[d, "\partial_K"] & E^0_{2,1} \arrow[l, "\partial_N"] \arrow[d, "\partial_K"] & E^0_{2,2} \arrow[l, "\partial_N"] \arrow[d, "\partial_K"] & \cdots \arrow[l, "\partial_N"] \\
C_1\left(K^U\right) & E^0_{1,0} \arrow[d, "\partial_K"] & E^0_{1,1} \arrow[l, "\partial_N"] \arrow[d, "\partial_K"] & E^0_{1,2} \arrow[l, "\partial_N"] \arrow[d, "\partial_K"] & \cdots \arrow[l, "\partial_N"] \\
C_0\left(K^U\right) & E^0_{0,0} & E^0_{0,1} \arrow[l, "\partial_N"] & E^0_{0,2} \arrow[l, "\partial_N"] & \cdots \arrow[l, "\partial_N"]
\end{tikzcd}
$$

$$E^0_{p,q} = \langle p\text{-chains in a } q\text{-way intersection} \rangle$$

The first two differentials:

$$d_0 = \partial_K \text{ and } d_1 = \partial_N$$

Mayer Vietoris Spectral Sequence

$$\begin{tikzcd}
\vdots \arrow[d, "\partial_K"] & \vdots \arrow[d, "\partial_K"] & \vdots \arrow[d, "\partial_K"] & \\
E^0_{2,0} \arrow[d, "\partial_K"] & E^0_{2,1} \arrow[l, "\partial_N"] \arrow[d, "\partial_K"] & E^0_{2,2} \arrow[l, "\partial_N"] \arrow[d, "\partial_K"] & \cdots \arrow[l, "\partial_N"] \\
E^0_{1,0} \arrow[d, "\partial_K"] & E^0_{1,1} \arrow[l, "\partial_N"] \arrow[d, "\partial_K"] & E^0_{1,2} \arrow[l, "\partial_N"] \arrow[d, "\partial_K"] & \cdots \arrow[l, "\partial_N"] \\
E^0_{0,0} & E^0_{0,1} \arrow[l, "\partial_N"] & E^0_{0,2} \arrow[l, "\partial_N"] & \cdots \arrow[l, "\partial_N"]
\end{tikzcd}$$

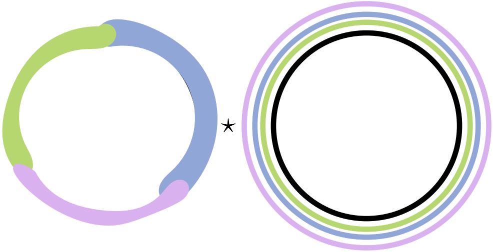
$$E^0_{p,q} = \langle p\text{-chains in a } q\text{-way intersection}\rangle$$

The first two differentials:

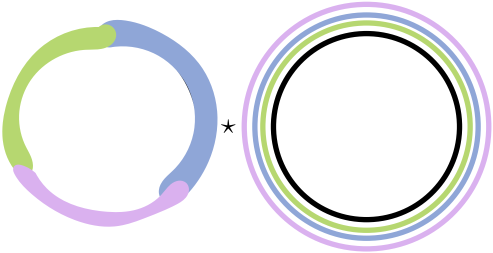
$$d_0 = \partial_K \text{ and } d_1 = \partial_N$$

We can construct the blowup *chain complex* where $C_d = \bigoplus_{p+q=d} E^0_{p,q}$ with $\partial = d_0 + (-1)^q d_1$ Let's try an example!

Example: $S^1 \star S^1$

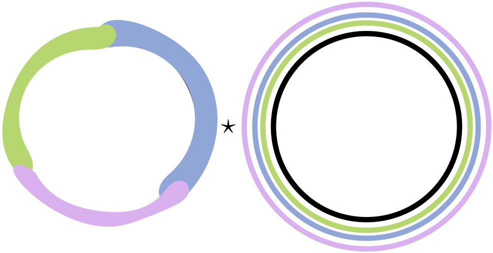


Example: $S^1 \star S^1$



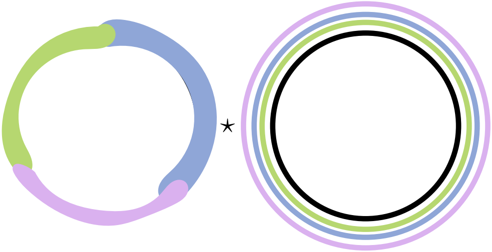
1. Each of the three sets are a copy of $I \star S^1$

Example: $S^1 \star S^1$



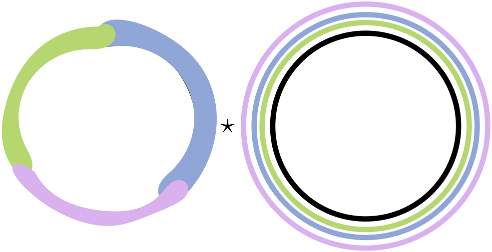
1. Each of the three sets are a copy of $I \star S^1$
2. Each of the three pairwise intersection is $\{\mathrm{pt}\} \star S^1$

Example: $S^1 \star S^1$



1. Each of the three sets are a copy of $I \star S^1$
2. Each of the three pairwise intersection is $\{\mathrm{pt}\} \star S^1$
3. Single triple intersection is a copy of S^1.

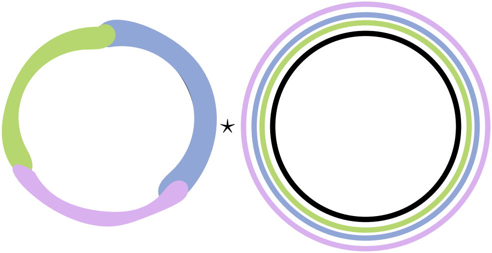
Example: $S^1 \star S^1$



The E_1 page has terms with the following data:

$$\begin{array}{ccccc} 0 & & 0 & & 1 \\ \\ 3 & \xleftarrow[\begin{pmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \end{pmatrix}]{} & 3 & \xleftarrow[\begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix}]{} & 1 \end{array}$$

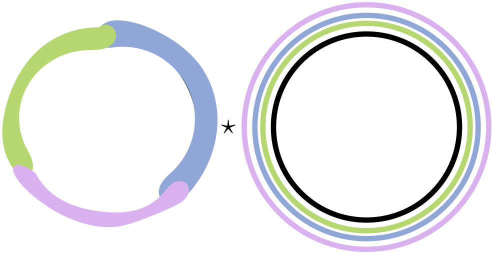
Example: $S^1 \star S^1$



The E_2 page has terms of the following data:

$$\begin{tikzcd}
0 & 0 & 1 \\
1 & 0 & 0 \arrow[ull, "0"]
\end{tikzcd}$$

Example: $S^1 \star S^1$



The E_2 page has terms of the following data:

$$\begin{tikzcd}
0 & 0 & 1 \\
1 & 0 & 0 \arrow[ull, "0"']
\end{tikzcd}$$

$$H_d(K^U) = \bigoplus_{p+q=d} E^{\infty}_{p,q}$$

$$H_0(S^1 \star S^1) \cong H_2(S^1 \star S^1) = 1$$