Time-optimal Persistent Homology Representatives

for Data Growing Over Time

António Leitão

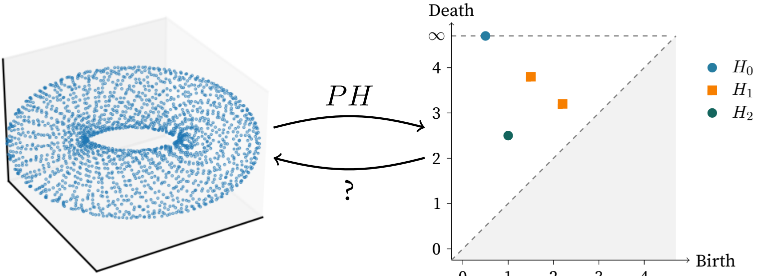
Joint work with Nina Otter

Scuola Normale Superiore, Pisa
Datashape, INRIA, Saclay

13 October 2025

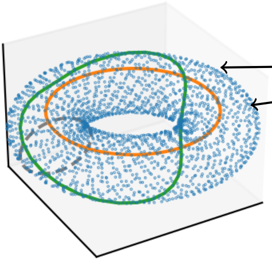
ComPer 2025: Workshop on Computational Topology

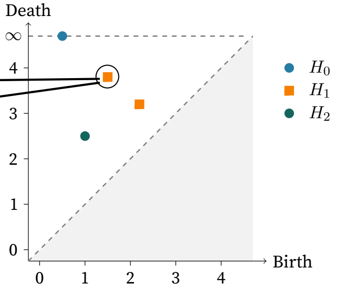
Motivation



How can we better interpret
persistent homology information?

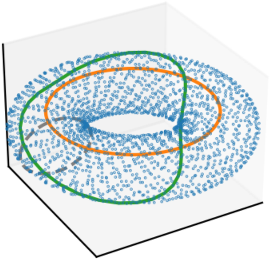
Motivation

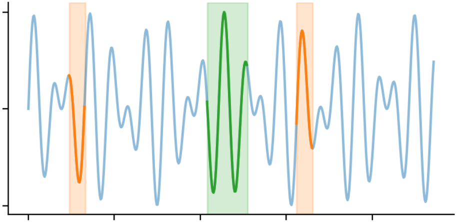




Each homology class has
multiple representatives

Motivation





With time-indexed data we would prefer representatives that are *"close"* in time

Previous Work

Optimal Cycles

1. Localizing homology classes (Chen & Freedman 2008)
2. ℓ_0 minimization is NP-hard (Chen & Freedman 2010)
3. Linear Programming ℓ_1 norm (Dey et al. 2011)
4. Volume-optimal cycles (Obayashi 2018)

Previous Work

Optimal Cycles

1. Localizing homology classes (Chen & Freedman 2008)
2. ℓ_0 minimization is NP-hard (Chen & Freedman 2010)
3. Linear Programming ℓ_1 norm (Dey et al. 2011)
4. Volume-optimal cycles (Obayashi 2018)

Filtrations

1. Optimal filtered cycle bases (Escolar & Hiraoka 2016)
2. Filtered 1-cycles is NP-hard (Dey, Hou & Mandal 2019)
3. LP solutions almost always yield $\{-1, 0, 1\}$ coefficients (Li et al. 2021)

Previous Work

Optimal Cycles

1. Localizing homology classes (Chen & Freedman 2008)
2. ℓ_0 minimization is NP-hard (Chen & Freedman 2010)
3. Linear Programming ℓ_1 norm (Dey et al. 2011)
4. Volume-optimal cycles (Obayashi 2018)

Filtrations

1. Optimal filtered cycle bases (Escolar & Hiraoka 2016)
2. Filtered 1-cycles is NP-hard (Dey, Hou & Mandal 2019)
3. LP solutions almost always yield $\{-1, 0, 1\}$ coefficients (Li et al. 2021)

Others

1. Harmonic (Basu & Cox 2021) Stable volumes (Obayashi 2021) Zigzag (Dey, Hou & Morozov 2024)

Outline

Representative Cycles

K: Simplicial Complex

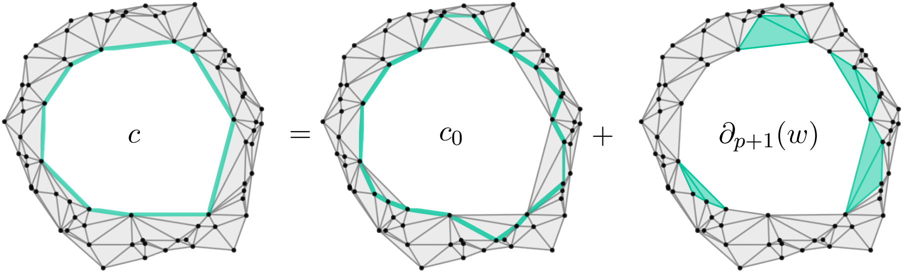
$S_p(K)$ its set of p-simplices

$C_p(K)$ vector space over $\mathbb{F}$

$\partial_p : C_p(K) \longrightarrow C_{p-1}(K)$

p-cycles: elements of kernel of ∂_p

Representative Cycles



Representative Cycles

$$\begin{aligned} \text{min} \quad & \ell(c) \\ \text{subject to} \quad & c = c_0 + \partial_{p+1}(w) \\ & w \in C_{p+1}(K) \end{aligned}$$

Search for homologous p-cycles
by adding boundaries of $p+1$-simplicies.

Linear Programming

$$
\begin{aligned}
\min \quad & \|W\mathbf{c}\|_1 = \sum_i \sum_j w_{ij}(c_j^+ + c_j^-) \\
\text{subject to} \quad & (\mathbf{c}^+ - \mathbf{c}^-) = \mathbf{c_0} + \partial_{p+1}(\mathbf{w}) \\
& \mathbf{c}^+, \mathbf{c}^- \geq 0
\end{aligned}
$$

Dey, Hirani, and Krishnamoorthy 2010
Li et al. (2021), *"Minimal Cycle Representatives in Persistent Homology Using Linear Programming: An Empirical Study With User's Guide"*

Linear Programming

$$\begin{bmatrix} w_{1,1} & 0 & 0 & \cdots & 0 \\ 0 & w_{2,2} & 0 & \cdots & 0 \\ 0 & 0 & w_{3,3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & w_{n,n} \end{bmatrix} \left(\begin{bmatrix} c_0^+ \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ c_k^- \\ 0 \end{bmatrix} \right)$$

Diagonal Matrix

$w_{ii} =$ cost of having σ_i in the chain (e.g area/length)

$w_{ii} = 1 \approx$ minimizing the count

Linear Programming

$$\begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} & \cdots & w_{1,n} \\ w_{2,1} & w_{2,2} & w_{2,3} & \cdots & w_{2,n} \\ w_{3,1} & w_{3,2} & w_{3,3} & \cdots & w_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{n,1} & w_{n,2} & w_{n,3} & \cdots & w_{n,n} \end{bmatrix} \left(\begin{bmatrix} c_0^+ \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ c_k^- \\ 0 \end{bmatrix} \right)$$

Full weight matrix

$w_{ij} =$ cost of having both σ_i and σ_j together in the chain.

Filtered Cycle Representative

What if we have a filtered simplicial complex?

Filtered Cycle Representative

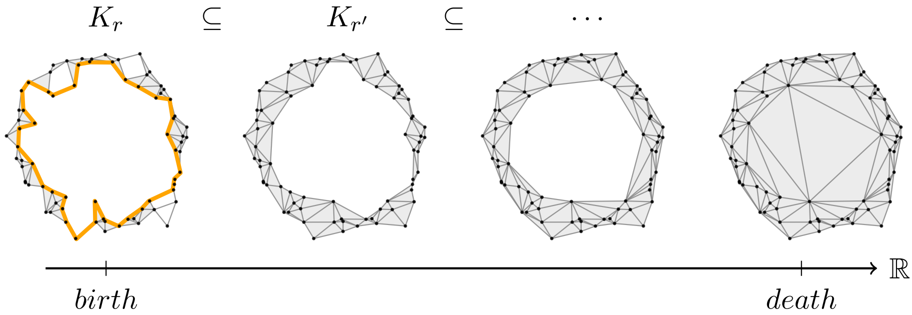
$$\begin{aligned} \min \quad & \ell(c) \\ \text{subject to} \quad & c = c_0 + \partial_{p+1}(w) \\ & w \in C_{p+1}(K) \end{aligned}$$

Li et al. (2021), *"Minimal Cycle Representatives in Persistent Homology Using Linear Programming: An Empirical Study With User's Guide"*
Chen and Freedman (2008), *Quantifying Homology Classes*

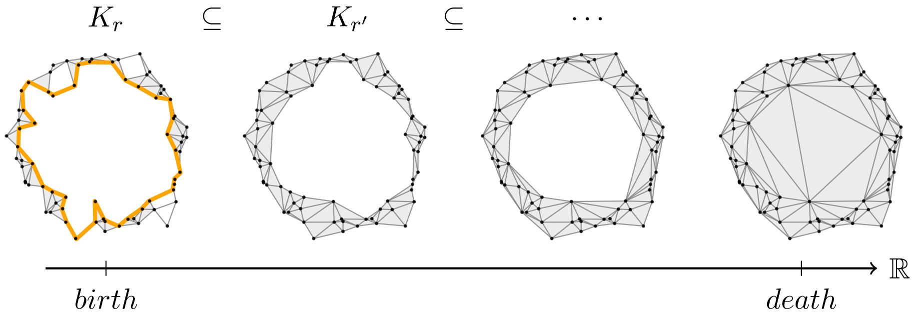
Filtered Cycle Representative

$$\begin{aligned} \min \quad & \ell(c) \\ \text{subject to} \quad & c = c_0 + \partial_{p+1}(w) \\ & w \in C_{p+1}(K) \\ & \mathrm{Birth}(c) = a \\ & \mathrm{Death}(c) = b \end{aligned}$$

Li et al. (2021), *"Minimal Cycle Representatives in Persistent Homology Using Linear Programming: An Empirical Study With User's Guide"*
Chen and Freedman (2008), *Quantifying Homology Classes*



We do not consider simplicies
born *after* the birth of the class



$$P = \{\sigma \in S_p(K_b) \mid \text{birth}(\sigma) \leq b\}$$

$$Q = \{\sigma \in S_{p+1}(K_b) \mid \text{birth}(\sigma) \leq b\}$$

Filtered Cycle Representative

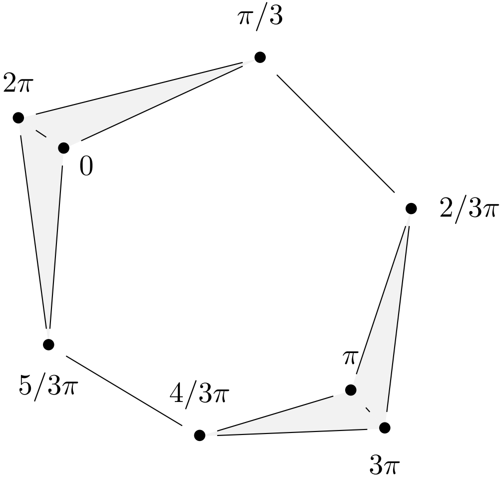
$$
\begin{aligned}
\text{min} \quad & \|W\mathbf{c}\|_1 = \sum_i \sum_j w_{ij}(c_j^+ + c_j^-) \\
\text{subject to} \quad & (\mathbf{c}^+ - \mathbf{c}^-) = \mathbf{c_0} + \partial_{p+1}[P,Q](\mathbf{w}) \\
& \mathbf{w} \in \mathbb{R}^{|Q|} \\
& \mathbf{c} \in \mathbb{R}^{|P|} \\
& \mathbf{c}^+, \mathbf{c}^- \geq 0
\end{aligned}
$$

Dey, Hirani, and Krishnamoorthy (2010), *"Optimal homologous cycles, total unimodularity, and linear programming"*
Li et al. (2021), *"Minimal Cycle Representatives in Persistent Homology Using Linear Programming: An Empirical Study With User's Guide"*

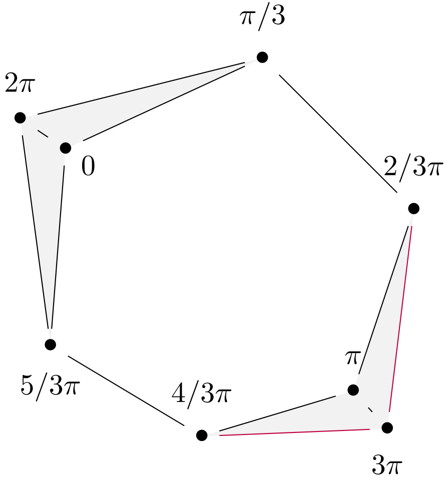
Time Optimality

How to measure how spread out representatives are in time

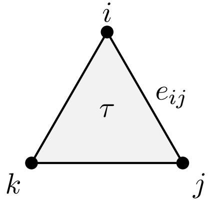
What if we have time data



What if we have time data



Time at Vertex-Level

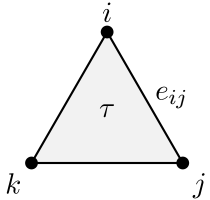


Cost of selecting e_{ij}

$$w_{e_{ij}} = |T_i - T_j|$$

We find chains whose vertices have
time labels that are close to each other

Time at Vertex-Level

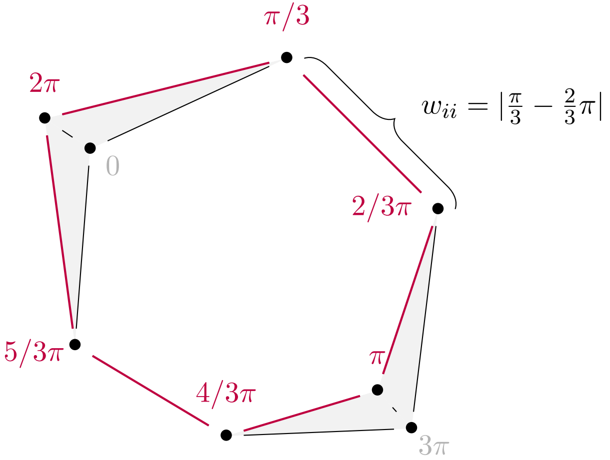


Cost of selecting τ

$$w_\tau = \max_{r\in\{i,j,k\}} T_r - \min_{s\in\{i,j,k\}} T_s$$

We find chains whose vertices have
time labels that are close to each other

Time at Vertex-Level

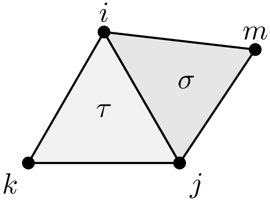


Time at Vertex-Level

$$W_{\text{vertex}}$$

$$\begin{bmatrix} w_{1,1} & 0 & 0 & \cdots & 0 \\ 0 & w_{2,2} & 0 & \cdots & 0 \\ 0 & 0 & w_{3,3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & w_{n,n} \end{bmatrix}$$

Time at Simplex-Level

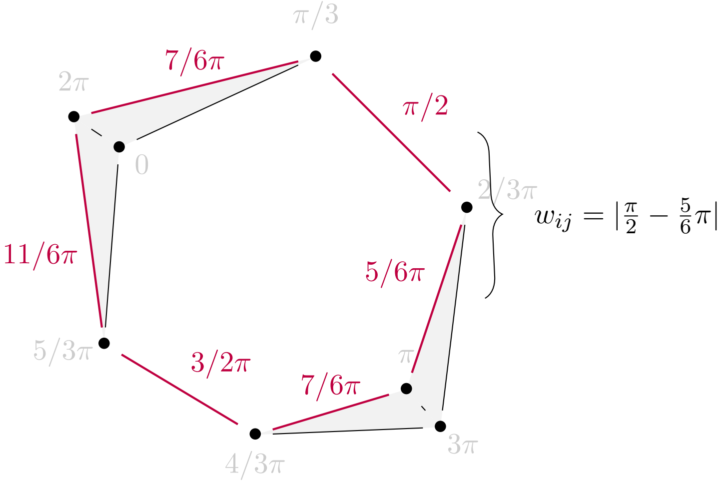


Cost of selecting both σ and τ

$$w_{\sigma,\tau} = |T_\sigma - T_\tau|$$

We first situate each simplex in time
and find a chain of closely situated simplices

Time at Simplex Level



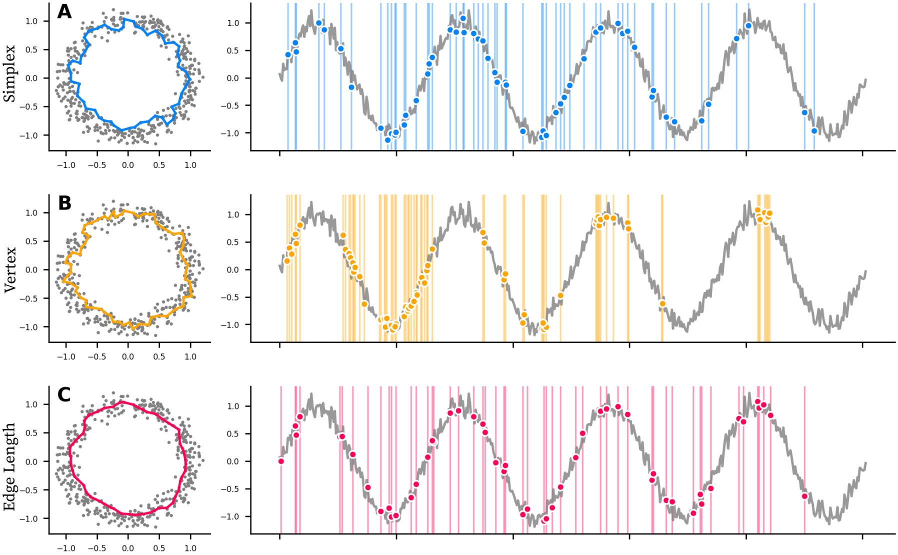
Time at Simplex Level

$$W_{\text{simplex}}$$

$$\begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} & \cdots & w_{1,n} \\ w_{2,1} & w_{2,2} & w_{2,3} & \cdots & w_{2,n} \\ w_{3,1} & w_{3,2} & w_{3,3} & \cdots & w_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{n,1} & w_{n,2} & w_{n,3} & \cdots & w_{n,n} \end{bmatrix}$$

Example

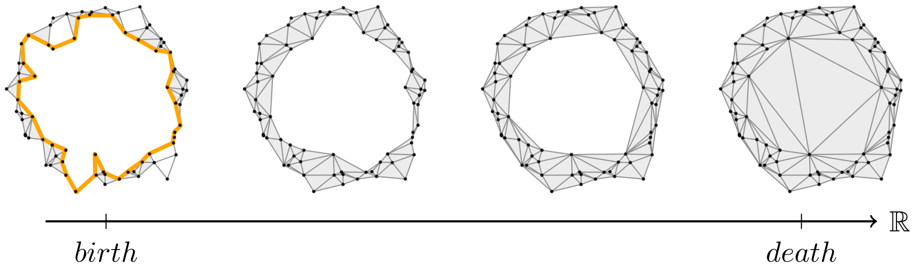
Noisy sine with 4 revolutions



A
B
C
Simplex
Vertex
Edge Length
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0

So what happened?

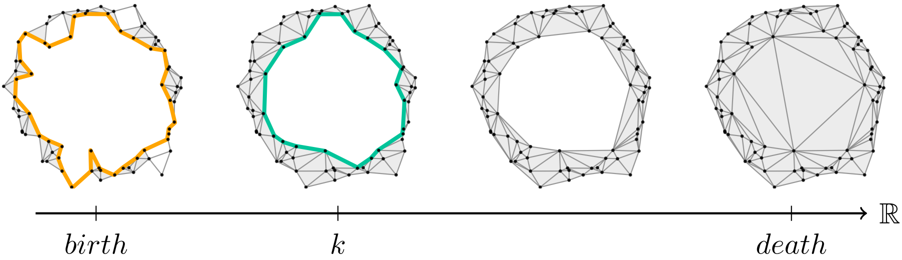
We have to increase our search space



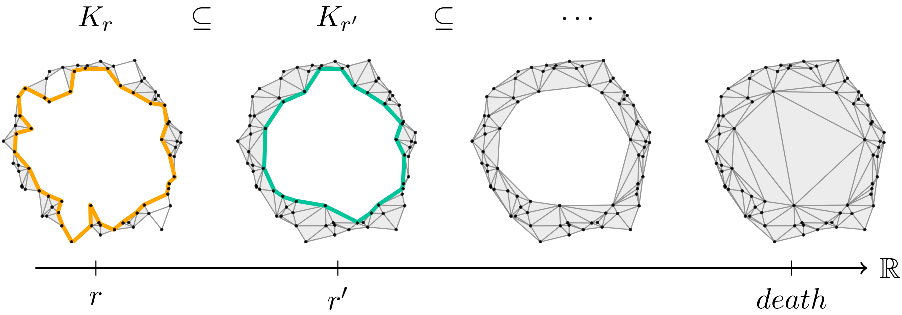
Relaxing Persistence Constraint

Increases search space

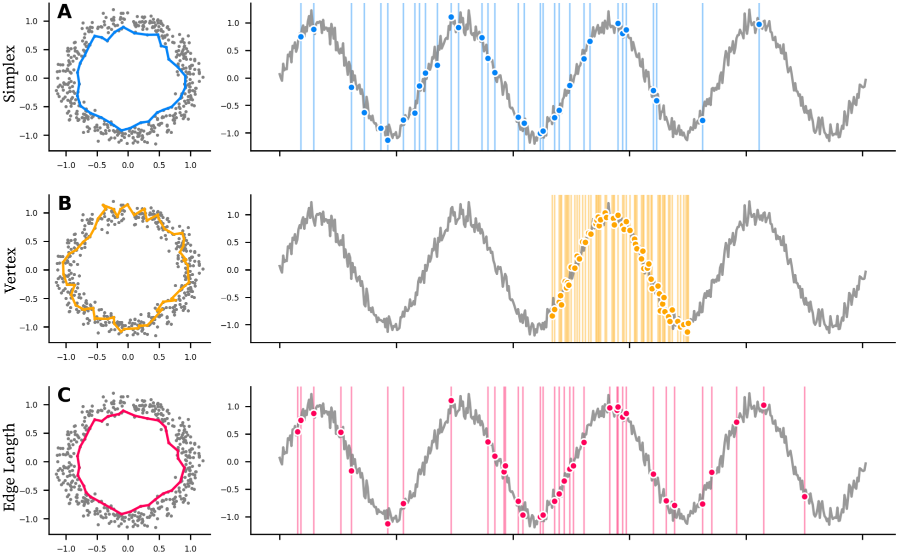
Makes sense when optimization is not area



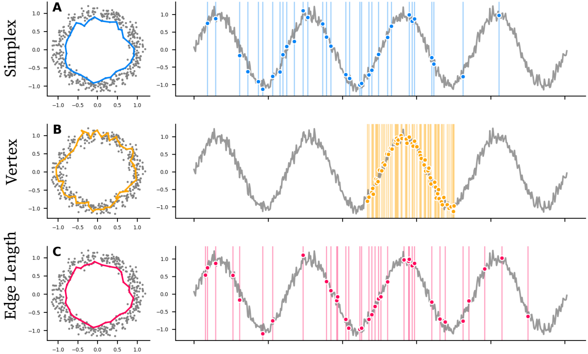
Resulting persistence is $death - k$



$$r \leq r' \implies \min\{f(c) \mid c \in C_p(K_r)\} \geq \min\{f(c) \mid c \in C_p(K_{r'})\}$$



Representative has 95% of total persistence

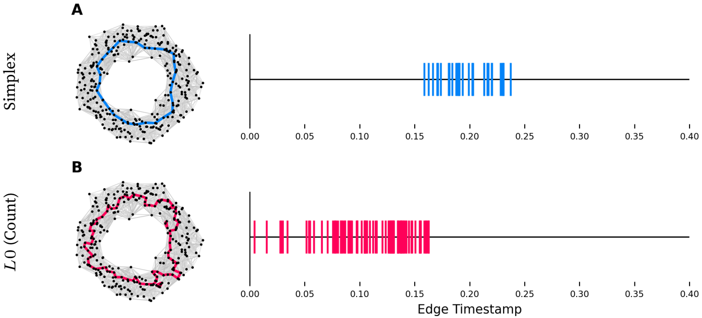


Method	L_1	Time Dispersion
Initial Representative	9.03	370
Time (Simplex)	5.51	112
Time (Vertex)	7.78	**81**
L_1 Optimal	**4.93**	178

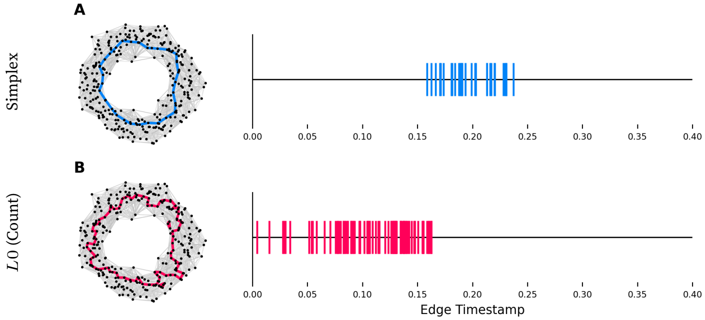
Example

Network with time weighted edges

Example (Time Network)



Example (Time Network)



Method	Cycle Length	Time Dispersion
Initial Representative	75	255.69
Time-Optimal	26	**12.24**
L_0 Minimisation	**24**	20.15

Example

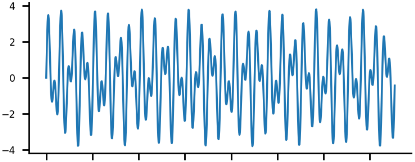
Quasi periodic signal

$$f(t) = 2\sin t + 1.8\sin\sqrt{3}t \quad 0 \le t \le 60\pi$$

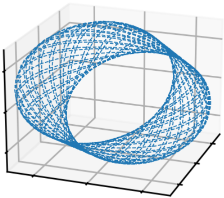
Gakhar and Perea (2023), *"Sliding Window Persistence of Quasiperiodic Functions"*

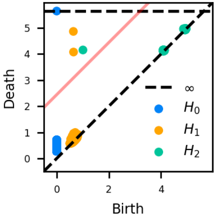
Example (Quasi-Periodic Signal)

$$f(t) = 2\sin t + 1.8\sin\sqrt{3}t \quad 0 \leq t \leq 60\pi$$



Time-series

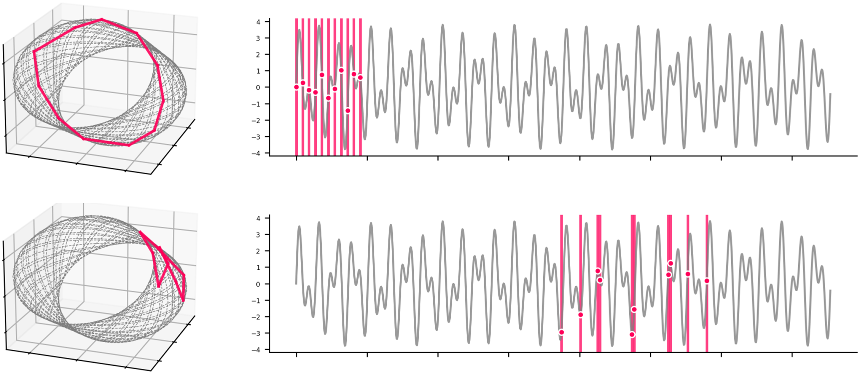




Sliding window Embedding

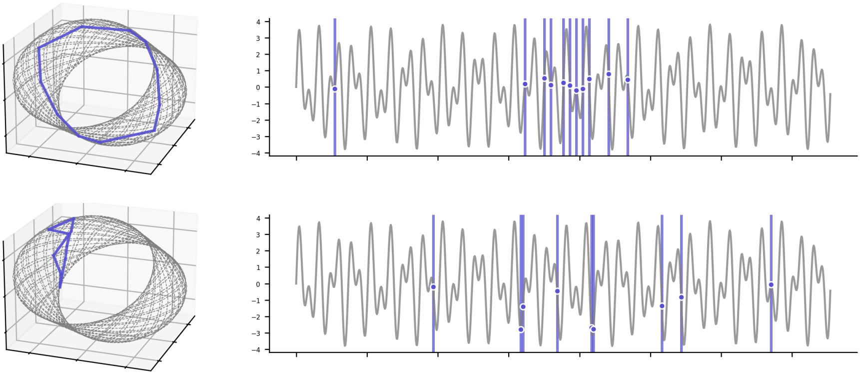
Gakhar and Perea (2023), *"Sliding Window Persistence of Quasiperiodic Functions"*

Example (Quasi-Periodic Signal)



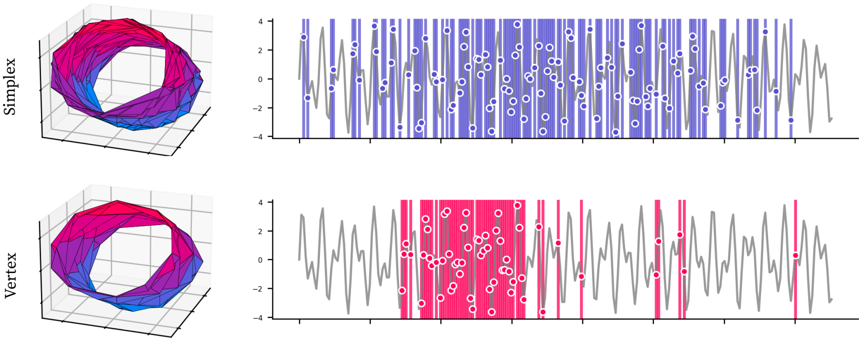
Vertex Based

Example (Quasi-Periodic Signal)



Simplex Based

Example (Quasi-Periodic Signal)

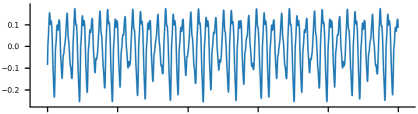


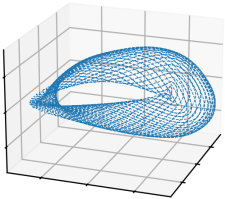
Application

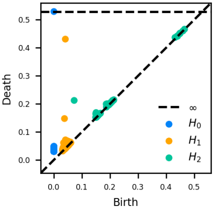
El Niño Southern Oscillation
climate model (ENSO)

Christensen et al. (2017), *"Stochastic Parameterization and El Niño–Southern Oscillation"*

Application (El Niño)



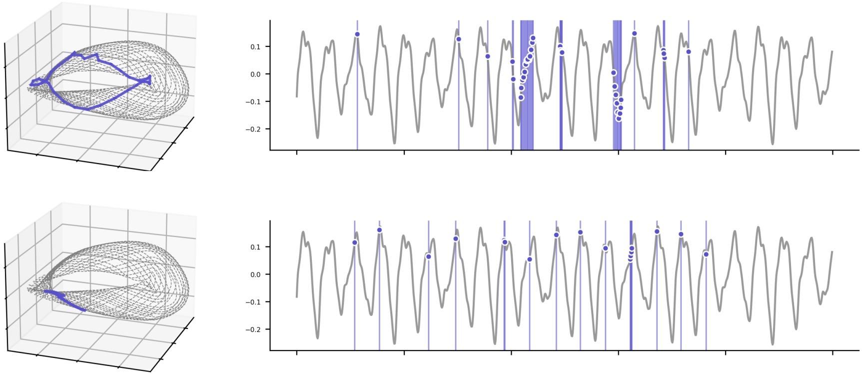




Time-series

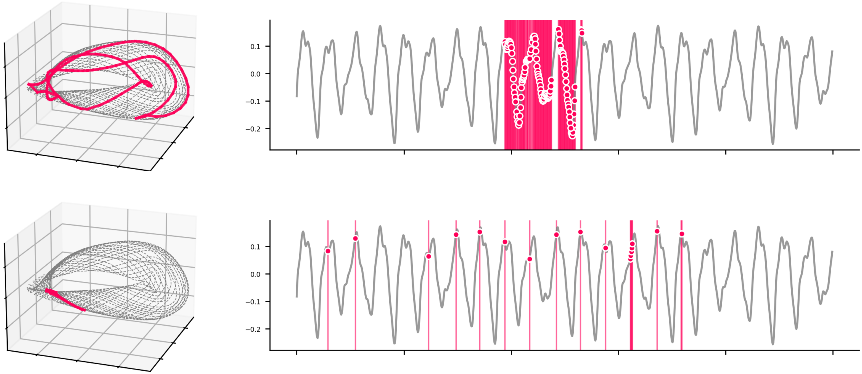
Sliding window Embedding

Application (El Niño)



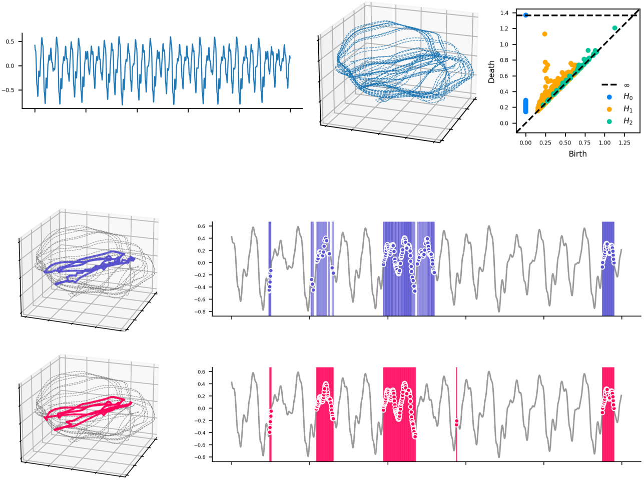
Simplex Based

Application (El Niño)



Vertex Based

Application (El Niño)

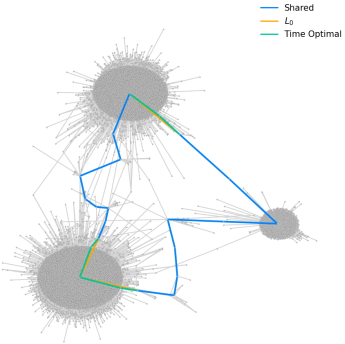


Application

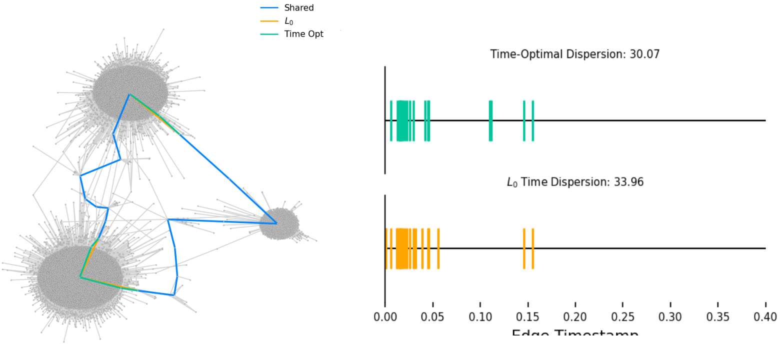
Bank transaction networks
for credit fraud detection

Altman et al. (2024), *Realistic Synthetic Financial Transactions for Anti-Money Laundering Models*

Example (Time Network)



Example (Time Network)



Future Directions

Thank You

In collaboration with Nina Otter
Special Thanks to Hannah Christenssen