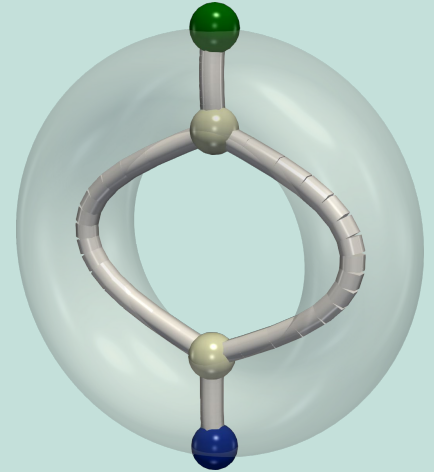


# Topological Data Analysis with the Topology ToolKit



Julien Tierny

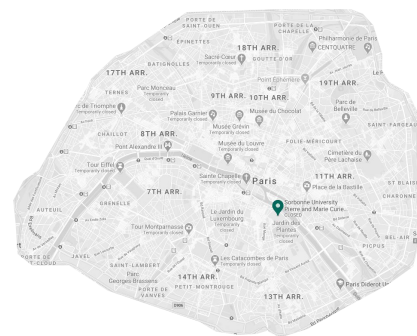
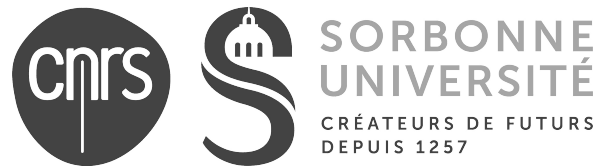


SORBONNE  
UNIVERSITÉ  
CRÉATEURS DE FUTURS  
DEPUIS 1257



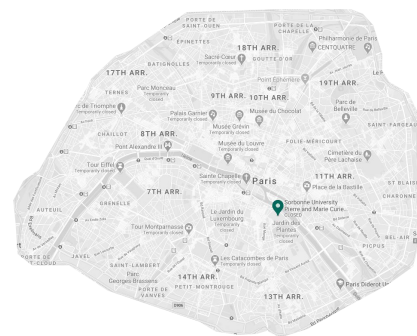
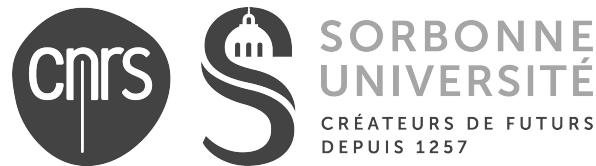
# About our research

- **TDA @ Sorbonne**
  - Topological data analysis and visualization
    - Interactive data analysis
    - Data from engineering & science
  - Computational aspects
    - Practical algorithms
    - Emerging data types



# About our research

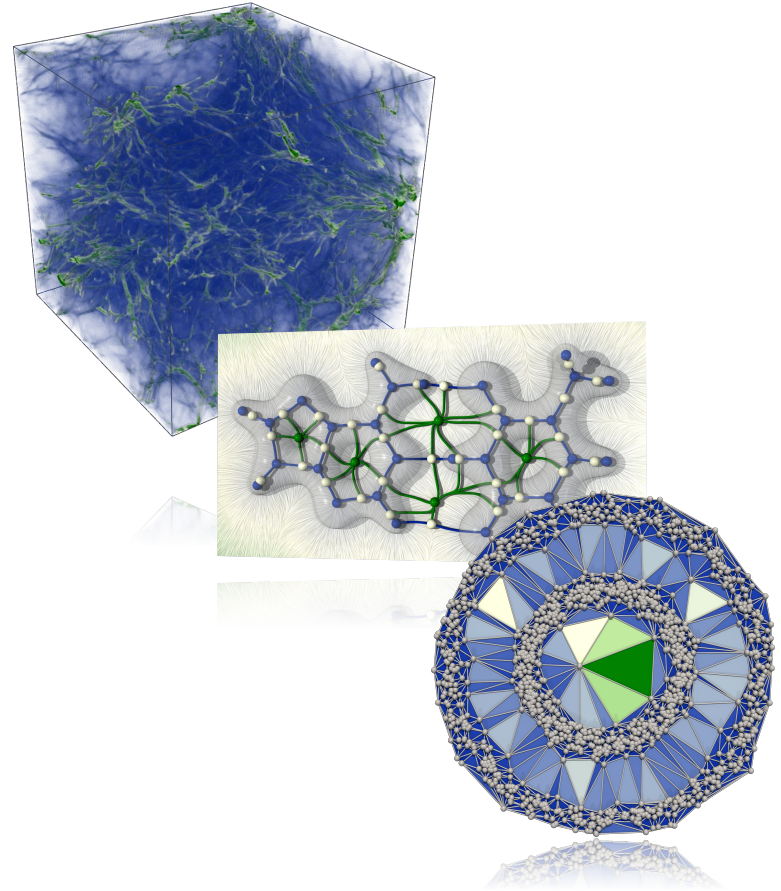
- **TDA @ Sorbonne**
  - Topological data analysis and visualization
    - Interactive data analysis
    - Data from engineering & science
  - Computational aspects
    - Practical algorithms
    - Emerging data types
  - Main venues
    - IEEE VIS, EuroVis, IEEE LDAV, TopoInVis



# What is Topological Data Analysis?

- **Context**

- Data
- On “meshes”, or “meshable” things



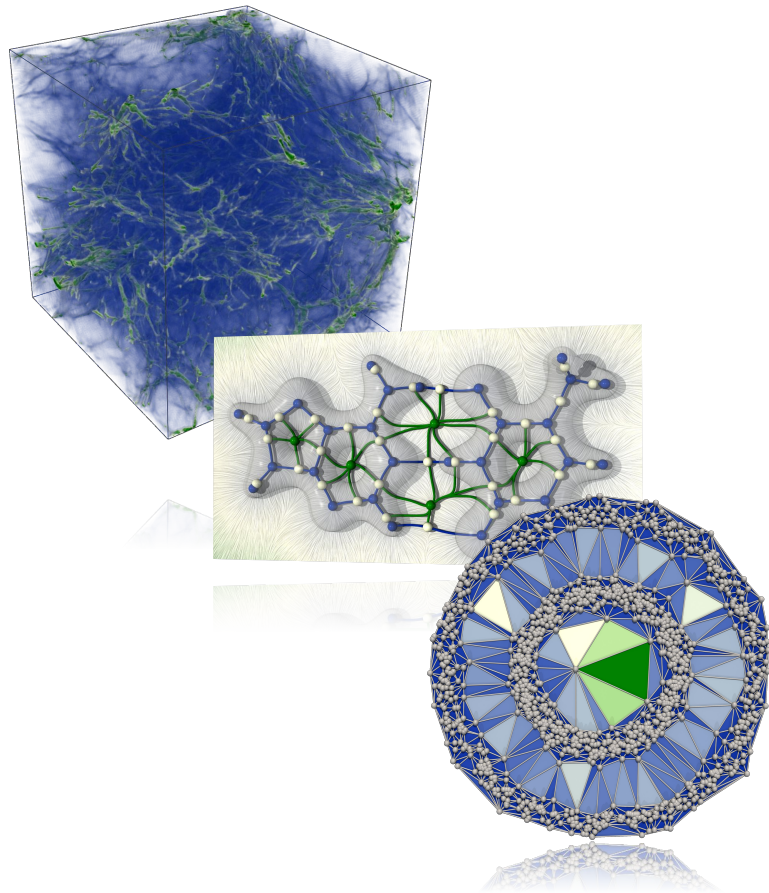
# What is Topological Data Analysis?

- **Context**

- Data
- On “meshes”, or “meshable” things

- **Swiss-army knife for feature extraction**

- Points, curves, surfaces, volumes, ...
- Robustness
- Multi-scale nature
- From raw data to features of interest



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

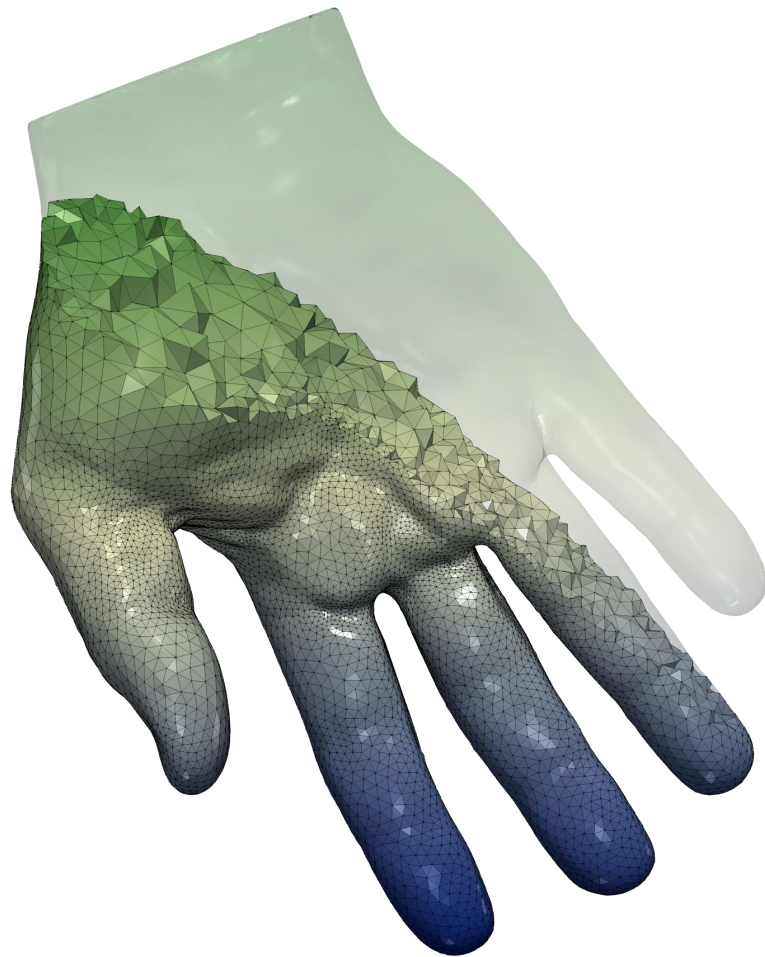


# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- $\mathcal{M}$ : simplicial complex



# Piecewise linear setting

- **Input PL scalar data**

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- $\mathcal{M}$ : simplicial complex

- Triangulated surface

- Tetrahedral volume

- 2D pixel image

- 3D voxel image

- Point cloud data





# Piecewise linear setting

- **Input PL scalar data**

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- $\mathcal{M}$ : simplicial complex

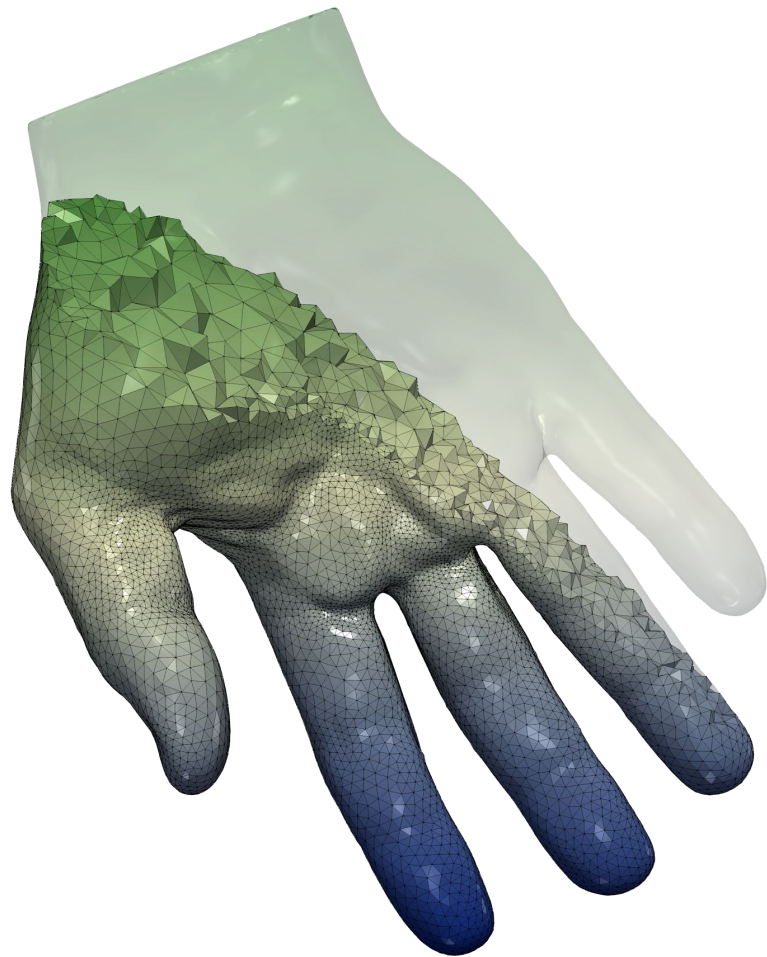
- Triangulated surface

- Tetrahedral volume

- 2D pixel image

- 3D voxel image

- Point cloud data



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

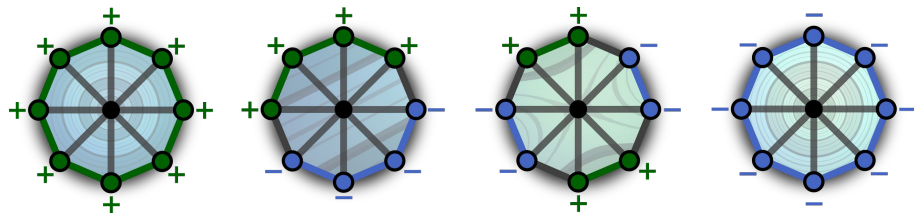
- Critical points

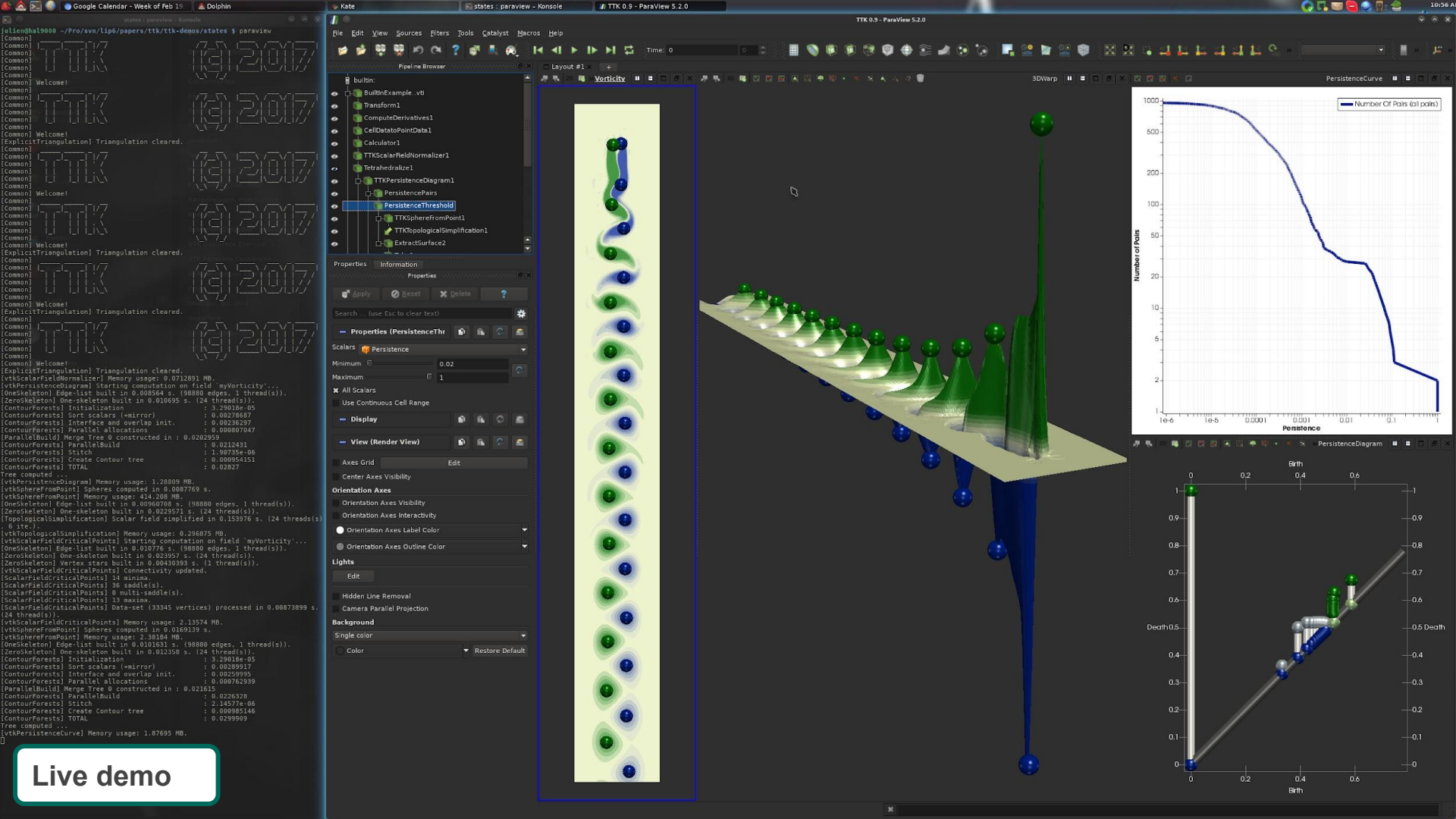


# Critical point extraction

- **Local link inspection**

- Banchoff 1970





Live demo



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

- Critical points



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

- Critical points



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

- Critical points



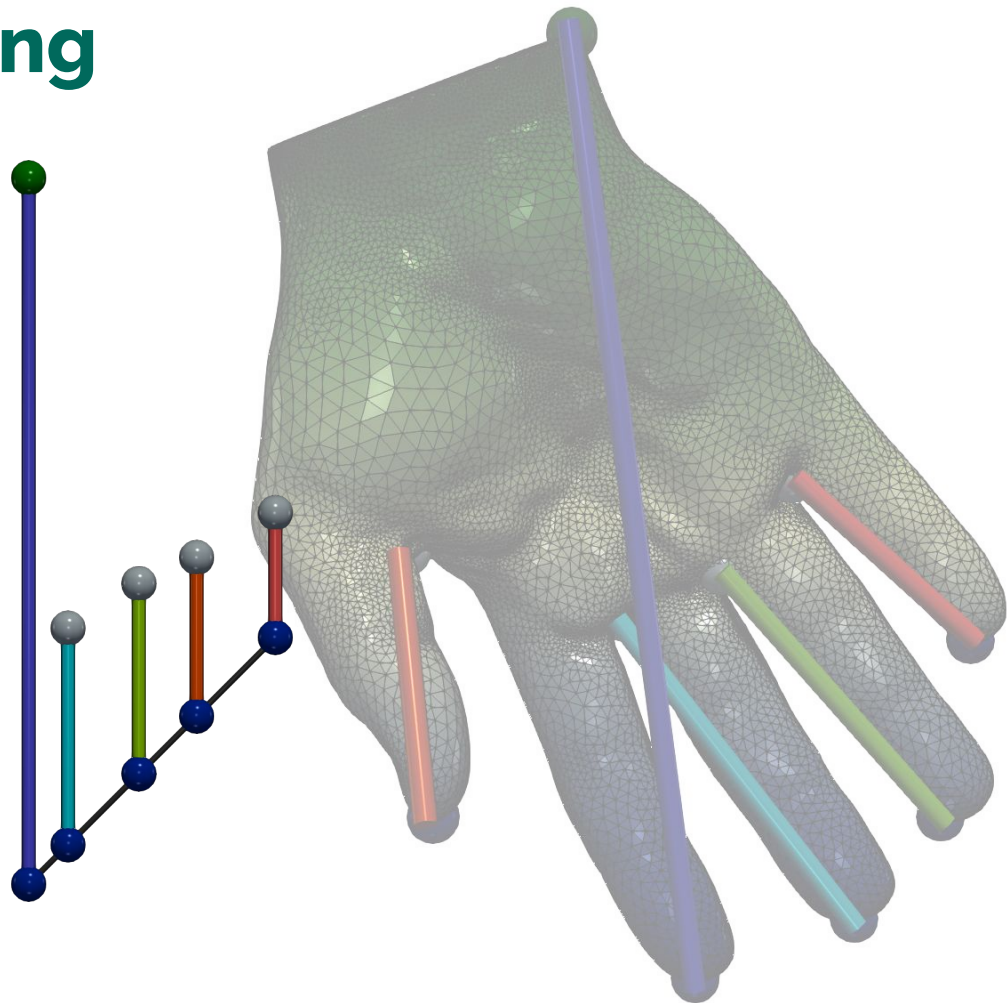
# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

- Critical points
- Persistence diagrams



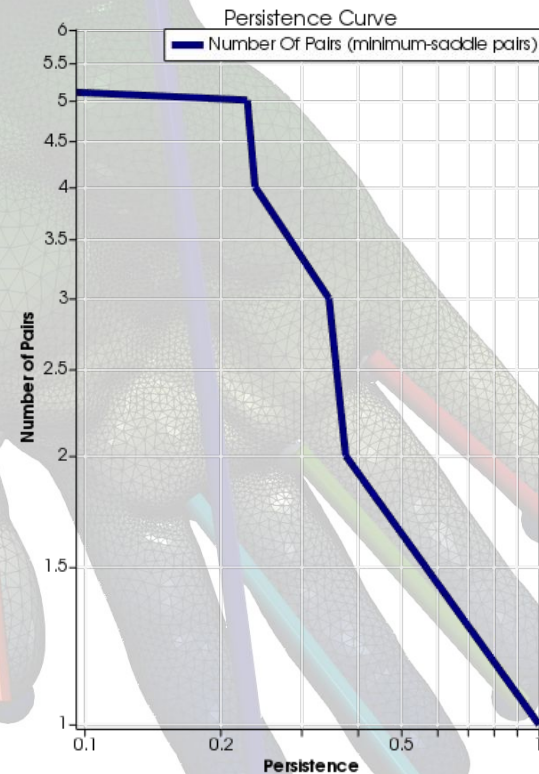
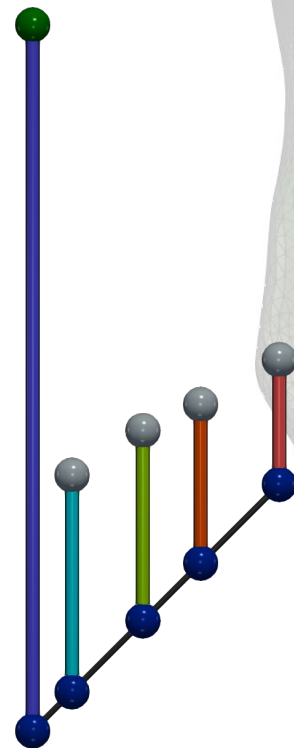
# Piecewise linear setting

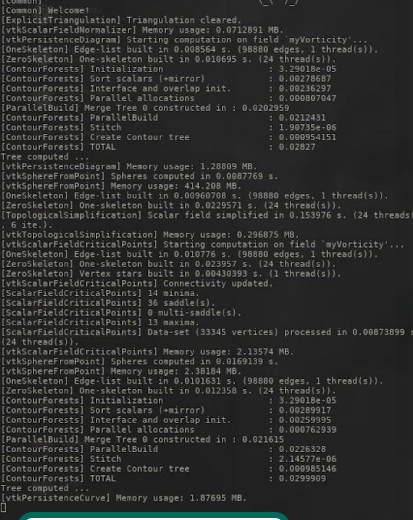
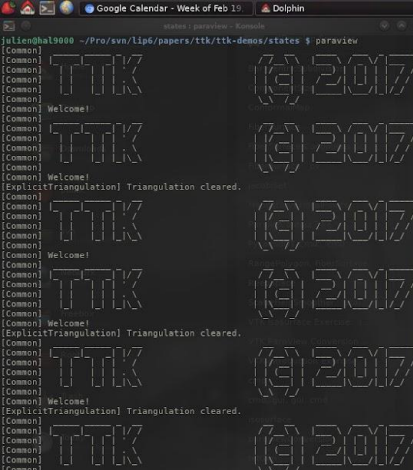
- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

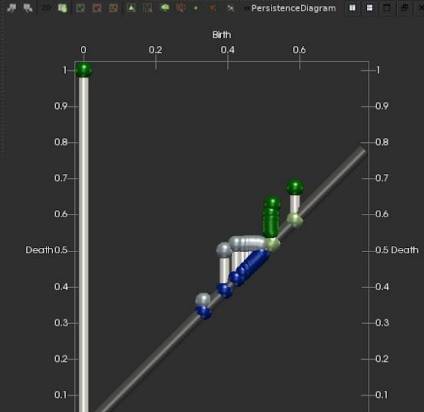
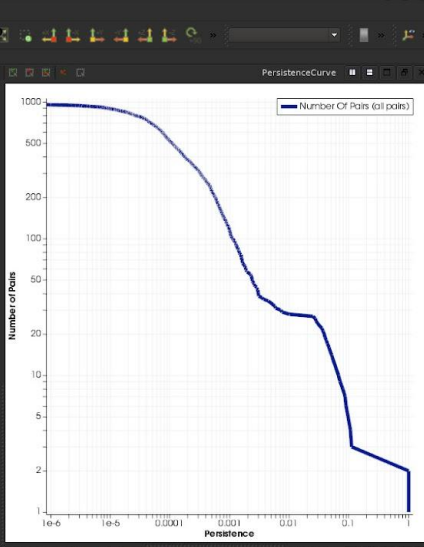
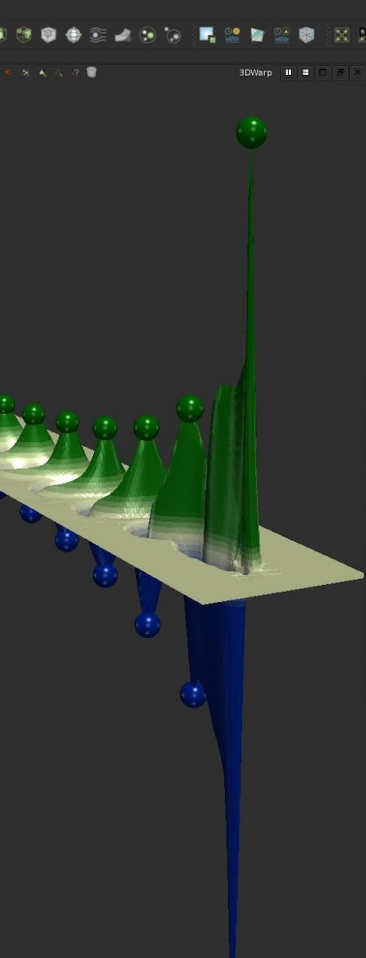
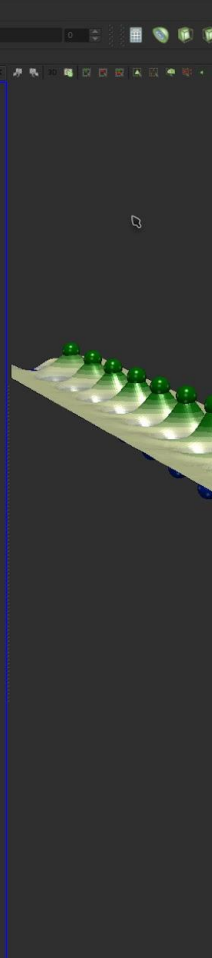
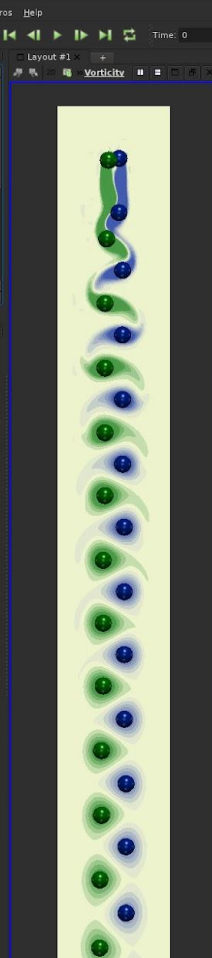
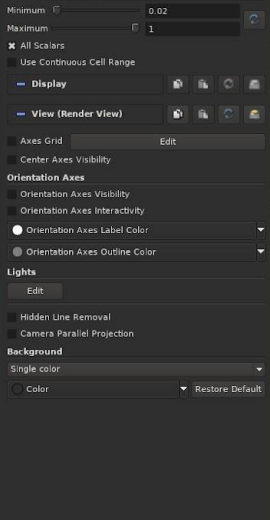
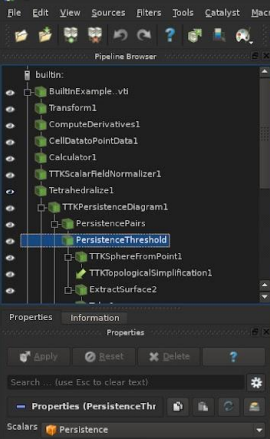
- Topological abstractions

- Critical points
- Persistence diagrams
- Persistence curves





Live demo



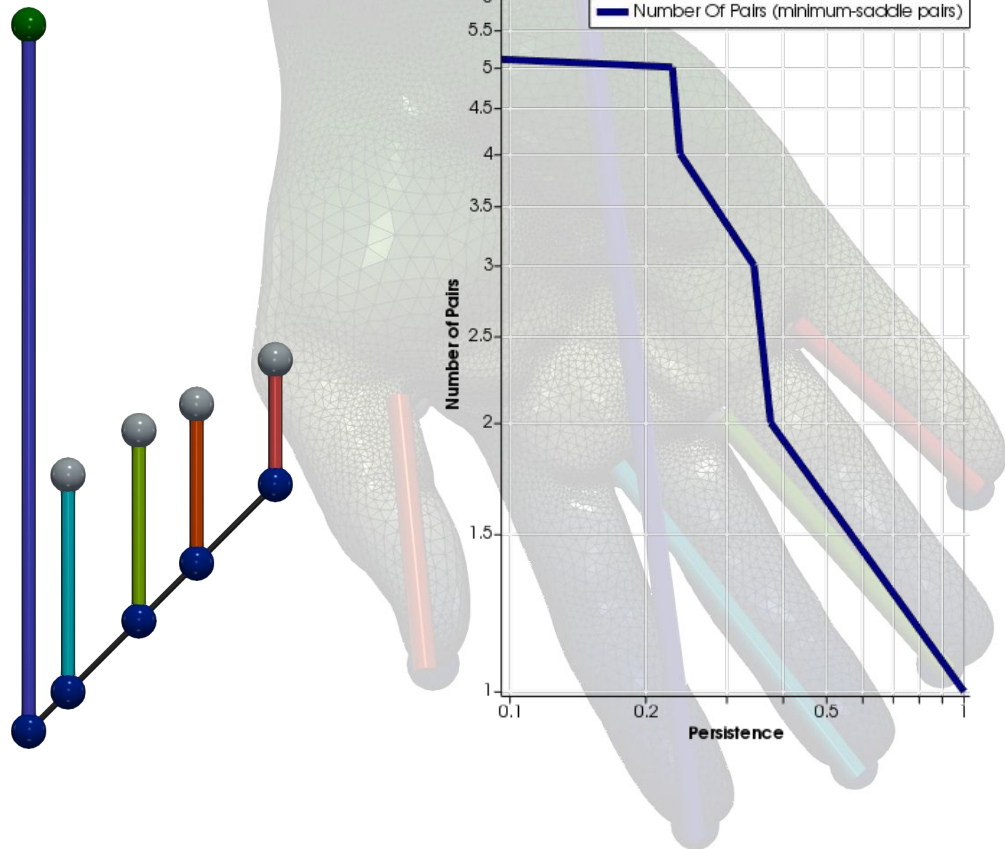
# Persistence diagrams/curves

- **Arbitrary dimension**

- Boundary matrix reduction
- Edelsbrunner et al. 2002

- **Low-dimensions**

- Union-Find data structures
  - Min-saddle pairs
  - Saddle-max pairs
  - *Gueunet et al. 2017*
- Saddle connectors
  - Saddle-saddle pairs



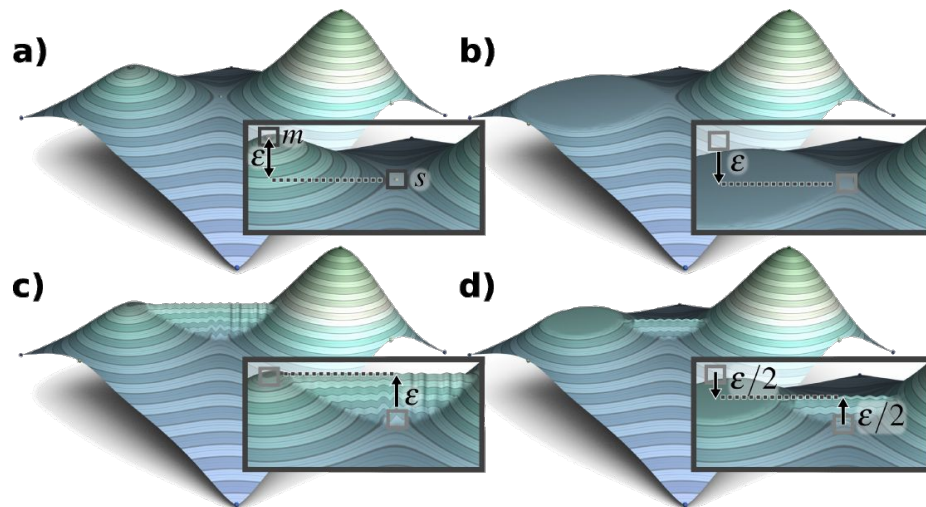
# Persistence simplification

- **Simplify the data**

- Retain only persistent features

- **Algorithms**

- Edelsbrunner et al. 2006, Attali et al. 2009, *Tierny and Pascucci 2012*, Bauer et al. 2012, *Lukasczyk et al. 2020*







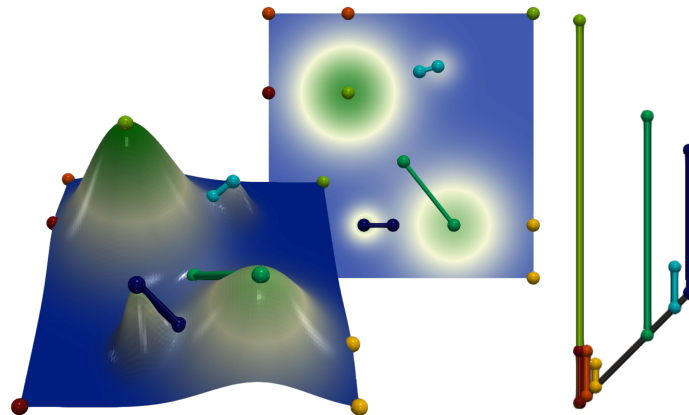
# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

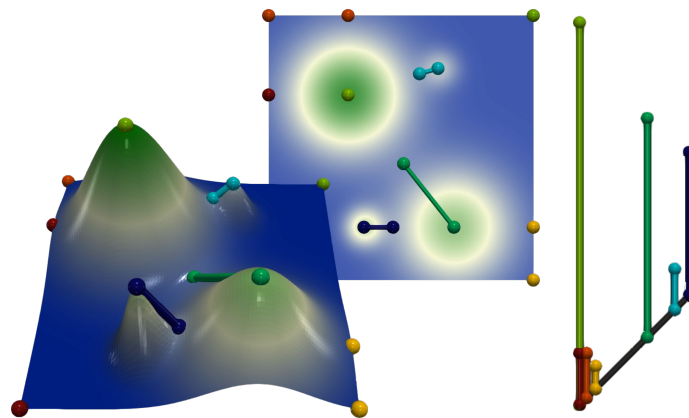
- Critical points
- Persistence diagrams
- Persistence curves



# Piecewise linear setting

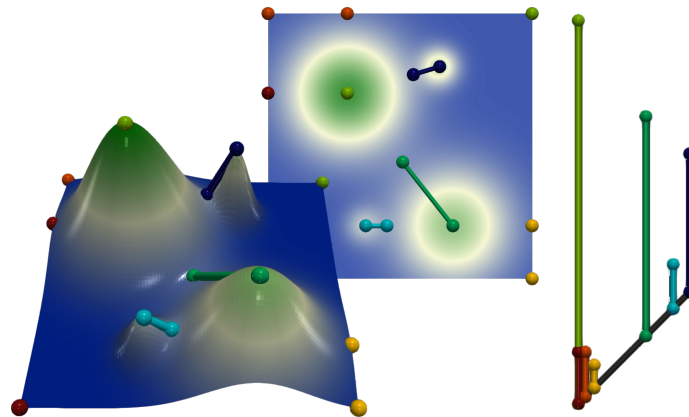
- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$



- Topological abstractions

- Critical points
- Persistence diagrams
- Persistence curves



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

- Critical points
- Persistence diagrams
- Persistence curves
- Reeb graphs



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

- Critical points
- Persistence diagrams
- Persistence curves
- Reeb graphs



# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

- Critical points
- Persistence diagrams
- Persistence curves
- Reeb graphs



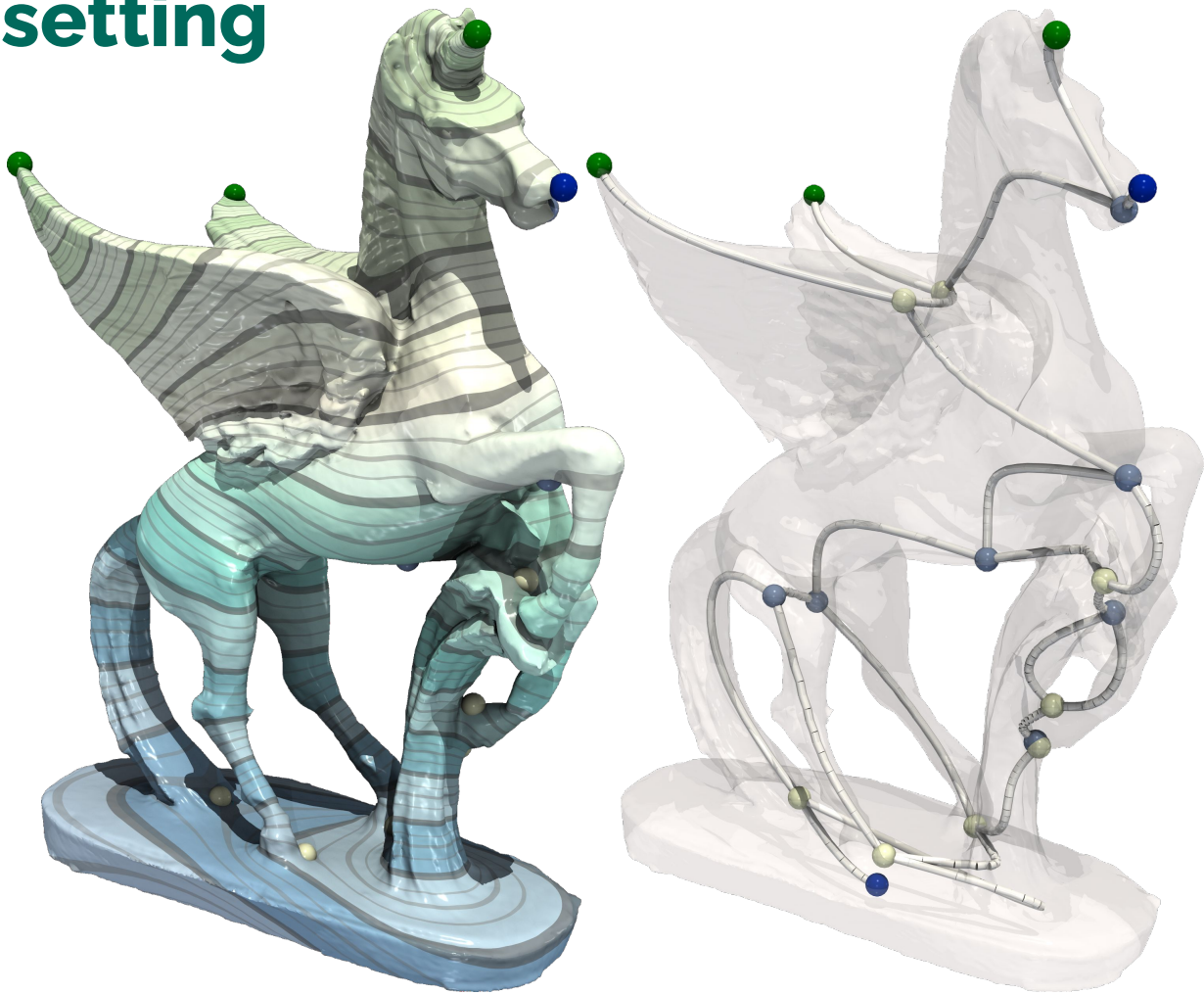
# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

- Critical points
- Persistence diagrams
- Persistence curves
- Reeb graphs



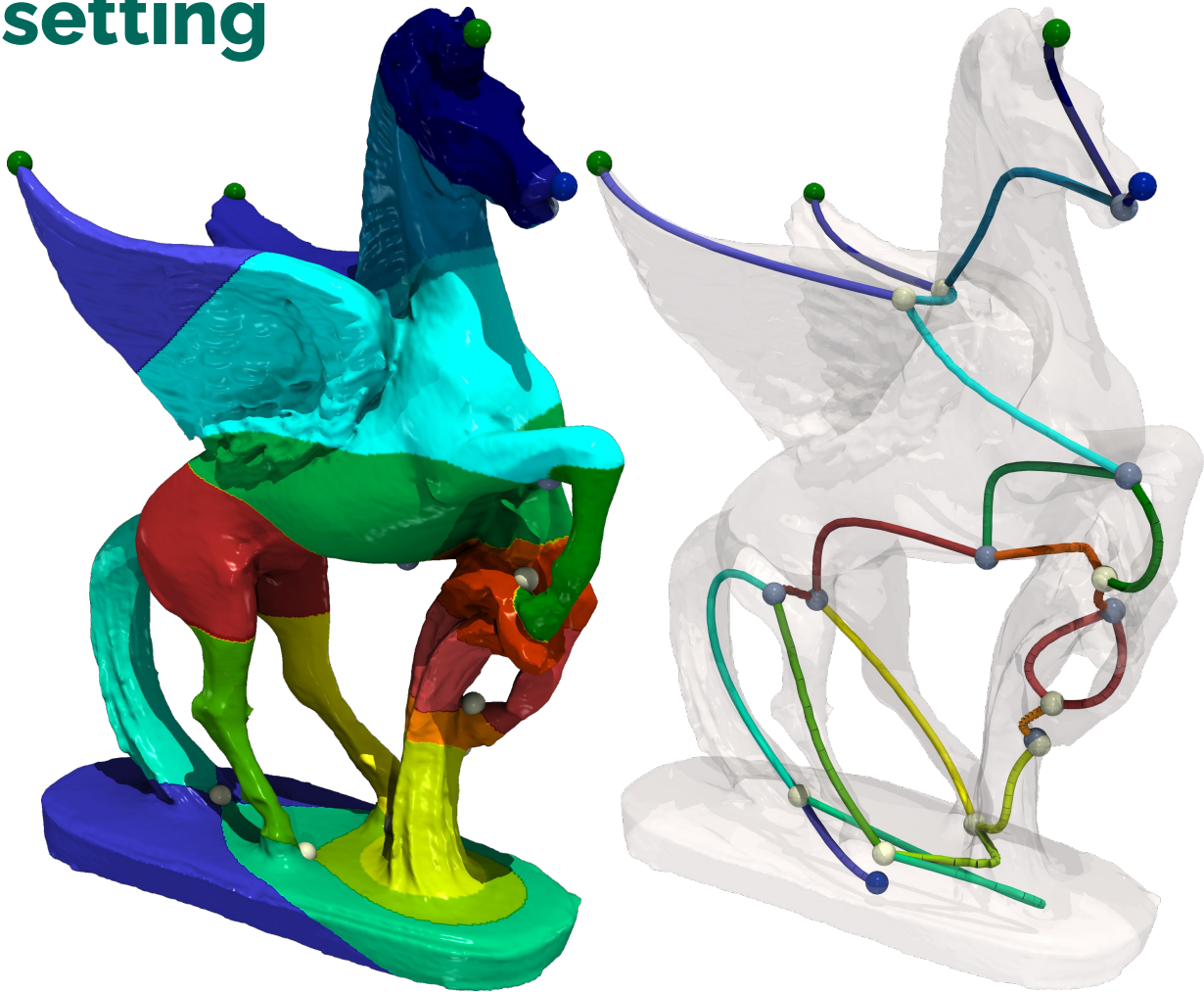
# Piecewise linear setting

- Input PL scalar data

- $f : \mathcal{M} \rightarrow \mathbb{R}$

- Topological abstractions

- Critical points
- Persistence diagrams
- Persistence curves
- Reeb graphs

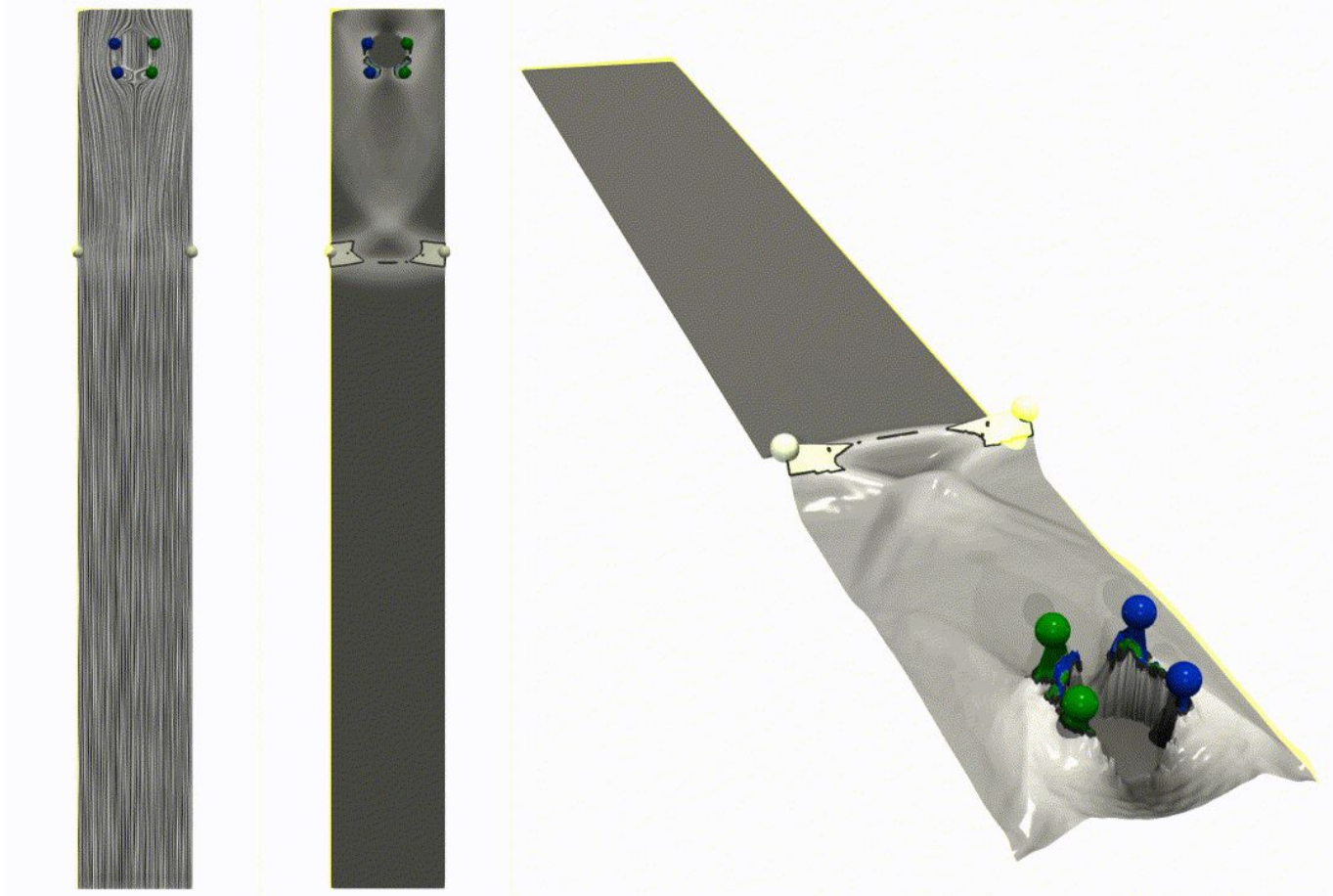




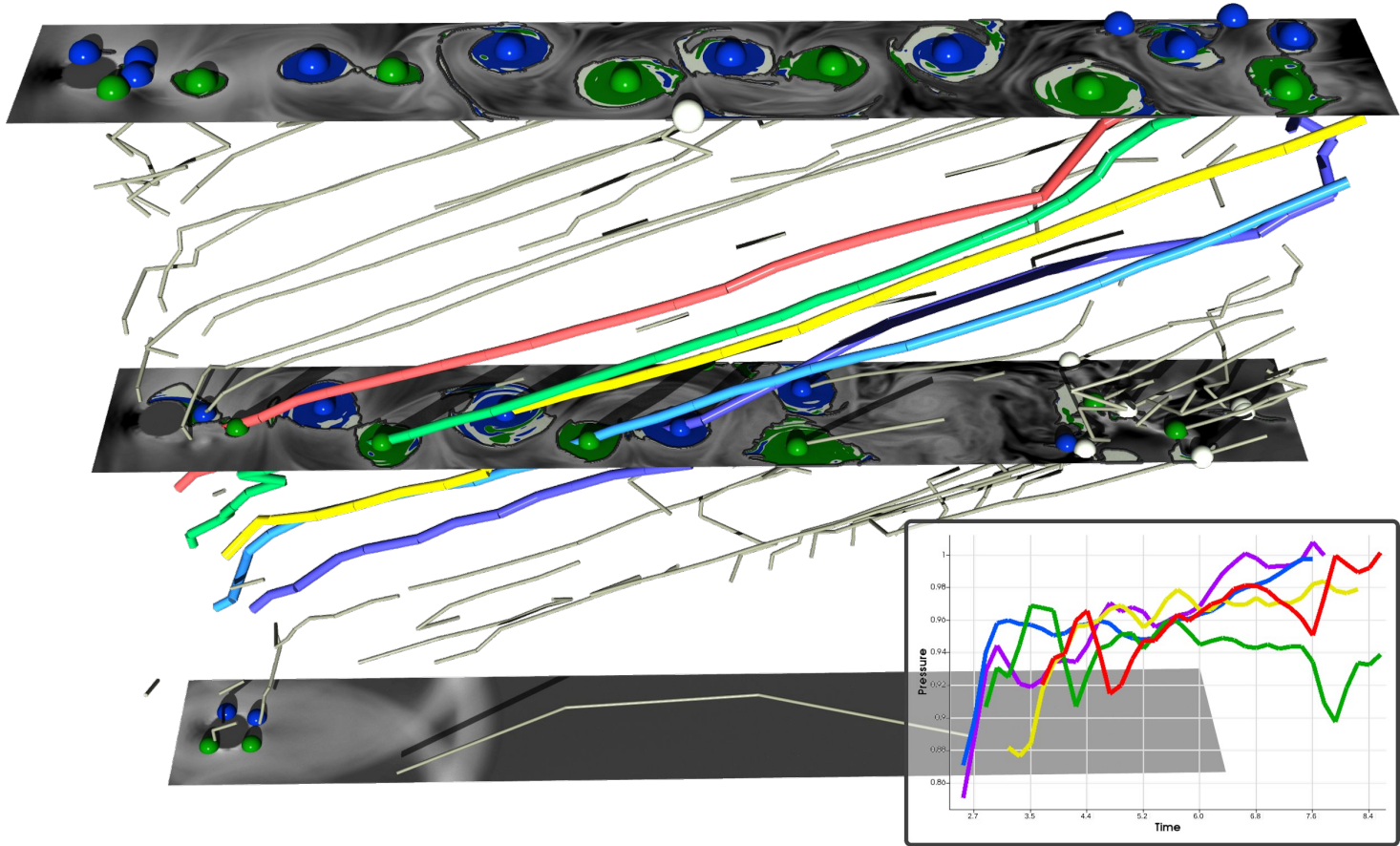




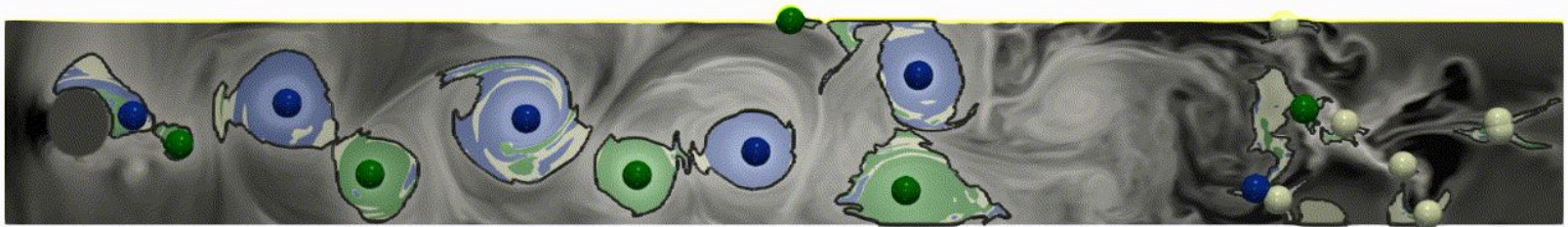
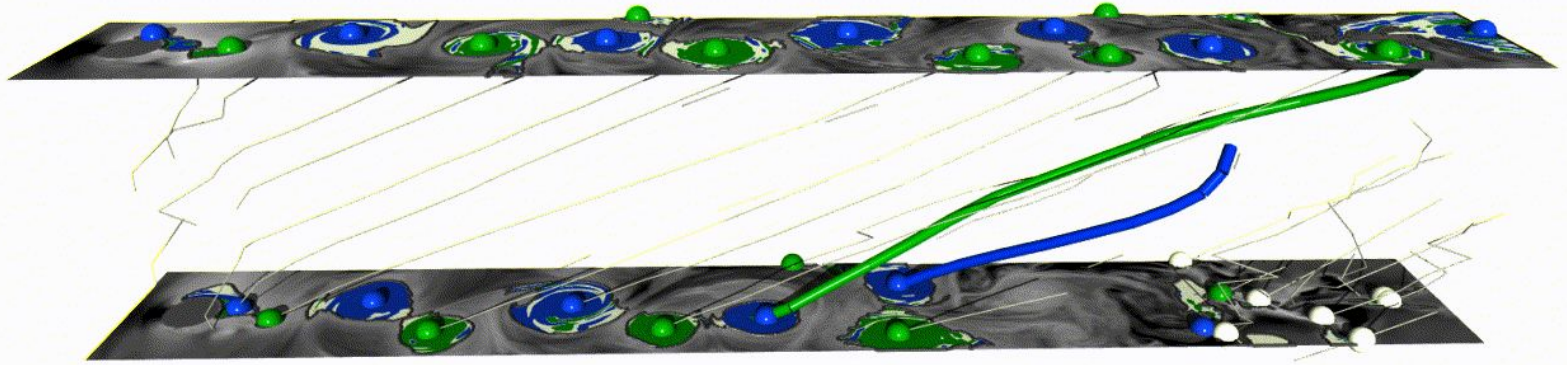
# Vortex extraction



# Vortex trajectory analysis



# Vortex tracking



# Reeb graphs

- **Vertex based contouring**

- Shinagawa and Kunii 1991

- **Quantized range contouring**

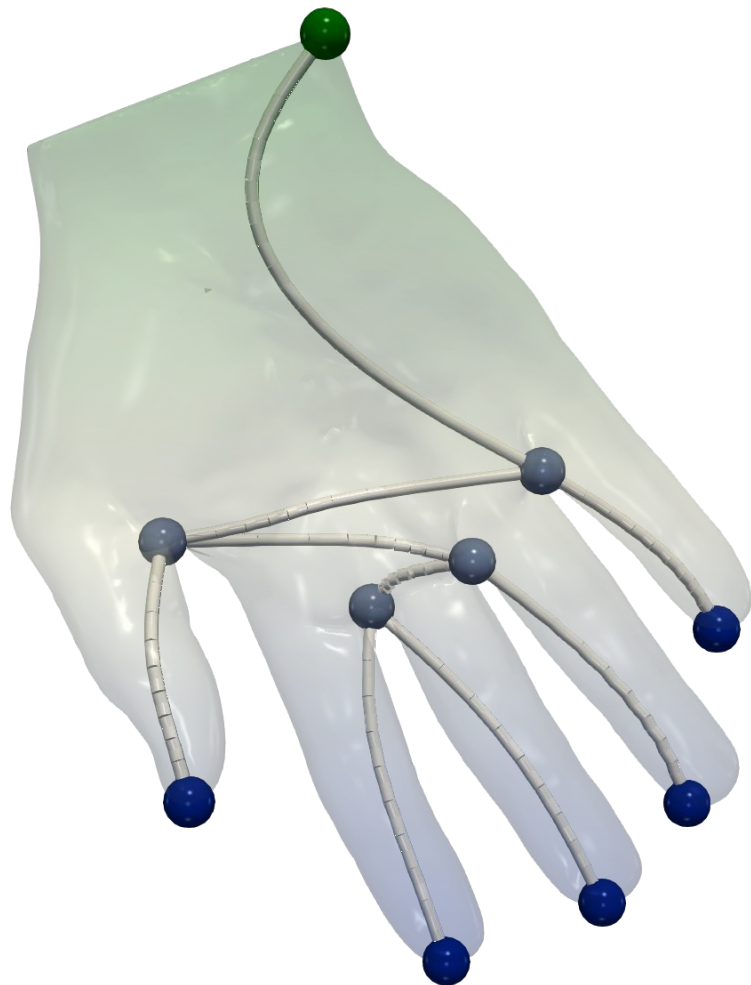
- Biasotti et al. 2000, Hilaga et al. 2001, Wood et al. 2004

- **Critical contouring**

- Patane et al. 2008, *Tierny et al. 2009*, Doraiswamy and Natarajan 2013, Hajji and Rosen 2018

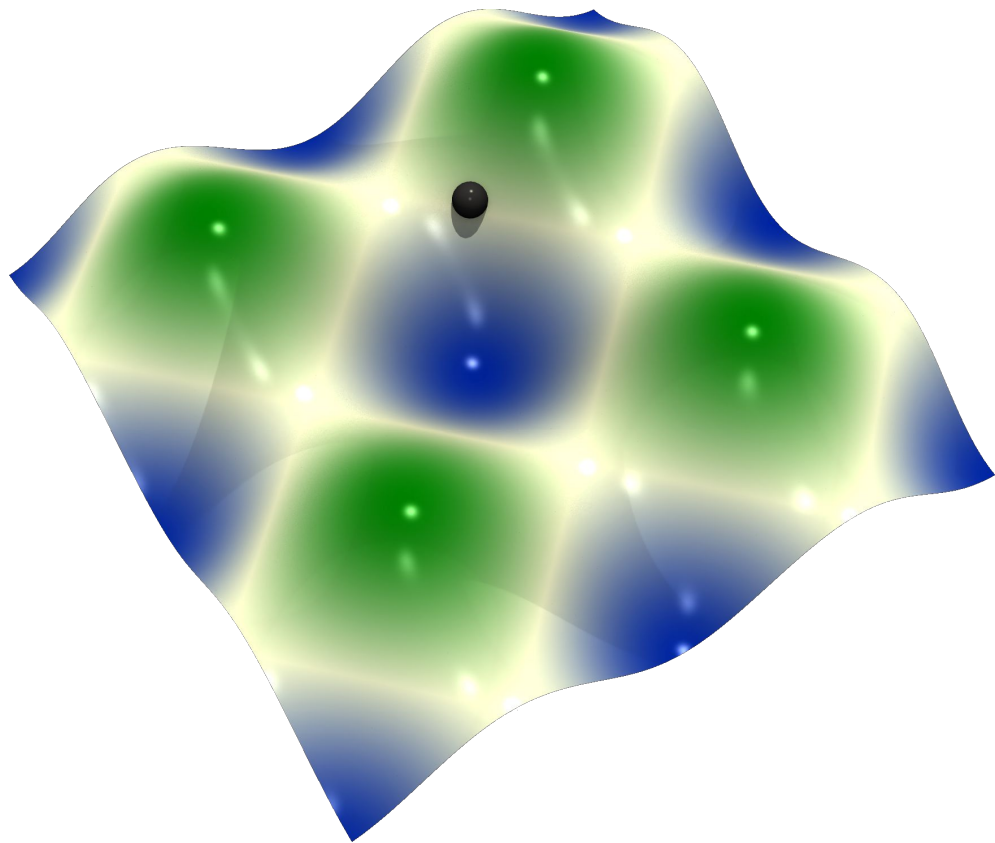
- **Dynamic connectivity**

- Cole-McLaughlin et al. 2003, Pascucci et al. 2007, Doraiswamy and Natarajan 2009, Parsa 2013, *Gueunet et al. 2019*



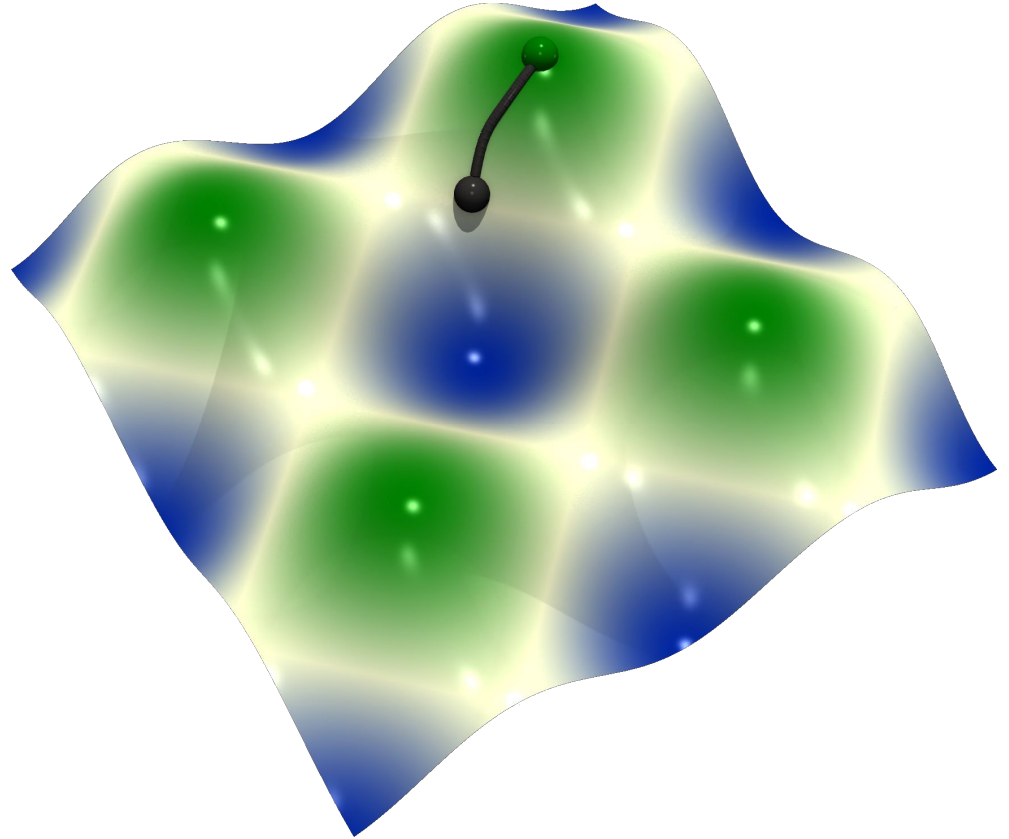
# Discrete Morse Theory

- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation



# Discrete Morse Theory

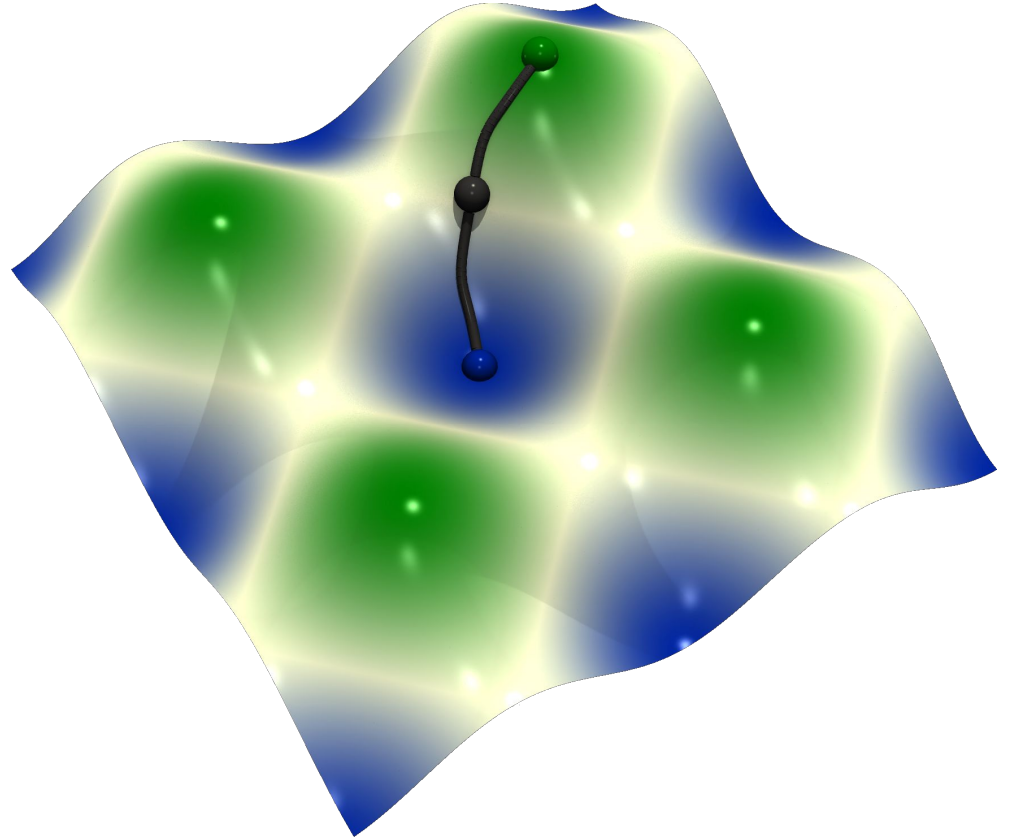
- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation





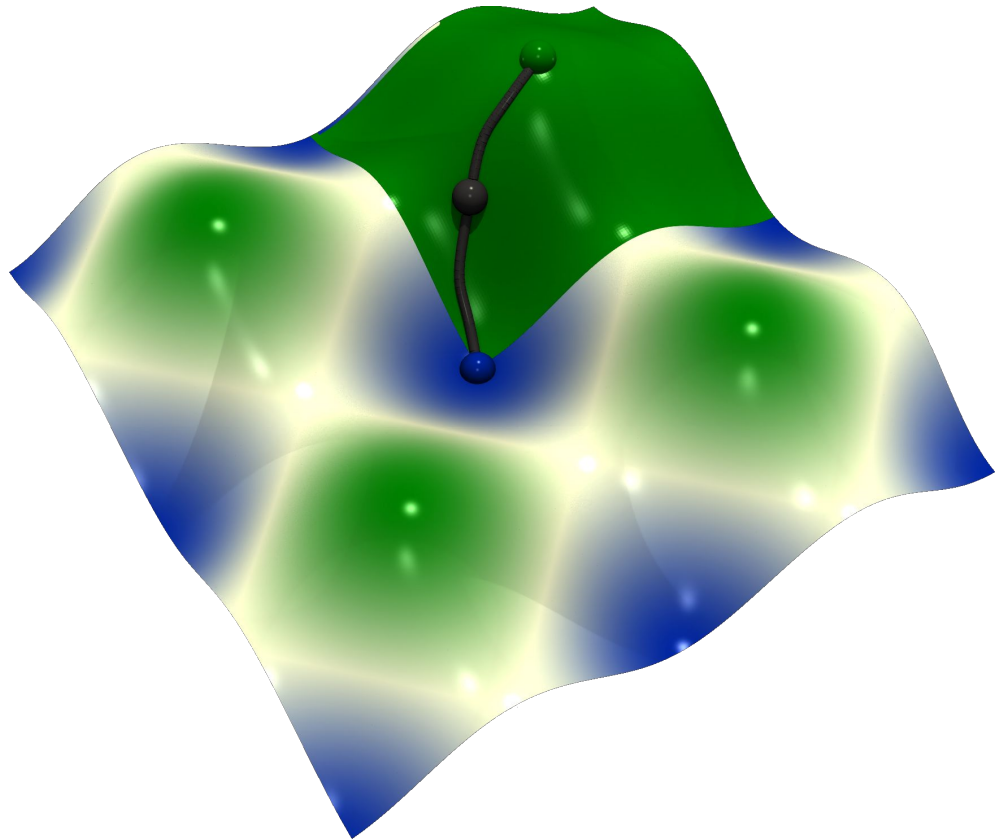
# Discrete Morse Theory

- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation



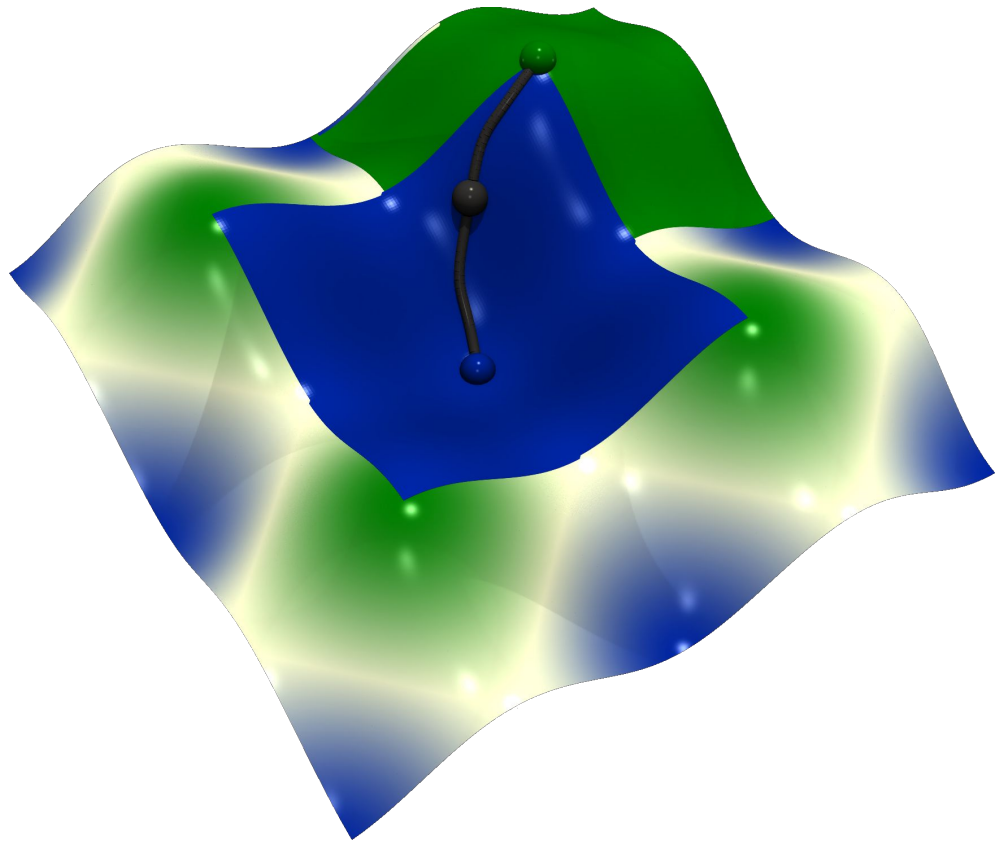
# Discrete Morse Theory

- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation



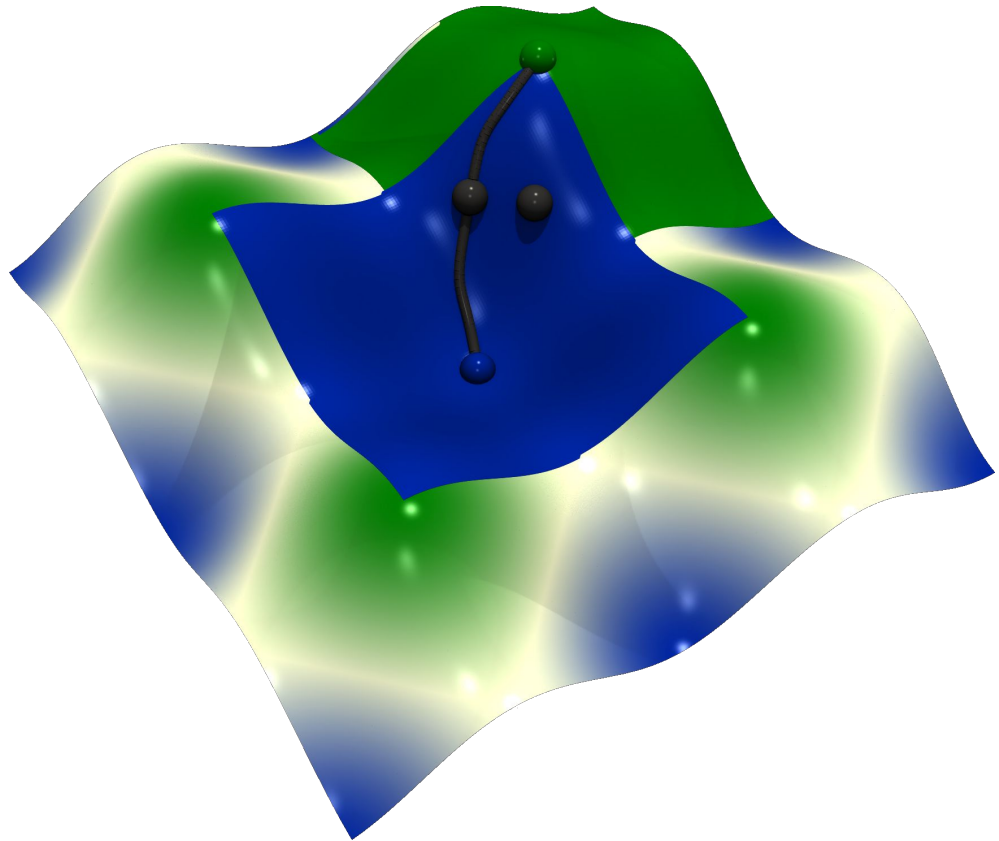
# Discrete Morse Theory

- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation



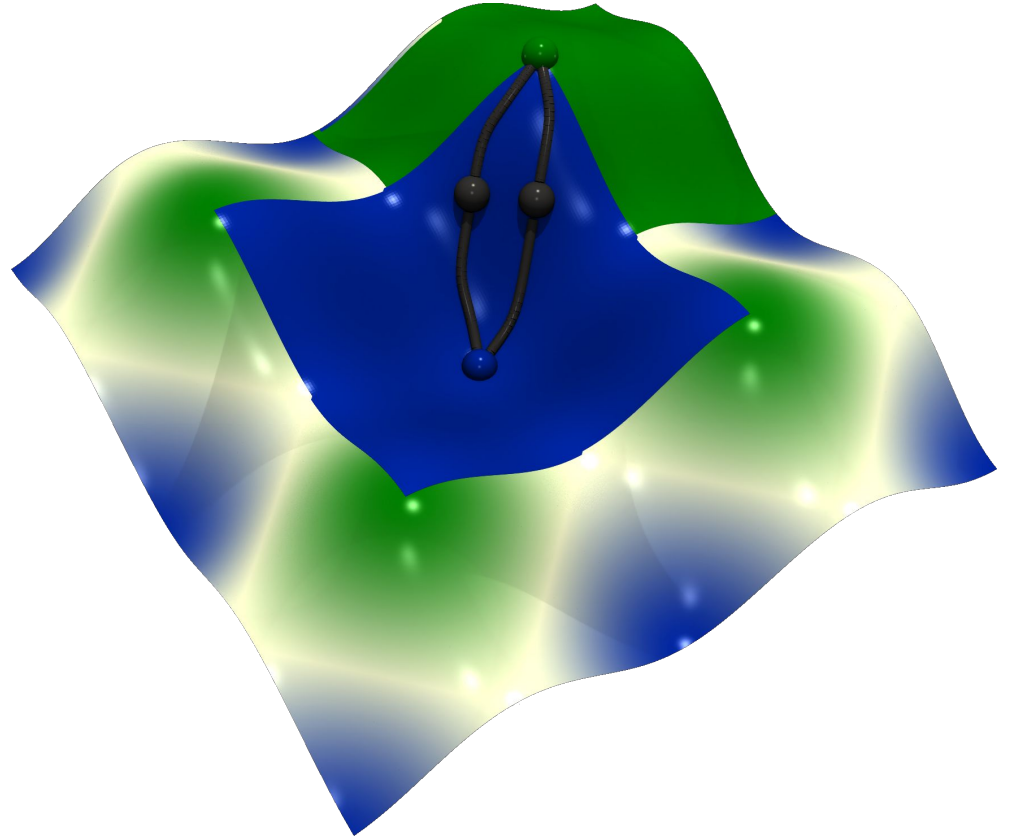
# Discrete Morse Theory

- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation



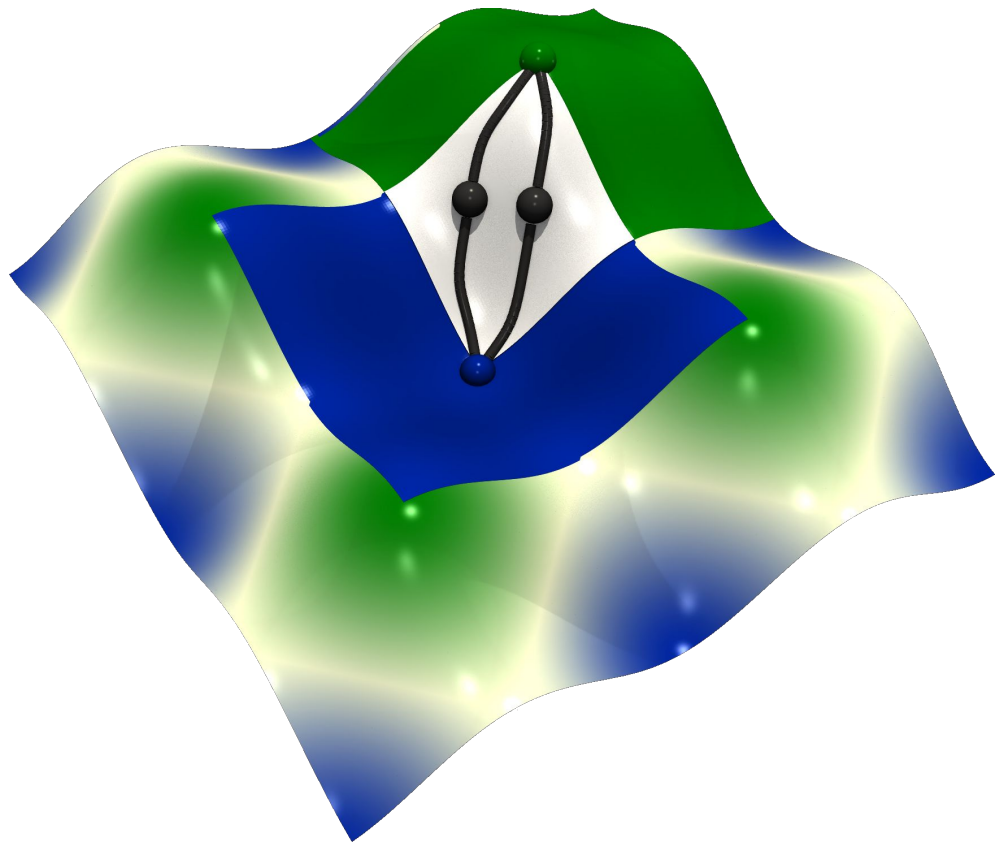
# Discrete Morse Theory

- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation



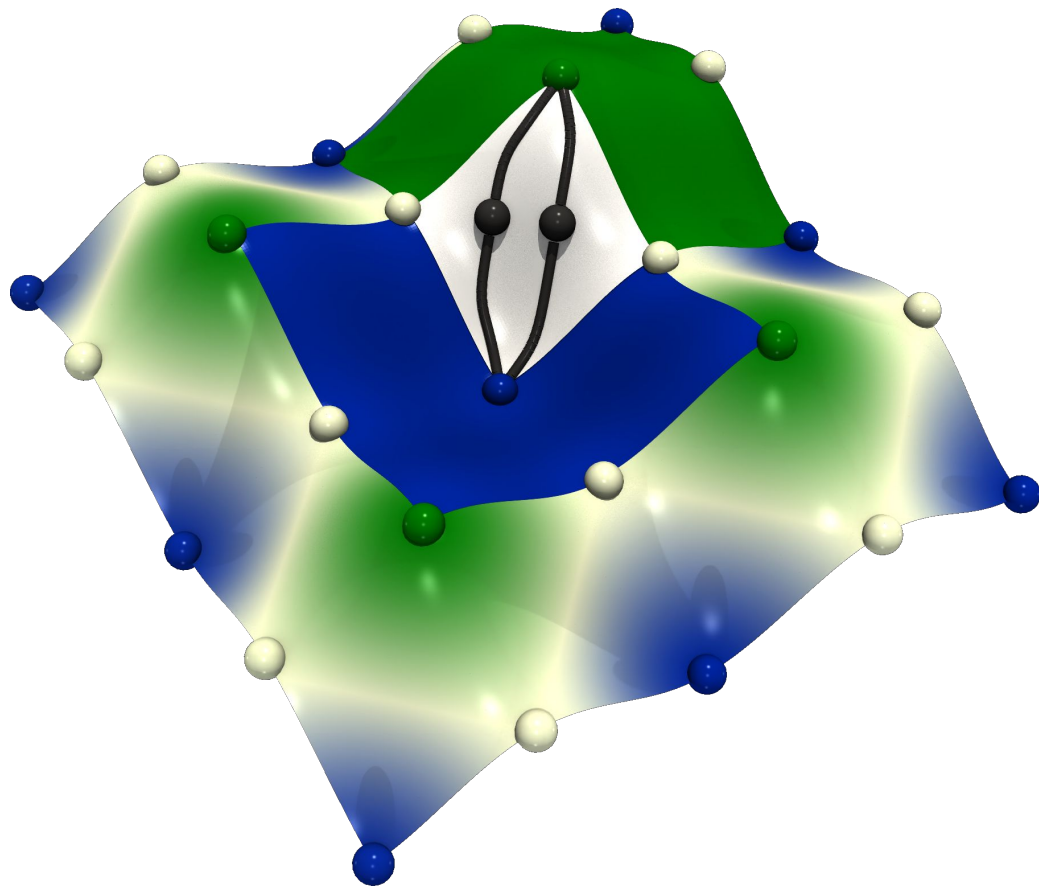
# Discrete Morse Theory

- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation



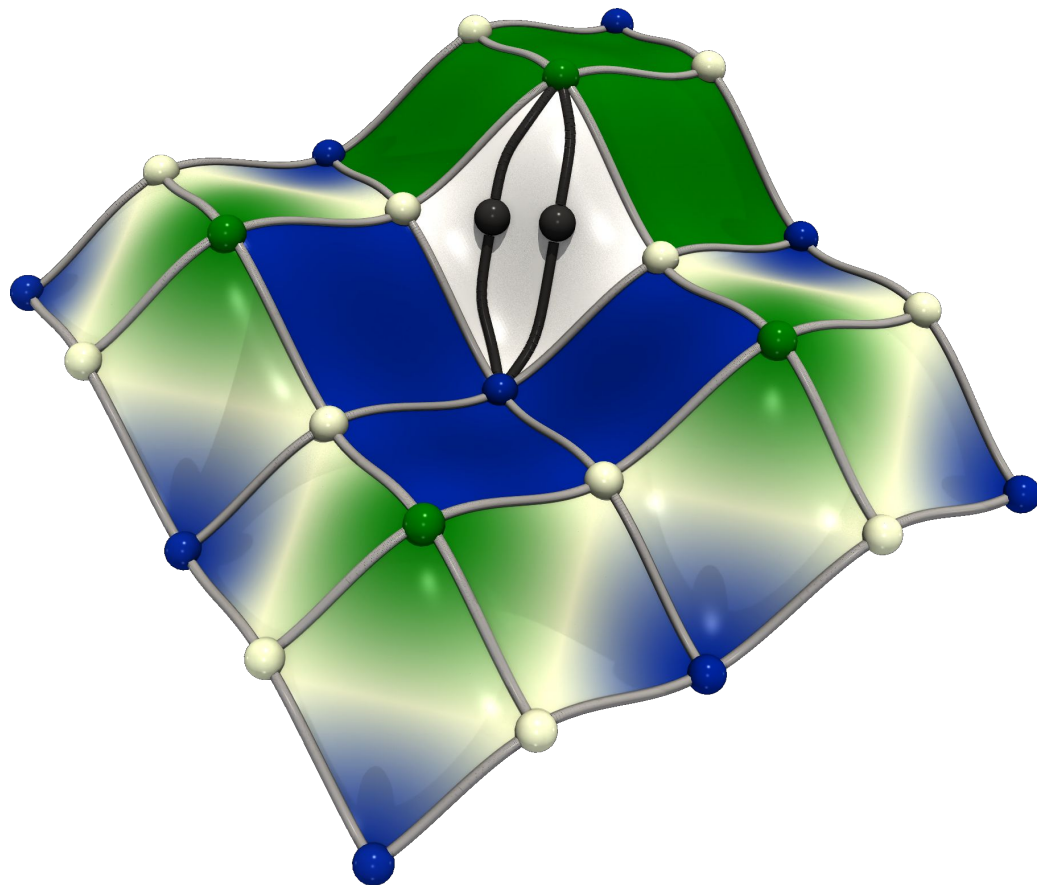
# Discrete Morse Theory

- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation



# Discrete Morse Theory

- **Morse-Smale complex**
  - Integration equivalence
  - Challenging PL computation





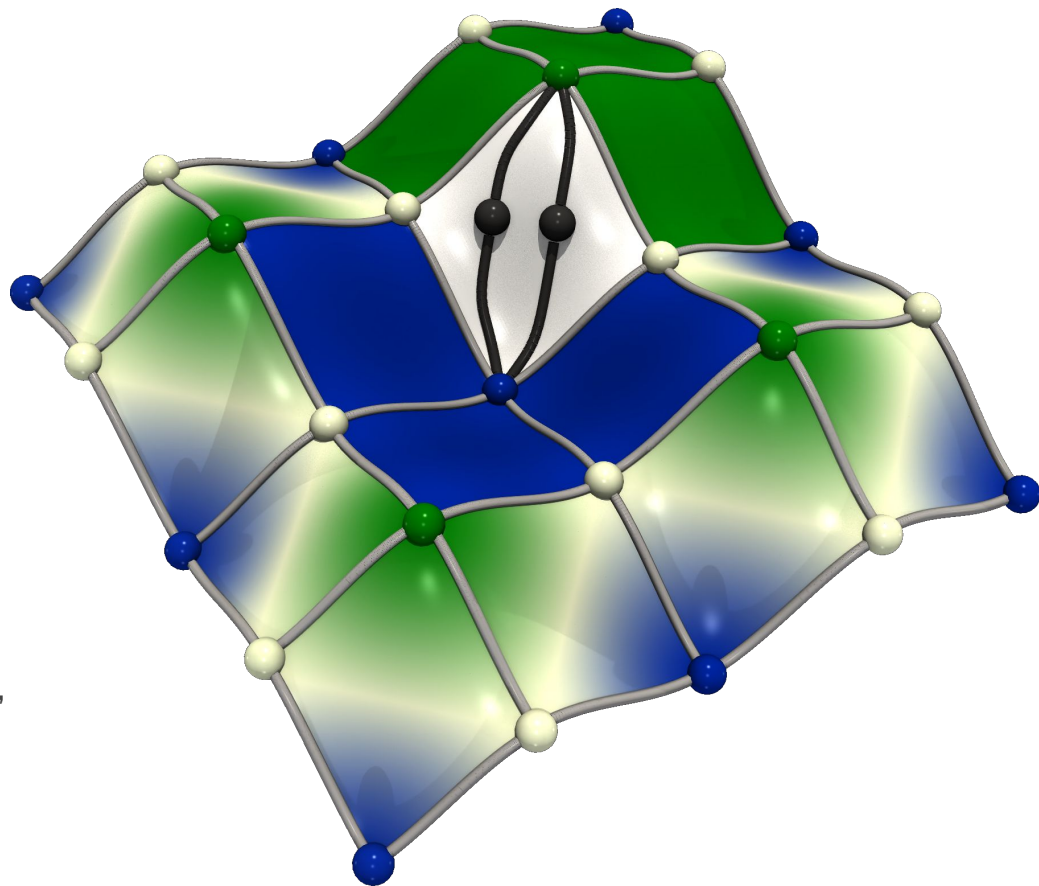
# Discrete Morse Theory

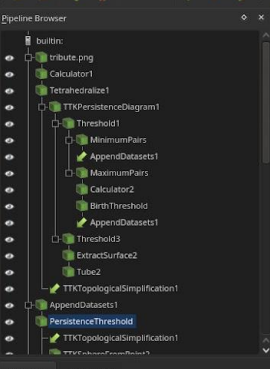
- **Morse-Smale complex**

- Integration equivalence
- Challenging PL computation

- **Discrete Morse theory**

- Forman 1998
- Algorithms
- Gyulassy 2008, Robins 2011, Shivashankar and Natarajan 2012, *Tierny et al. 2017*





Properties Information

Search... (Use Esc to clear text)

Properties (PersistenceT...

Scalars Persistence

Minimum 8.5  
Maximum 102.106426713982

Use Continuous Cell Range

Display

View (Render View)

Axes Grid Edit

Center Axes Visibility

Orientation Axes

Orientation Axes Visibility

Orientation Axes Interactivity

Orientation Axes Label Color

Orientation Axes Outline Color

Lights

Edit

Hidden Line Removal

Camera Parallel Projection

Background

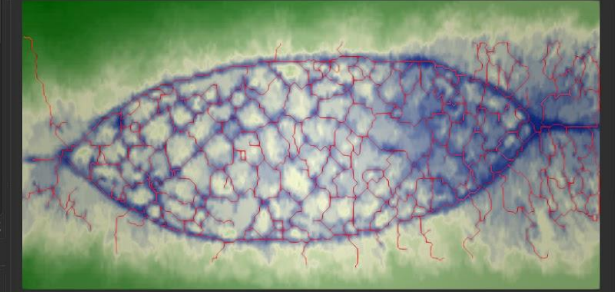
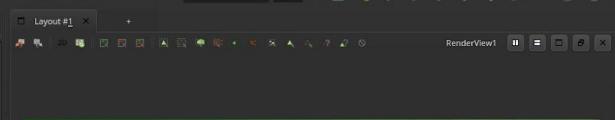
TTKPersistenceDiagram starting computation on field 'originalData'...

[OneSkeleton] Edge-list built in 0.00518210 s. (5143 edges, 1 thread(s)).

[ZeroSkeleton] one-skeleton built in 0.01267 s. (24 thread(s)).

[ZeroSkeleton] Vertex stars built in 0.0032597 s. (1 thread(s)).

[OneSkeleton] Edge stars built in 0.013261 s. (5143 edges, 24 thread(s)).



debug lvl : 0  
tree type : Join + Split  
[FTM] number of threads : 24  
[FTM] alloc in 0.0021098 s.  
[FTM] visit in 0.050292 s.  
[FTM] sort step in 0.024229 s.  
[FTM] leafSearch JT in 0.00219888 s.  
[FTM] leafSearch ST in 0.02939496 s.  
[FTM] trunk JT in 0.00120595 s.  
[FTM] leafSearch ST in 0.00151191 s.  
[FTM] leafSearch ST in 0.003072 s.  
[FTM] merge trees in 0.00218885 s.  
[FTM] trunk ST in 0.0724711 s.  
[FTM] build tree in 0.072525 s.  
[FTM] total in 0.133837 s.

[[tkPersistenceDiagram] Memory usage: 418.897 MB.

[OneSkeleton] Edge-list built in 0.00527293 s. (5143 edges, 1 thread(s)).

[ZeroSkeleton] one-skeleton built in 0.012229 s. (24 thread(s)).

[TopologicalSimplification] Scalar field simplified in 0.0026028 s. (24 thread(s)).

[TTKTopologicalSimplification] Memory usage: 0 MB.

[OneSkeleton] Edge-list built in 0.00508398 s. (5143 edges, 1 thread(s)).

[VertexStars] Vertex stars built in 0.0034984 s. (11 thread(s)).

[OneSkeleton] Edge stars built in 0.0173883 s. (5143 edges, 24 thread(s)).

[ZeroSkeleton] one-skeleton built in 0.006054 s. (24 thread(s)).

[ZeroSkeleton] Vertex edges built in 0.00049499 s. (1 thread(s)).

[ThreeSkeleton] Cell edges built in 0.00861311 s. (24 thread(s)).

[TwoSkeleton] Cell neighbors (69516 cells) computed in 0.0189598 s. (24 thread(s)).

[[tkMorseSmaleComplex] launching computation on field 'originalData'...

[DiscreteOriented] Data-set (30028 points) processed in 0.0143471 s. (24 thread(s)).

[DiscreteOriented] Data-set (30028 points) post-processed in 0.00272393 s. (24 thread(s)).

[ScalarFieldCriticalPoints] 240 minima.

[ScalarFieldCriticalPoints] 380 saddle(s).

[ScalarFieldCriticalPoints] 2 multi-saddle(s).

[ScalarFieldCriticalPoints] 150 maxima.

[ScalarFieldCriticalPoints] Data-set (38028 vertices) processed in 0.0100078 s. (24 thread(s)).

[DiscreteOriented] 240 0-cell(s) and 235 interior PL.

[DiscreteOriented] 3812 2-cell(s) and 375 interior PL.

[DiscreteOriented] 2772 2-cell(s) and 247 interior PL.

[DiscreteOriented] Initialization step : 0.0107951 s.

[DiscreteOriented] Ordering of the wpaths : 0.00130406 s.

[DiscreteOriented] Processing of the wpaths : 0.00390196 s.

[DiscreteOriented] Gradient reversal step : 0.00027895 s.

[DiscreteOriented] Saddle-Maximum pairs simplified in 0.0228741 s, 24 thread(s).

[DiscreteOriented] Initialization step : 0.01052 s.

[DiscreteOriented] Ordering of the wpaths : 1.00073e-05 s.

[DiscreteOriented] Processing of the wpaths : 1.28740e-05 s.

[DiscreteOriented] Gradient reversal step : 0.000728e-05 s.

[DiscreteOriented] Saddle-Maximum pairs simplified in 0.010097 s, 24 thread(s).

[DiscreteOriented] 240 0-cell(s).

[DiscreteOriented] 380 2-cell(s).

[DiscreteOriented] 147 2-cell(s).

[[MorseSmaleComplex] Data-set (30028 points) processed in 0.104118 s. (24 thread(s)).

[[tkMorseSmaleComplex] Memory usage: 7.76307 MB.

[[tkIdentifierRandomizer] Shuffling vertex field 'AscendingManifold'...

[[tkIdentifierRandomizer] Memory usage: 0 MB.

[[tkSphereFromPoint] Spheres computed in 0.136837 s.

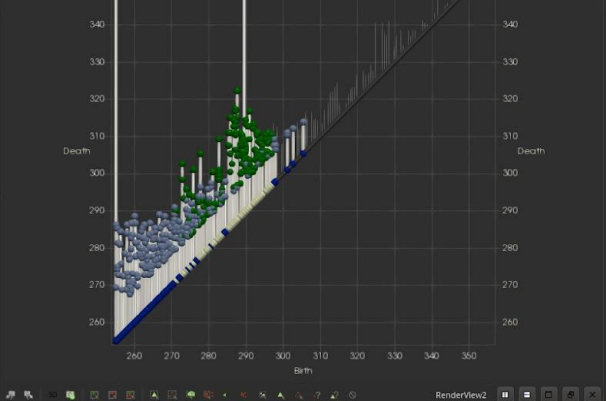
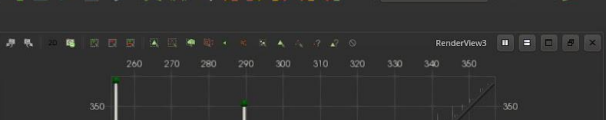
[[tkSphereFromPoint] Memory usage: 10.0962 MB.

[[tkSphereFromPoint] Spheres computed in 0.119745 s.

[[tkSphereFromPoint] Memory usage: 14.1875 MB.

[[tkSphereFromPoint] Spheres computed in 0.137463 s.

[[tkSphereFromPoint] Memory usage: 10.5538 MB.



[[tkPersistenceDiagram] Memory usage: 418.897 MB.

[OneSkeleton] Edge-list built in 0.00518210 s. (5143 edges, 1 thread(s)).

[ZeroSkeleton] one-skeleton built in 0.01267 s. (24 thread(s)).

[ZeroSkeleton] Vertex stars built in 0.0032597 s. (1 thread(s)).

[OneSkeleton] Edge stars built in 0.013261 s. (5143 edges, 24 thread(s)).

debug lvl : 0  
tree type : Join + Split  
[FTM] number of threads : 24  
[FTM] alloc in 0.0021098 s.  
[FTM] visit in 0.050292 s.  
[FTM] sort step in 0.024229 s.  
[FTM] leafSearch JT in 0.00219888 s.  
[FTM] leafSearch ST in 0.02939496 s.  
[FTM] trunk JT in 0.00120595 s.  
[FTM] leafSearch ST in 0.00151191 s.  
[FTM] leafSearch ST in 0.003072 s.  
[FTM] merge trees in 0.00218885 s.  
[FTM] trunk ST in 0.0724711 s.  
[FTM] build tree in 0.072525 s.  
[FTM] total in 0.133837 s.

[[tkPersistenceDiagram] Memory usage: 418.897 MB.

[OneSkeleton] Edge-list built in 0.00527293 s. (5143 edges, 1 thread(s)).

[ZeroSkeleton] one-skeleton built in 0.012229 s. (24 thread(s)).

[TopologicalSimplification] Scalar field simplified in 0.0026028 s. (24 thread(s)).

[TTKTopologicalSimplification] Memory usage: 0 MB.

[OneSkeleton] Edge-list built in 0.00508398 s. (5143 edges, 1 thread(s)).

[VertexStars] Vertex stars built in 0.0034984 s. (11 thread(s)).

[OneSkeleton] Edge stars built in 0.0173883 s. (5143 edges, 24 thread(s)).

[ZeroSkeleton] one-skeleton built in 0.006054 s. (24 thread(s)).

[ZeroSkeleton] Vertex edges built in 0.00049499 s. (1 thread(s)).

[ThreeSkeleton] Cell edges built in 0.00861311 s. (24 thread(s)).

[TwoSkeleton] Cell neighbors (69516 cells) computed in 0.0189598 s. (24 thread(s)).

[[tkMorseSmaleComplex] launching computation on field 'originalData'...

[DiscreteOriented] Data-set (30028 points) processed in 0.0143471 s. (24 thread(s)).

[DiscreteOriented] Data-set (30028 points) post-processed in 0.00272393 s. (24 thread(s)).

[ScalarFieldCriticalPoints] 240 minima.

[ScalarFieldCriticalPoints] 380 saddle(s).

[ScalarFieldCriticalPoints] 2 multi-saddle(s).

[ScalarFieldCriticalPoints] 150 maxima.

[ScalarFieldCriticalPoints] Data-set (38028 vertices) processed in 0.0100078 s. (24 thread(s)).

[DiscreteOriented] 240 0-cell(s) and 235 interior PL.

[DiscreteOriented] 3812 2-cell(s) and 375 interior PL.

[DiscreteOriented] 2772 2-cell(s) and 247 interior PL.

[DiscreteOriented] Initialization step : 0.0107951 s.

[DiscreteOriented] Ordering of the wpaths : 0.00130406 s.

[DiscreteOriented] Processing of the wpaths : 0.00390196 s.

[DiscreteOriented] Gradient reversal step : 0.00027895 s.

[DiscreteOriented] Saddle-Maximum pairs simplified in 0.0228741 s, 24 thread(s).

[DiscreteOriented] Initialization step : 0.01052 s.

[DiscreteOriented] Ordering of the wpaths : 1.00073e-05 s.

[DiscreteOriented] Processing of the wpaths : 1.28740e-05 s.

[DiscreteOriented] Gradient reversal step : 0.000728e-05 s.

[DiscreteOriented] Saddle-Maximum pairs simplified in 0.010097 s, 24 thread(s).

[DiscreteOriented] 240 0-cell(s).

[DiscreteOriented] 380 2-cell(s).

[DiscreteOriented] 147 2-cell(s).

[[MorseSmaleComplex] Data-set (30028 points) processed in 0.104118 s. (24 thread(s)).

[[tkMorseSmaleComplex] Memory usage: 7.76307 MB.

[[tkIdentifierRandomizer] Shuffling vertex field 'AscendingManifold'...

[[tkIdentifierRandomizer] Memory usage: 0 MB.

[[tkSphereFromPoint] Spheres computed in 0.136837 s.

[[tkSphereFromPoint] Memory usage: 10.0962 MB.

[[tkSphereFromPoint] Spheres computed in 0.119745 s.

[[tkSphereFromPoint] Memory usage: 14.1875 MB.

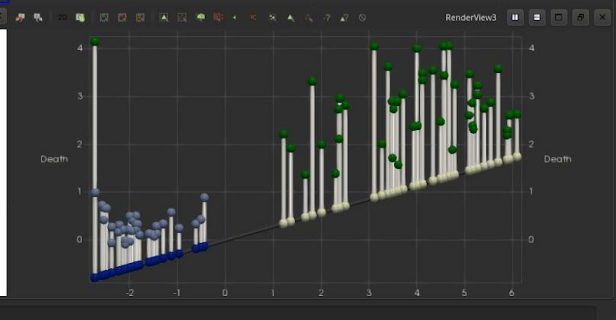
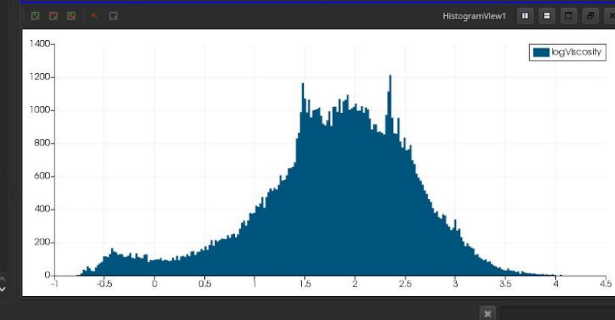
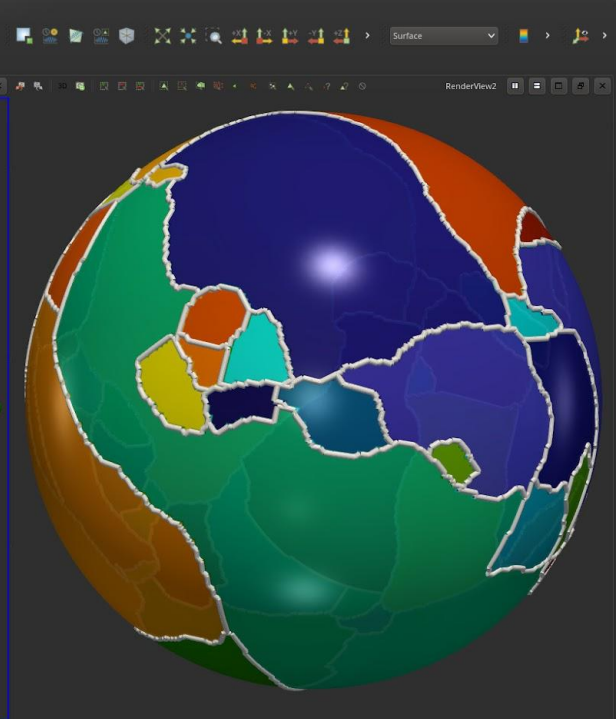
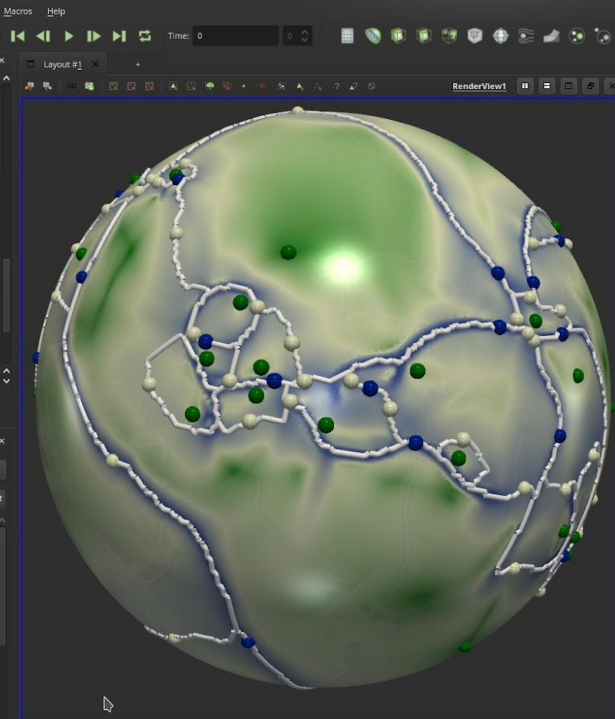
[[tkSphereFromPoint] Spheres computed in 0.137463 s.

[[tkSphereFromPoint] Memory usage: 10.5538 MB.

```

[[[TKMorseSmaleComplex]]] Memory usage: 21.0898 MB.
[[[TKSphereFromPoint]]] Spheres computed in 0.048862 s.
[[[TKSphereFromPoint]]] Memory usage: 3.03072 MB.
[[[TopologicalSimplification]]] Scalar field simplified in 0.300071 s. (24 threads)
1 s.
[[[TKTopologicalSimplification]]] Memory usage: 0.0005409 MB.
[[[TKPersistenceDiagram]]] starting computation on field 'logViscosity'...
-----
[FTM] number of threads : 24
[FTM] debug lvl : 3
[FTM] tree type : Join + Split
-----
[FTM] alloc in 0.0802586
[FTM] init in 0.080219
[FTM] sort step in 0.0300721
[FTM] leafSearch JT in 0.016167
[FTM] leafSearch JT in 0.00051106
[FTM] trunk JT in 0.00100313
[FTM] leafSearch ST in 0.0110941
[FTM] leafSearch ST in 0.0223270
[FTM] trunk ST in 0.00401906
[FTM] merge trees in 0.0560519
[FTM] build tree in 0.0557601
[FTM] total in 0.174975
[[[TKPersistenceDiagram]]] Memory usage: 2.03379 MB.
[[[TKSphereFromPoint]]] Spheres computed in 0.018751 s.
[[[TKSphereFromPoint]]] Memory usage: 0 MB.
[[[TopologicalSimplification]]] Scalar field simplified in 0.348824 s. (24 threads)
1 s.
[[[TKTopologicalSimplification]]] Memory usage: 0.203072 MB.
[[[TKMorseSmaleComplex]]] Launching computation on field 'logViscosity'...
[[[DiscreteGradient]]] Data-set (92050 points) processed in 0.030001 s. (24 threads)
1 s.
[[[DiscreteGradient]]] Data-set (92050 points) post-processed in 0.022071 s. (24 threads)
1 s.
[[[ScalarFieldCriticalPoints]]] 35 minima.
[[[ScalarFieldCriticalPoints]]] 79 saddle(s).
[[[ScalarFieldCriticalPoints]]] 0 multi-cell(s).
[[[ScalarFieldCriticalPoints]]] 46 maxima.
[[[ScalarFieldCriticalPoints]]] Data-set (92050 vertices) processed in 0.046805 s. (24 threads)
1 s.
[[[DiscreteGradient]]] 35 0-cell(s) and 35 interior PL.
[[[DiscreteGradient]]] 1006 1-cell(s) and 79 interior PL.
[[[DiscreteGradient]]] 1003 2-cell(s) and 46 interior PL.
[[[DiscreteGradient]]] Initialization step : 0.0195749 s.
[[[DiscreteGradient]]] ordering of the vpaths : 0.00080080 s.
[[[DiscreteGradient]]] Processing of the vpaths : 0.00495028 s.
[[[DiscreteGradient]]] Gradient reversal step : 0.000150104 s.
[[[DiscreteGradient]]] Saddle-Maximum pairs simplified in 0.0306072 s, 24 thread(s).
[[[DiscreteGradient]]] Initialization step : 0.018811 s.
[[[DiscreteGradient]]] ordering of the vpaths : 4.05312e-06 s.
[[[DiscreteGradient]]] Processing of the vpaths : 4.3806e-06 s.
[[[DiscreteGradient]]] Gradient reversal step : 2.88532e-06 s.
[[[DiscreteGradient]]] Saddle-Maximum pairs simplified in 0.028371 s, 24 thread(s).
[[[DiscreteGradient]]] 35 0-cell(s).
[[[DiscreteGradient]]] 79 1-cell(s).
[[[DiscreteGradient]]] 46 2-cell(s).
[[[MorseSmaleComplex]]] Data-set (92050 points) processed in 0.240017 s. (24 threads)
1 s.
[[[TKMorseSmaleComplex]]] Memory usage: 3.85449 MB.
[[[TKIdentifierRandomizer]]] shuffling vertex field 'AscendingManifold'...
[[[TKIdentifierRandomizer]]] Memory usage: 0 MB.
[[[TKSphereFromPoint]]] Spheres computed in 0.048223 s.
[[[TKSphereFromPoint]]] Memory usage: 2.14648 MB.
[[[TKPersistenceDiagram]]] starting computation on field 'logViscosity'...
-----
[FTM] number of threads : 24
[FTM] debug lvl : 3
[FTM] tree type : Join + Split
-----
[FTM] alloc in 0.081122
[FTM] init in 0.0473981
[FTM] sort step in 0.0323330
[FTM] leafSearch JT in 0.0123391
[FTM] leafSearch JT in 0.00052110
[FTM] trunk JT in 0.00073307
[FTM] leafSearch ST in 0.0279401
[FTM] leafSearch ST in 0.0148571
[FTM] trunk ST in 0.00225984
[FTM] merge trees in 0.071516
[FTM] build tree in 0.071509
[FTM] total in 0.232124
[[[TKPersistenceDiagram]]] Memory usage: 0 MB.
[[[TKSphereFromPoint]]] Spheres computed in 0.0238731 s.
[[[TKSphereFromPoint]]] Memory usage: 1.4082 MB.
[[[TKSphereFromPoint]]] Memory usage: 1.31248 MB.
[[[TKSphereFromPoint]]] Spheres computed in 0.01808 s.
[[[TKSphereFromPoint]]] Spheres computed in 0.0192912 s.
[[[TKSphereFromPoint]]] Memory usage: 0 MB.
[[[TKSphereFromPoint]]] Spheres computed in 0.0162001 s.
[[[TKSphereFromPoint]]] Memory usage: 0 MB.
[[[TKSphereFromPoint]]] Spheres computed in 0.028055 s.
[[[TKSphereFromPoint]]] Spheres computed in 0.019981 s.
[[[TKSphereFromPoint]]] Memory usage: 0 MB.

```

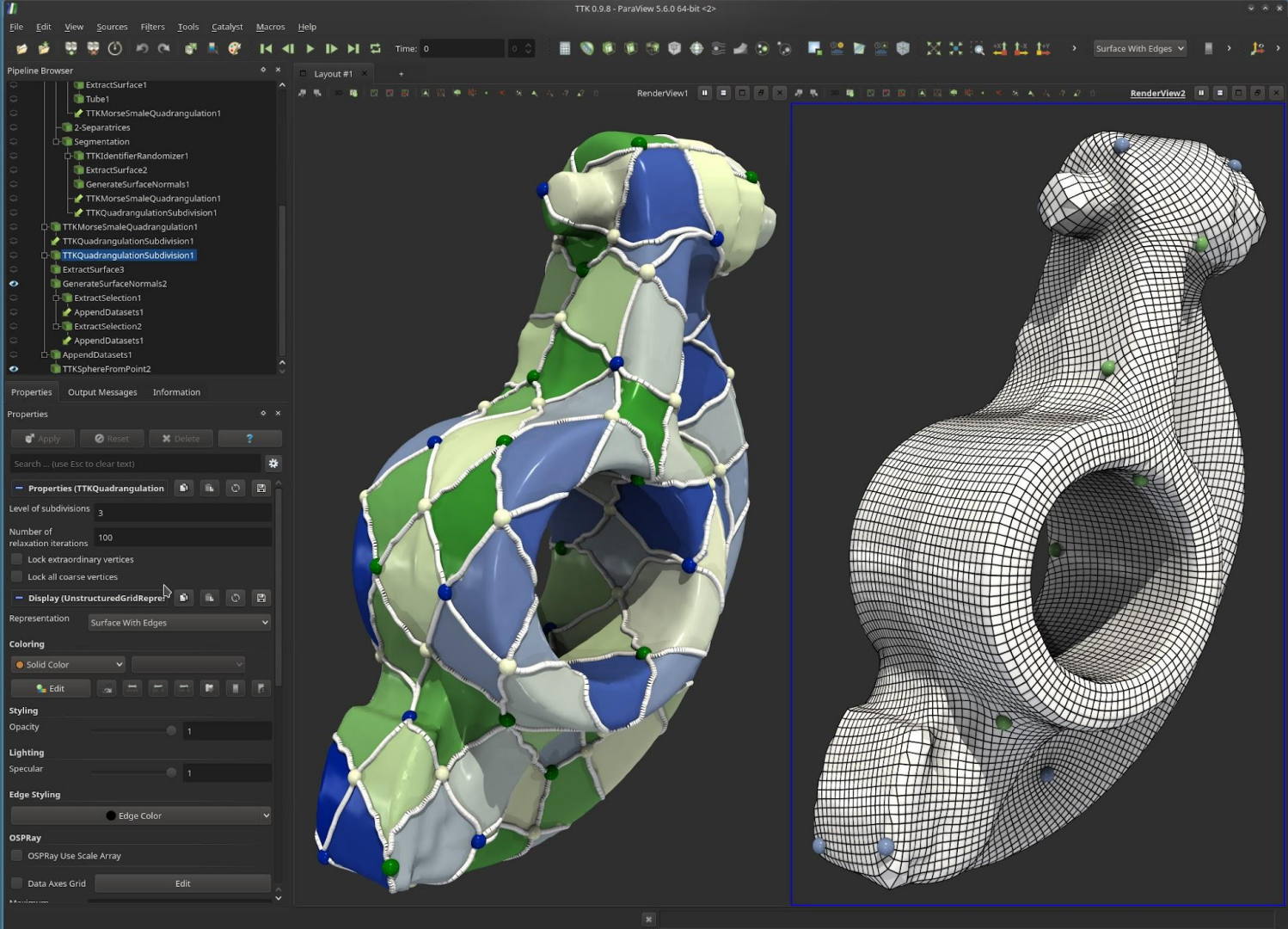




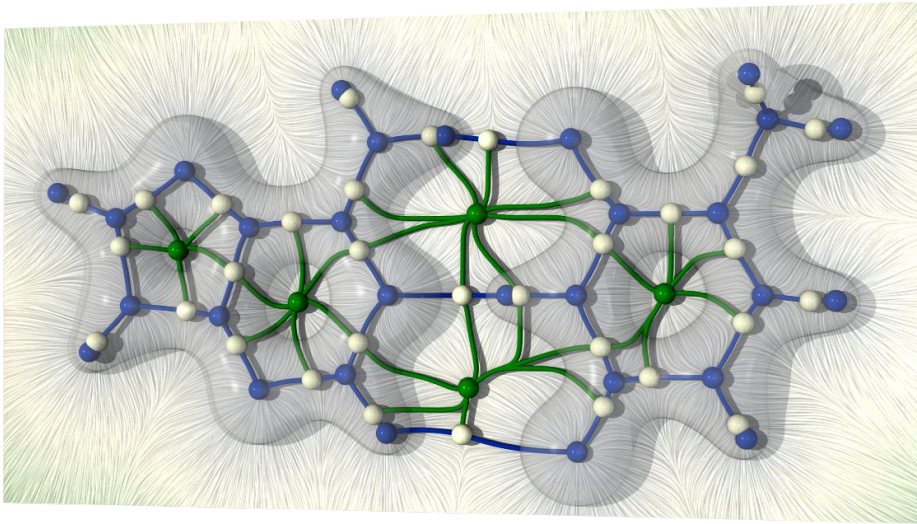
```

[QuadrangulationSubdivision] Projected 18800 points in 0.118999 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.115738 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.124929 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.121589 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.117924 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.118997 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.187913 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119172 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.121935 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.122115 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.128403 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.29398 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.115573 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.125102 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.24922 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119276 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.128124 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.122025 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.147813 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.122896 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.118922 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.183569 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.117079 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.147433 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.115774 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.115788 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.116981 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.144451 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.116994 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.113875 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.117408 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.11844 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.122712 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119288 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.127396 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.131415 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.115455 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.122755 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.118795 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.115685 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.126241 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119646 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.127306 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.118691 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119684 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.121679 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.117288 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.184658 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.118814 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.123936 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.124555 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.118932 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.128973 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.125905 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119997 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.138488 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.122786 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.116181 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.115856 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.11845 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119387 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.114929 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.111119 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119919 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.122592 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.129951 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.125955 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.124836 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.128388 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119174 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.117535 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.12122 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.119772 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.129839 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.131818 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.188819 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.897418 s.
[QuadrangulationSubdivision] Projected 18800 points in 0.898504 s.
[QuadrangulationSubdivision] Produced 32800 quadrangles with 13860 points in 21.452 s. (24 thread(s)).
ttkQuadrangulationSubdivision] Memory usage: 0 MB.
[ttkSphereFromPoint] Spheres computed in 0.0958389 s.
[ttkSphereFromPoint] Memory usage: 0 MB.
[ttkSphereFromPoint] Spheres computed in 0.018138 s.
[ttkSphereFromPoint] Memory usage: 0 MB.
[OneSkeleton] Edge-list built in 0.899734896 s. (12338 edges, 1 thread(s)).
[Zeroskeleton] One-skeleton built in 0.86203895 s. (24 thread(s)).
[ScalTriSmooth] Data set (12339 points) smoothed in 0.8888977 s. (24 thread(s)).
[ttkCountrySmother] Memory usage: 0 MB.
[ttkIdentifierRandomizer] Shuffling vertex field 'MorseSmallEManifold'...
[ttkIdentifierRandomizer] Memory usage: 0 MB.

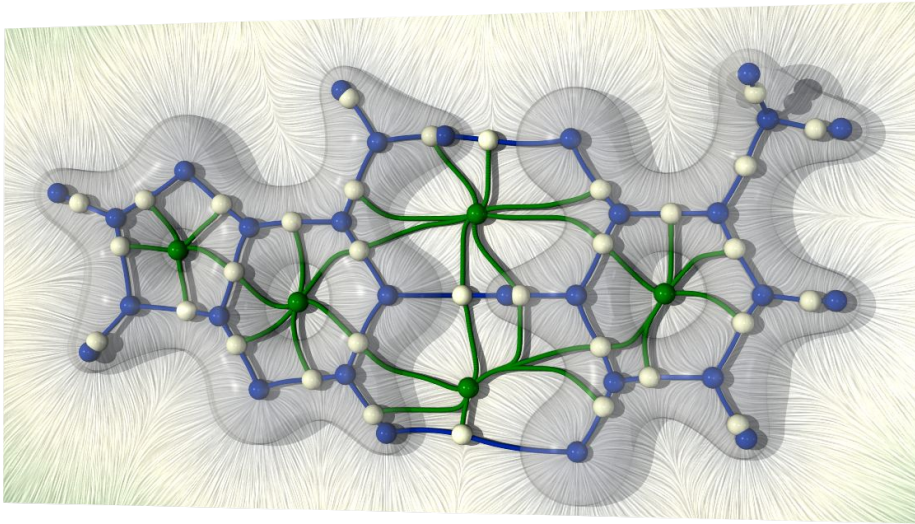
```



# Application of the Morse-Smale complex

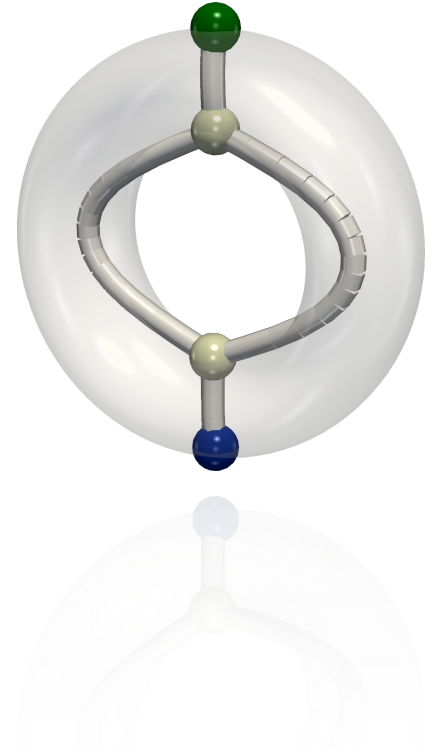


# Application of the Morse-Smale complex



# So far

- **TDA for low dimensional fields**
  - Data reduction by feature extraction

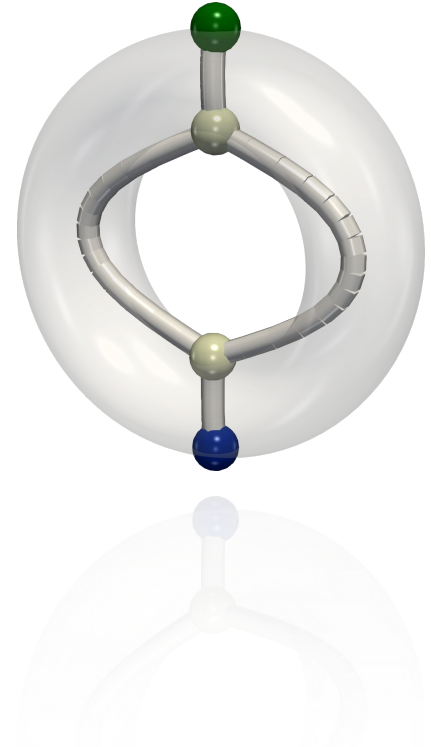




# So far

- **TDA for low dimensional fields**

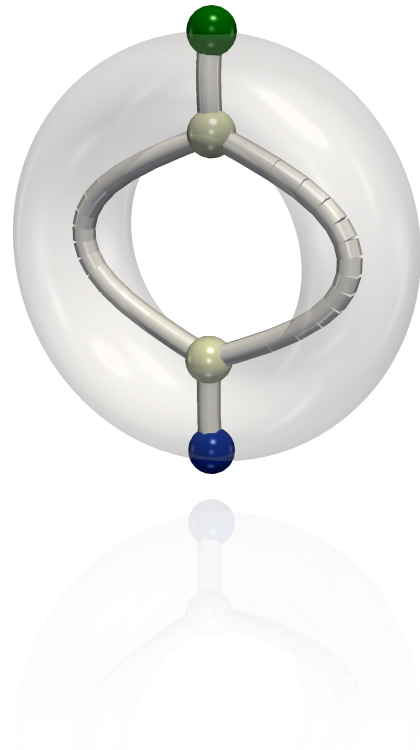
- Data reduction by feature extraction
  - Critical points (vortices)
  - 1-Separatrices (filament structures)
  - 2-Separatrices (walls)
  - Regions of interest



# So far

- **TDA for low dimensional fields**

- Data reduction by feature extraction
  - Critical points (vortices)
  - 1-Separatrices (filament structures)
  - 2-Separatrices (walls)
  - Regions of interest
- Only store topological information
  - Further analysis, measure, comparison
  - TDA driven lossy compression (*Soler et al. 2018*)

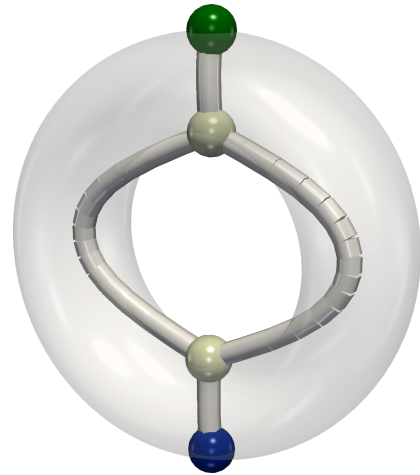


# So far

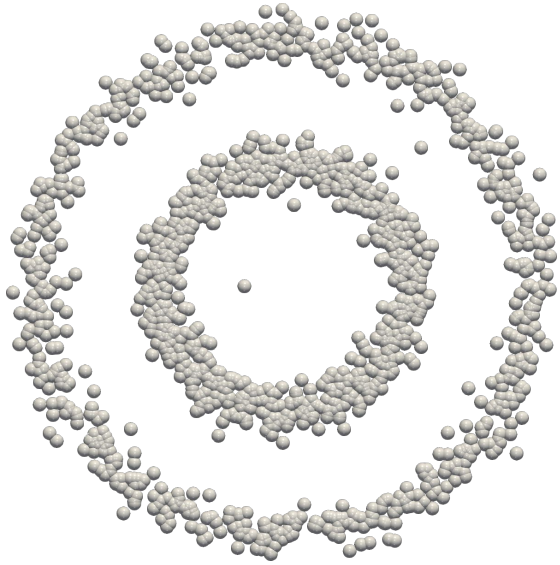
- **TDA for low dimensional fields**

- Data reduction by feature extraction
  - Critical points (vortices)
  - 1-Separatrices (filament structures)
  - 2-Separatrices (walls)
  - Regions of interest
- Only store topological information
  - Further analysis, measure, comparison
  - TDA driven lossy compression (*Soler et al. 2018*)

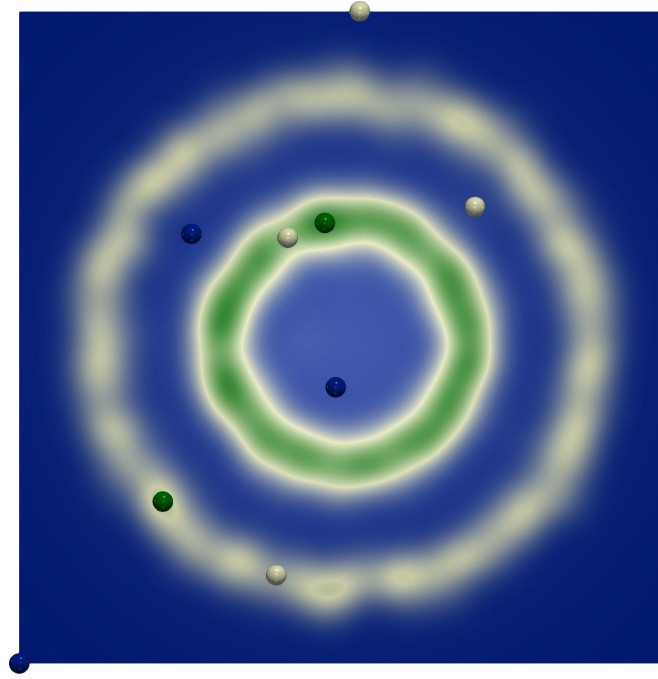
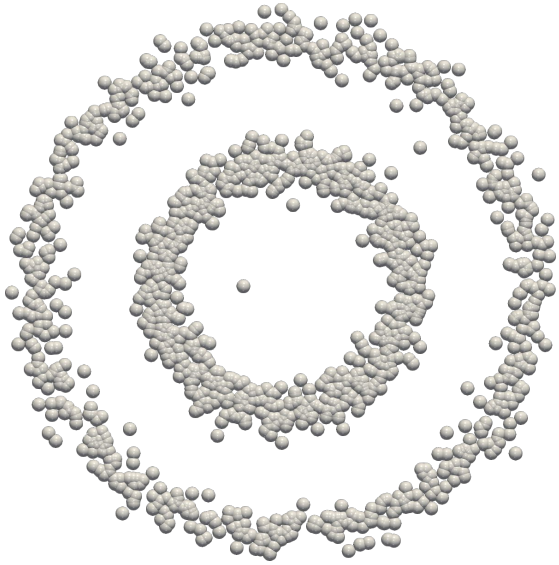
- **How about point cloud data?**



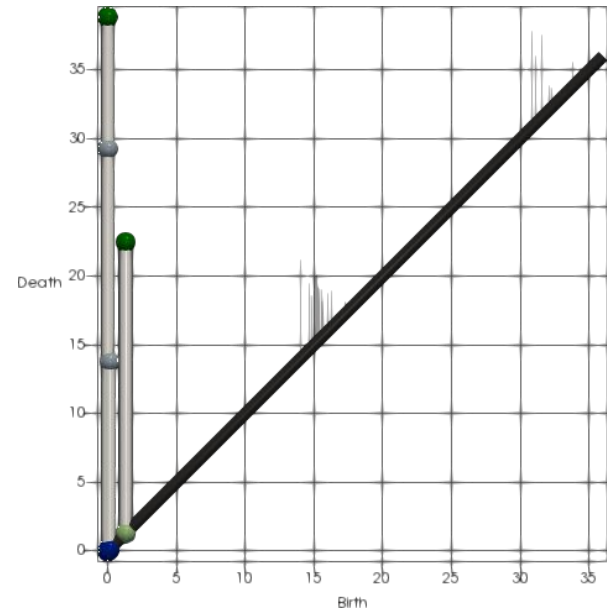
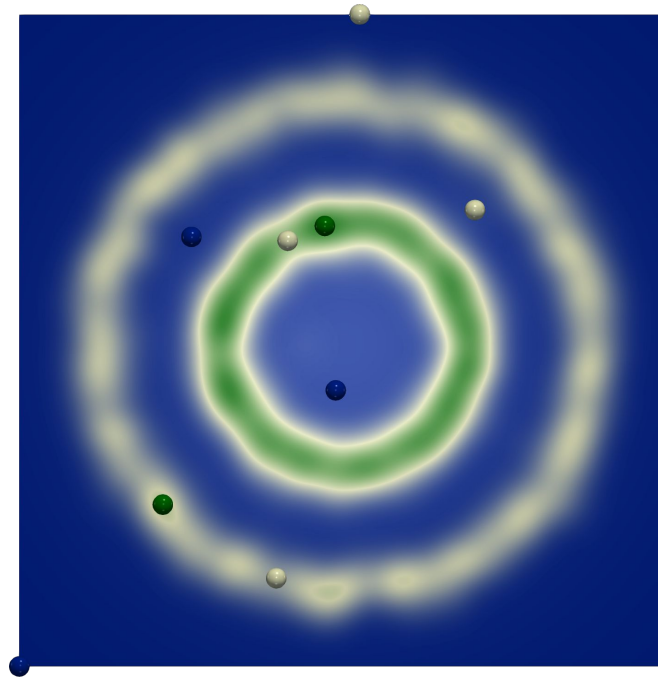
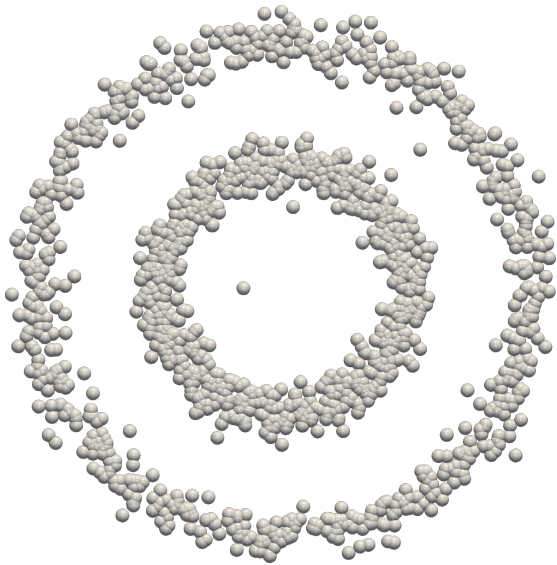
# What about point cloud data?



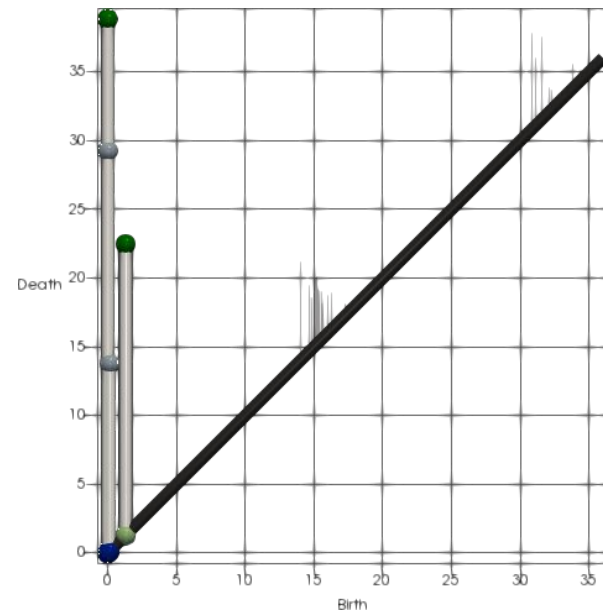
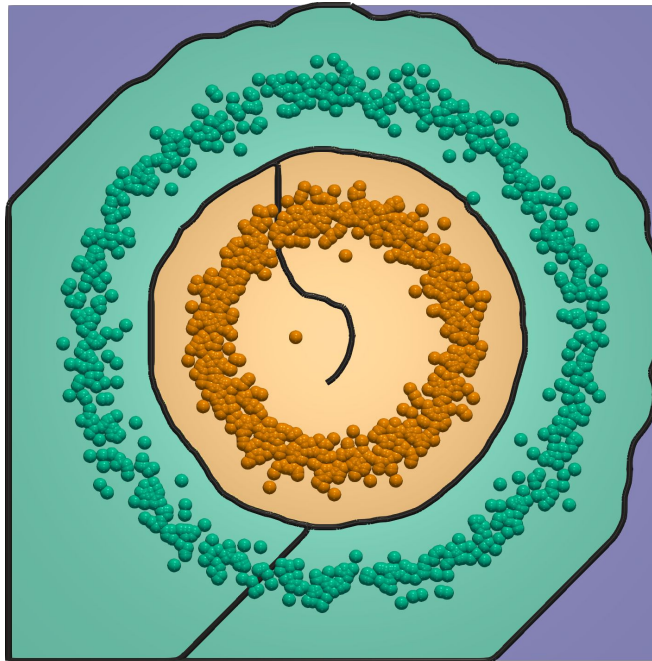
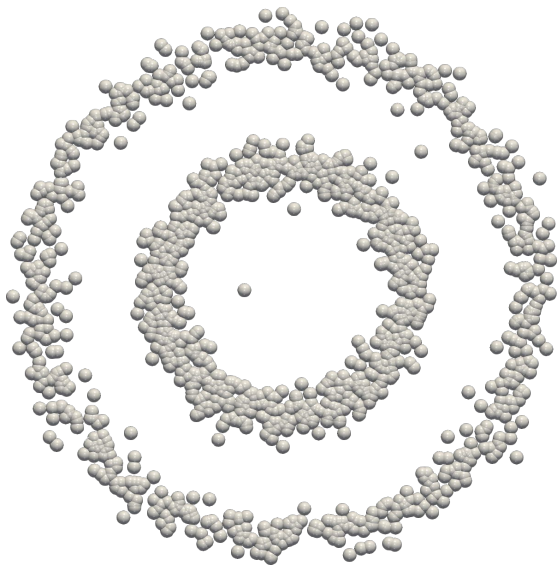
# What about point cloud data?



# What about point cloud data?



# What about point cloud data?



Pipeline Browser

- builtin:
  - clustering0.csv
  - TableToPoints1
    - GaussianResampling1
    - Slice1
      - TTKPersistenceDiagram
      - Threshold1
      - PersistenceThreshold0
      - TTKTopologicalSimplification1
      - TTKTopologicalSimplification1
      - TTKSphereFromPoint2
      - TTKDataSetInterpolator1
      - TTKTopologicalSimplification1
      - TTKMorseSmaleComplex1
        - Critical Points
        - 1-Separatrices
        - Threshold2
        - TTKGeometrySmoother1
        - ExtractSurface1
        - Tube1
        - 2-Separatrices
        - Segmentation

Output Message... Information... Properties

Apply Reset Delete

Properties (TTKMorseSma)

Input options

Scalar Field: SplatterValues

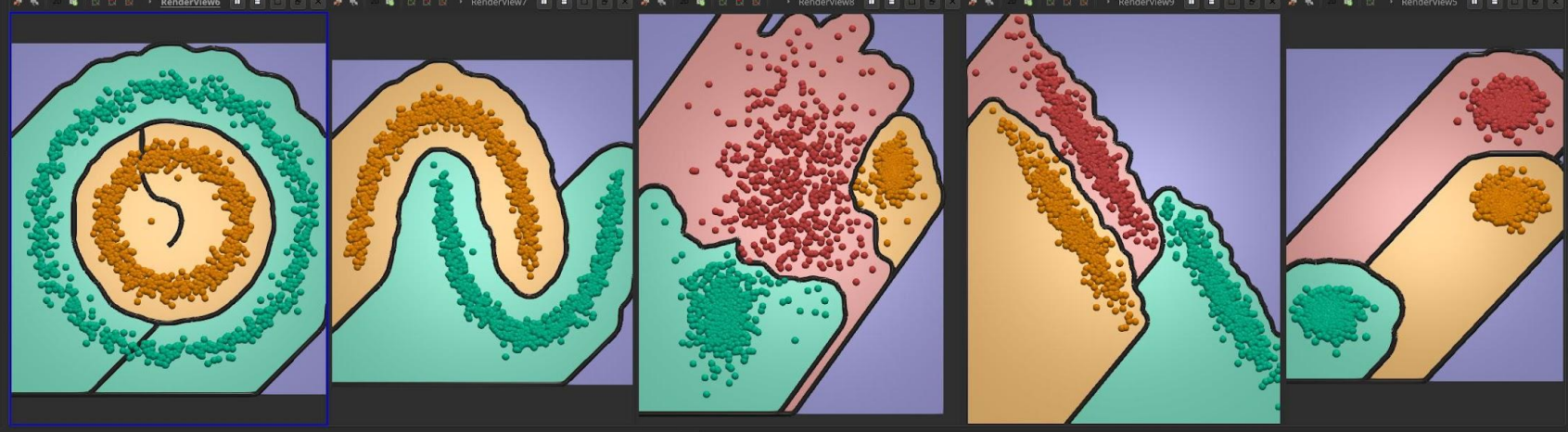
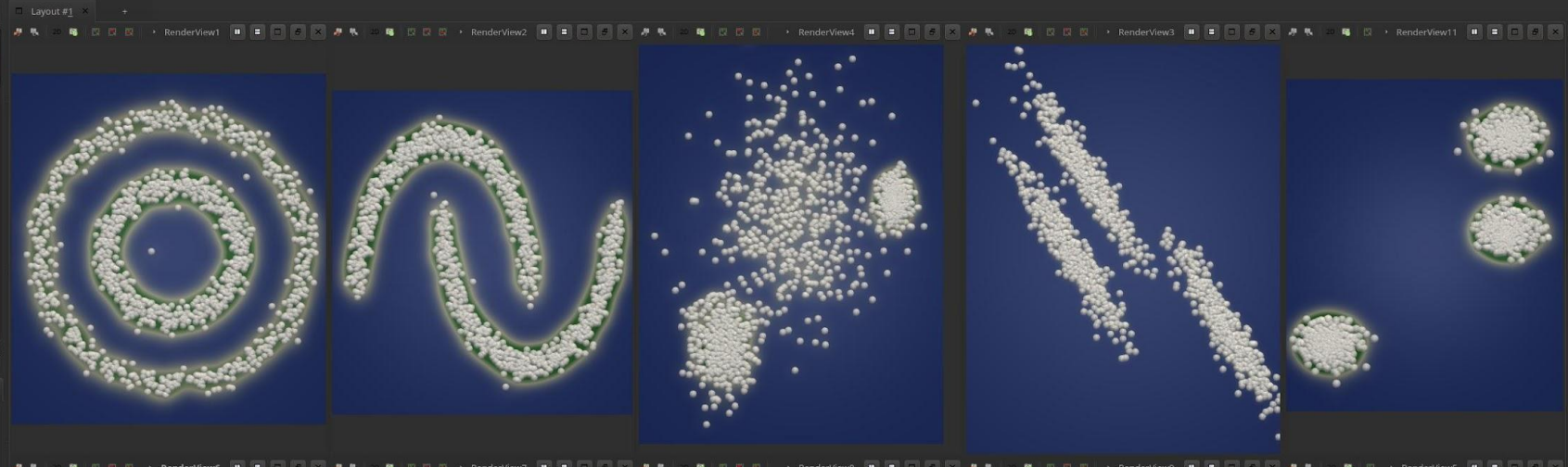
Force Input Offset Field

Output options

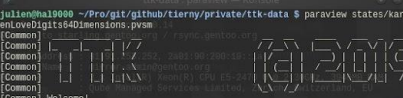
- PL-compliant extrema
- PL-compliant saddles
- Critical Points
- Ascending 1-Separatrices
- Descending 1-Separatrices
- Saddle Connectors
- Ascending 2-Separatrices
- Descending 2-Separatrices
- Ascending Segmentation
- Descending Segmentation
- Morse-Smale Complex Segmentation
- Return Saddle Connectors

Testing

- Prioritize Speed Over Memory
- Use All Cores





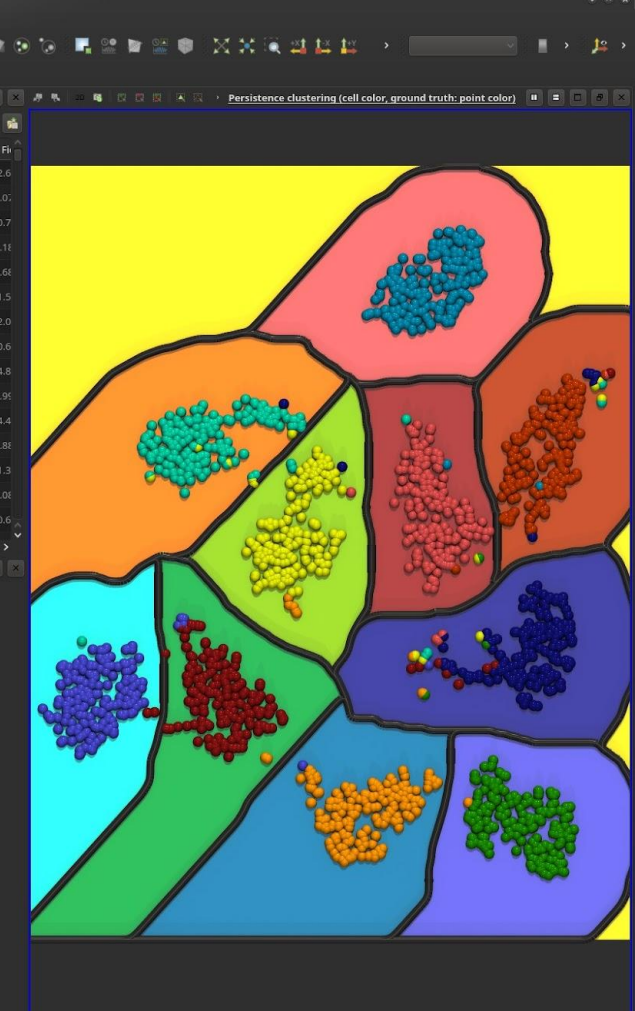
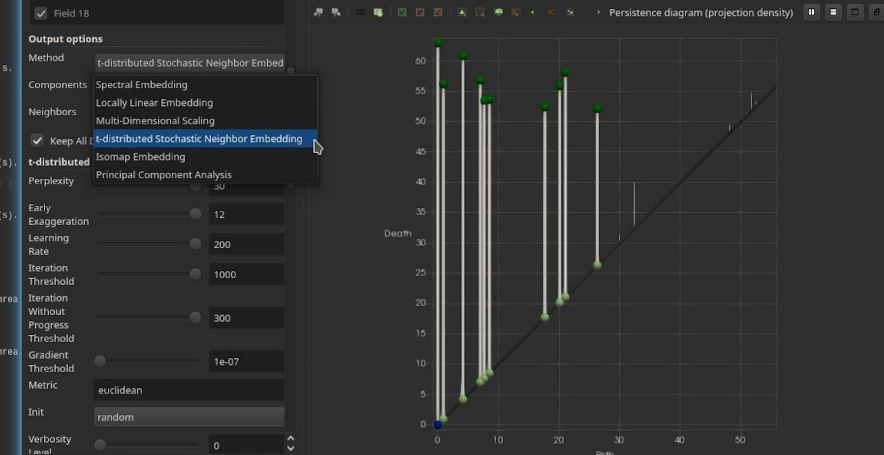


```
DimensionalReduction Loading Python script from: /usr/local/share/scripts/ttk
DimensionalReduction Python: numpy module found.
DimensionalReduction Python: scipy module found.
DimensionalReduction Python: sklearn module found.
DimensionalReduction C-SME computed in 41.1284 s.
[TTKDimensionalReduction] Memory usage: 83.3019 MB.
[ttkPersistenceDiagram] Starting computation on field SpatterValues...
[OneSkeleton] Edge stars built in 0.0697969 s. (195585 edges, 1 thread(s)).
[ZeroSkeleton] One-skeleton built in 0.0232246 s. (24 thread(s)).
[OneSkeleton] Vertex stars built in 0.08990412 s. (1 thread(s)).
[OneSkeleton] Edge stars built in 0.0382842 s. (195585 edges, 24 thread(s)).
[FM] number of threads: 24
[FM] debug lvl: 0
[FM] tree type: dots + Split
[FM] -----
[FM] alloc in mem: 0.00483645
[FM] init in mem: 0.0481091
[FM] sort step in mem: 0.0472759
[FM] leafSearch in mem: 0.00424095
[FM] leafSearch JT in mem: 0.00393195
[FM] trunk JT in mem: 0.008027074
[FM] leafSearch ST in mem: 0.0132246
[FM] trunk ST in mem: 0.008953702
[FM] merge cress in mem: 0.026278
[FM] build tree in mem: 0.0020398
[FM] Total in mem: 0.085798
[ttkPersistenceDiagram] Memory usage: 422.472 MB.
[OneSkeleton] Edge stars built in 0.069829297 s. (195585 edges, 1 thread(s)).
[ZeroSkeleton] One-skeleton built in 0.028211 s. (24 thread(s)).
[TopologicalSimplification] Scalar field simplified in 0.0410411 s. (24 threads (3) + 1 file).
[ttkTopologicalSimplification] Memory usage: 0 MB.
[ZeroSkeleton] Vertex stars built in 0.08873208 s. (1 thread(s)).
[OneSkeleton] Edge stars built in 0.029270 s. (195585 edges, 24 thread(s)).
[ZeroSkeleton] Vertex edges built in 0.0151331 s. (1 thread(s)).
[ThreeSkeleton] Cell edges built in 0.0122329 s. (24 thread(s)).
[TwoSkeleton] Cell neighbors (120656 cells) computed in 0.016376 s. (24 thread(s)).
[ttkMoreSsealComplex] Launching computation on field SpatterValues...
[DiscreteGradient] Data-set: 65536 v. 495585 e., 120656 c.
[DiscreteGradient] Data-set (65536 points) post-processed in 0.00419998 s. (24 thread(s)).
[MoreSsealComplexZD] Discrete gradient overall computed in 0.0185299 s.
[ScalarFieldCriticalPoints] 1 minima.
[ScalarFieldCriticalPoints] 0 saddles(s).
[ScalarFieldCriticalPoints] 0 multi-saddle(s).
[ScalarFieldCriticalPoints] 10 maxima.
[DiscreteGradient] Data-set (65536 vertices) processed in 0.0124412 s. (24 thread(s)).
[DiscreteGradient] 1 0-cells and 0 interior PL.
[DiscreteGradient] 112 1-cells(s) and 9 interior PL.
[DiscreteGradient] 112 2-cells(s) and 0 interior PL.
[DiscreteGradient] Initialization step: 0.008854987 s.
[DiscreteGradient] Ordering of the vpaths: 0.308294e-06 s.
[DiscreteGradient] Processing of the vpaths: 0.00272836 s.
[DiscreteGradient] Gradient reversal step: 2.81334e-06 s.
[DiscreteGradient] Saddle-Maxima pairs simplified in 0.019709 s, 24 thread(s).
[DiscreteGradient] Initialization step: 0.00835896 s.
[DiscreteGradient] Ordering of the vpaths: 2.86212e-06 s.
[DiscreteGradient] Processing of the vpaths: 1.92771e-06 s.
[DiscreteGradient] Gradient reversal step: 3.99944e-06 s.
[DiscreteGradient] Saddle-Maxima pairs simplified in 0.0121448 s, 24 thread(s).
[DiscreteGradient] Gradient reversal in 0.026945 s. (24 thread(s)).
[MoreSsealComplexZD] Descending 1-separatrices computed in 0.00441289 s.
[MoreSsealComplexZD] Ascending 1-separatrices computed in 0.00299788 s.
[MoreSsealComplexZD] Segmentation computed in 0.0463502 s.
[DiscreteGradient] 4 0-cells(s).
[DiscreteGradient] 10 1-cells(s).
[DiscreteGradient] 10 2-cells(s).
[DiscreteGradient] Data-set (65536 points) processed in 0.122486 s. (24 thread(s)).
[ttkMoreSsealComplex] Memory usage: 13.9414 MB.
[OneSkeleton] Edge-list built in 0.0099488 s. (14356 edges, 1 thread(s)).
[ZeroSkeleton] One-skeleton built in 0.00293389 s. (24 thread(s)).
[ScalarFieldSmoothing] Data-set (14370 points) smoothed in 0.043861 s. (24 thread(s)).
[ttkCountrySmoothing] Memory usage: 0.476703 MB.
[ttkIdentifierRandomizer] Shuffling vertex field AscendingManifold...
[ttkIdentifierRandomizer] Memory usage: 0 MB.
```

Input high dimensional point cloud

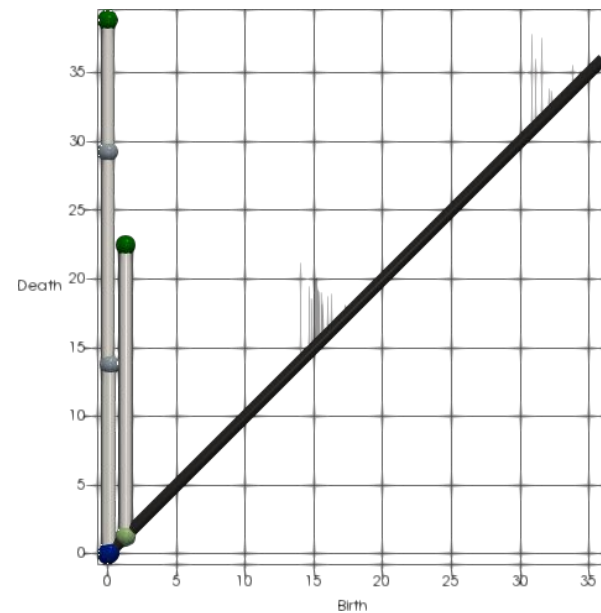
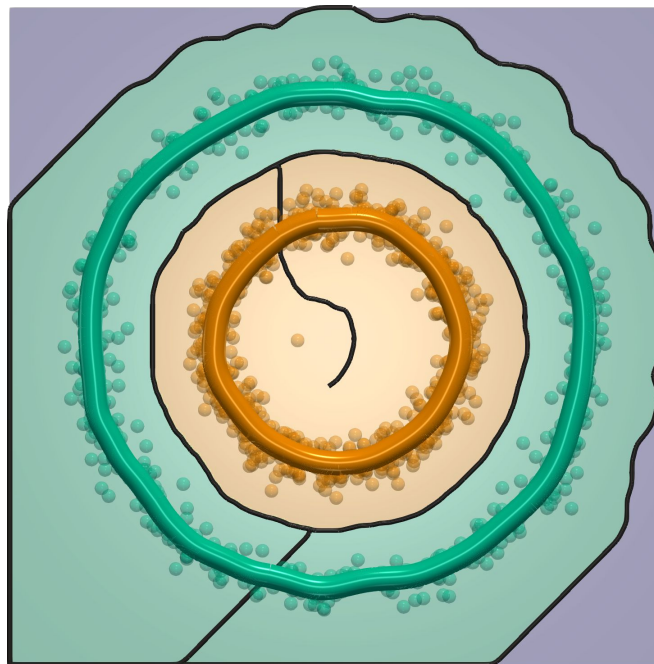
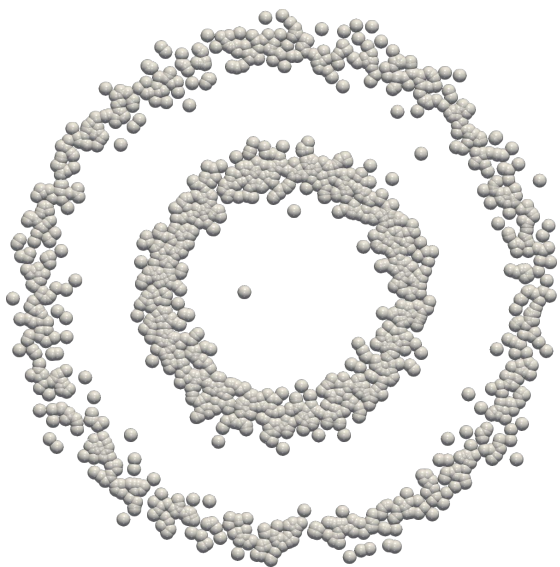
Showing karhunenLoveDigits64Dimensions.csv Attribute: Row Data Precision: 6

Row.ID	Field 0	Field 1	Field 10	Field 11	Field 12	Field 13	Field 14	Field 15	Field 16	Field 17	
0	0	0	-10.297	3.50308	-0.0640736	0.983302	2.00134	1.47868	-0.402921	1.60977	-2.6
1	1	0	-5.03601	1.34607	-1.88801	-1.42551	3.54611	-2.728	0.145671	-0.44373	1.0
2	2	0	-9.63916	4.1534	-4.73283	2.37187	-0.791784	2.60947	-2.20391	-0.43381	-0.7
3	3	0	-6.63037	1.15331	-0.525791	1.30655	3.48947	3.98759	-1.35633	-0.857526	1.18
4	4	0	-10.6645	-0.949622	-1.4932	-1.32539	1.72949	-0.638025	-1.78516	1.26611	0.68
5	5	0	3.43762	-2.78393	4.71745	6.02196	8.26852	1.53817	0.339721	4.25198	-1.5
6	6	0	-13.9869	-1.80014	-0.0350819	0.973274	1.77047	2.73174	-0.624779	1.52599	-2.0
7	7	0	-10.9515	1.78865	-0.631468	-2.37157	1.51382	-0.0580818	-1.35271	0.575439	-0.6
8	8	0	-13.1412	0.553618	-0.202969	-0.294338	2.40556	-0.941013	-0.344848	3.44002	-4.8
9	9	0	-10.3349	0.950181	-3.4854	3.27946	1.77234	2.80083	-2.15157	-2.38731	0.95
10	10	0	-9.98093	-4.04965	-3.03024	0.95547	-0.970807	1.54834	-4.19292	-1.72854	3.8
11	11	0	-10.9697	0.530705	-0.03798	2.0684	-0.968669	0.949393	-2.75193	-3.00611	-3.4
12	12	0	-5.54266	-6.36161	-0.816063	1.29036	4.25531	-0.061144	2.04598	1.61553	-1.3
13	13	0	-8.92591	-0.889281	0.0698092	0.515757	1.59764	-0.0228414	-1.55912	1.78995	0.08
14	14	0	-11.1141	-2.65117	-1.00536	1.82424	0.386598	2.7214	-0.208347	1.51755	-0.6



Live demo

# What about point cloud data?



```

[Common] Welcome!
[Common] Welcome!
[DimensionReduction] Loading Python script from: /usr/local/share/scripts/ttk
[DimensionReduction] Python: numpy module found.
[DimensionReduction] Python: scipy module found.
[DimensionReduction] Python: sklearn module found.
[ttkDimensionReduction] Memory usage: 133.514 MB.
[ttkDimensionReduction] Starting computation on field 'SplatterValues'...
[ttkPersistenceDiagram] Edge-list built in 0.00146704 s. (24193 edges, 1 thread(s)).
[ZeroSkeleton] One-skeleton built in 0.00207064 s. (24 threads(s)).
[ZeroSkeleton] Vertex stars built in 0.00077060 s. (1 thread(s)).
[OneSkeleton] Edge stars built in 0.00670409 s. (24193 edges, 24 thread(s))
[FTM] -----
[FTM] number of threads : 24
[FTM] * debug lvl : 3
[FTM] * tree type : Join + Split
[FTM] -----
[FTM] alloc in 0.000795941
[FTM] init in 0.004341
[FTM] sort step in 0.018899
[FTM] leafSearch JT in 0.000741065
[FTM] leafGrowth JT in 0.000730859
[FTM] trunk JT in 0.000933085
[FTM] leafSearch ST in 0.000707965
[FTM] leafGrowth ST in 0.000559992
[FTM] trunk ST in 0.000208139
[FTM] merge trees in 0.0052184
[FTM] build tree in 0.00507
[FTM] Total in 0.07338
[ttkPersistenceDiagram] Memory usage: 360.087 MB.
[OneSkeleton] Edge-list built in 0.0033300 s. (24193 edges, 1 thread(s)).
[ZeroSkeleton] One-skeleton built in 0.00529964 s. (24 thread(s)).
[TopologicalSimplification] Scalar field simplified in 0.01265 s. (24 threads(s),
  1 ite).
[ttkTopologicalSimplification] Memory usage: 0 MB.
[ZeroSkeleton] Vertex stars built in 0.00109951 s. (1 thread(s)).
[OneSkeleton] Edge stars built in 0.0070600 s. (24193 edges, 24 thread(s)).
[ZeroSkeleton] Vertex edges built in 0.00228 s. (1 thread(s)).
[ThreeSkeleton] Cell edges built in 0.00321067 s. (24 thread(s)).
[TwoSkeleton] Cell neighbors (16062 cells) computed in 0.0033606 s. (24 thread(s)).
[ttkMorseSmaleComplex] Launching computation on field 'SplatterValues'...
[DiscreteGradient] Data-set: 8192 v. 24193 e. 16062 f.
[DiscreteGradient] Processed in 0.00859996 s. (24 thread(s)).
[DiscreteGradient] Data-set (8192 points) post-processed in 0.00421308 s. (24 thread(s)).
[MorseSmaleComplex2D] Discrete gradient overall computed in 0.012794 s.
[ScalarFieldCriticalPoints] 1 minima
[ScalarFieldCriticalPoints] 7 saddle(s).
[ScalarFieldCriticalPoints] 0 multi-saddle(s).
[ScalarFieldCriticalPoints] 7 maxima
[ScalarFieldCriticalPoints] Data-set (8192 vertices) processed in 0.00769401 s. (24 thread(s)).
[DiscreteGradient] 1 0-cell(s) and 0 interior Pt.
[DiscreteGradient] 180 2-cell(s) and 6 interior Pt.
[DiscreteGradient] 180 2-cell(s) and 0 interior Pt.
[DiscreteGradient] Initialization step : 0.00688801 s.
[DiscreteGradient] Ordering of the vpaths : 0.00911907 s.
[DiscreteGradient] Processing of the vpaths : 0.008339631 s.
[DiscreteGradient] Gradient reversal step : 2.98923e-05 s.
[DiscreteGradient] Saddle-Maxima pairs simplified in 0.0187871 s. (24 thread(s)).
[DiscreteGradient] Initialization step : 0.0082829 s.
[DiscreteGradient] Ordering of the vpaths : 1.9079e-06 s.
[DiscreteGradient] Processing of the vpaths : 7.45250e-06 s.
[DiscreteGradient] Gradient reversal step : 3.09944e-06 s.
[DiscreteGradient] Saddle-Maxima pairs simplified in 0.0184855 s. (24 thread(s)).
[DiscreteGradient] Gradient reversal in 0.0048325 s. (24 thread(s)).
[MorseSmaleComplex2D] Descending 1-separatrices computed in 0.00420904 s.
[MorseSmaleComplex2D] Ascending 1-separatrices computed in 0.00365591 s.
[MorseSmaleComplex2D] Separatrices computed in 0.0153407 s.
[DiscreteGradient] 1 0-cell(s).
[DiscreteGradient] 7 1-cell(s).
[DiscreteGradient] 7 2-cell(s).
[MorseSmaleComplex2D] Data-set (8192 points) processed in 0.0714762 s. (24 thread(s)).
[ttkMorseSmaleComplex] Memory usage: 1.84668 MB.
[ttkSphereFromPoint] Spheres computed in 0.00451803 s.
[ttkSphereFromPoint] Memory usage: 0.45091 MB.
[OneSkeleton] Edge-list built in 0.000143051 s. (820 edges, 1 thread(s)).
[ZeroSkeleton] One-skeleton built in 0.000349998 s. (24 thread(s)).
[ScalarFieldSmoother] Data-set (832 points) smoothed in 0.408252 s. (24 thread(s)).
[ttkGeometrySmoother] Memory usage: 0 MB.
[ttkSphereFromPoint] Spheres computed in 0.136478 s.
[ttkSphereFromPoint] Memory usage: 22.3880 MB.

```

Pipeline Browser

- builtin:
  - pointCloud.csv
  - TTKDimensionReduction1
    - TableToPoints2
    - GaussianResampling2
    - Slice1
    - TTKPersistenceDiagram
    - Threshold1
    - PersistentThreshold
    - TTKTopologicalSimplification
    - TTKSphereFromPoint2
  - TableToPoints1
  - GaussianResampling1
  - Outline1
  - TTKTopologicalSimplification
  - TTKMorseSmaleComplex1
  - CriticalPoints
  - TTKSphereFromPoint1
  - 1-Separatrices
  - Threshold2
  - Threshold3

Output Message... Informat... Properti...

Properties

Apply Reset Delete ?

Search... (use Esc to clear text)

Properties (TTKDimensionReduction1)

Input options

- Input Columns
- Elevation
- Points:0
- Points:1
- Points:2

Output options

Method: Multi-Dimensional Scaling

Components

- Spectral Embedding
- Locally Linear Embedding
- Multi-Dimensional Scaling
- Isomap Embedding
- Principal Component Analysis

Neighbors

- Keep All
- t-distributed Stochastic Neighbor Embedding

Multi-Dimensional Scaling

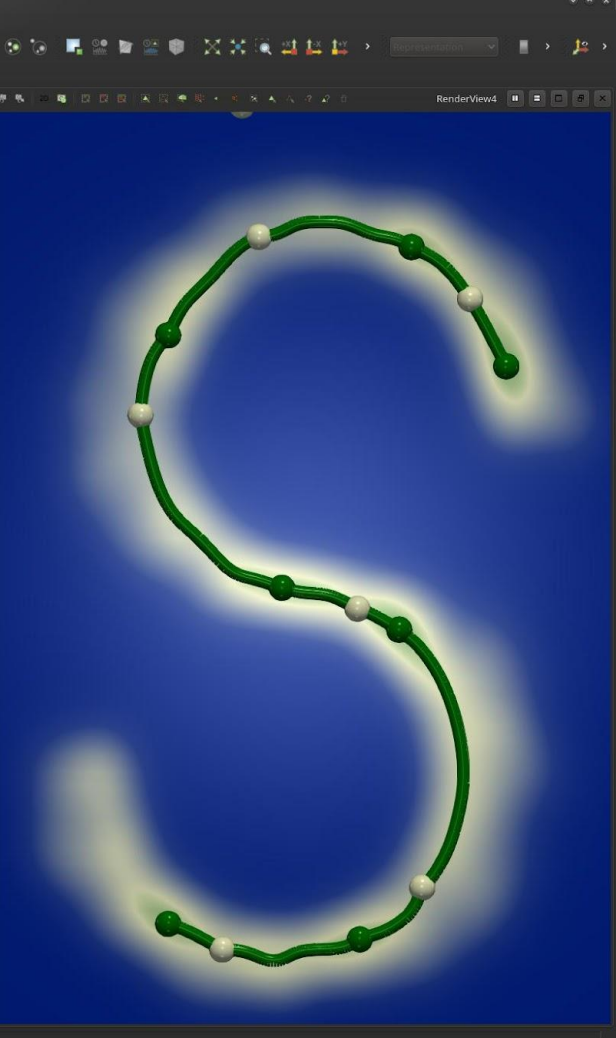
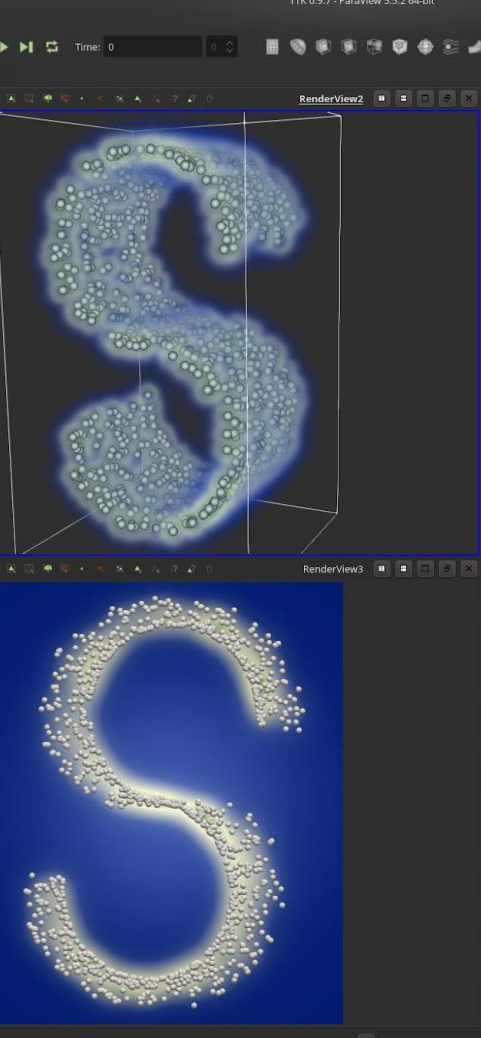
- Metric
- Principal Component Analysis

Number of iterations: 4

Iteration: 300

Threshold

Verbosity



```

[ttkTopologicalSimplification] Memory usage: 0 MB.
[ImplicitTriangulation] The getVertex() requests are accelerated.
[ttkSphereFromPoint] Launching computation on field 'SplatterValues'...
[DiscreteGradient] Data-set: 524288 V., 3588667 e., 6688898 t., 3924378 T.
[DiscreteGradient] Processed in 0.85227 s. (24 thread(s)).
[DiscreteGradient] Data-set (524288 points) post-processed in 0.819482 s. (24 thr
ad(s)).
[DiscreteGradient] Data-set (524288 points) post-processed in 0.809758 s. (24 thr
ad(s)).
[MorseSmaleComplex3D] Discrete gradient overall computed in 0.388869 s.
[ScalarFieldCriticalPoints] 56 minima.
[ScalarFieldCriticalPoints] 1254 1-saddle(s).
[ScalarFieldCriticalPoints] 1926 2-saddle(s).
[ScalarFieldCriticalPoints] 98 multi-saddle(s).
[ScalarFieldCriticalPoints] 429 maxima.
[ScalarFieldCriticalPoints] Data-set (524288 vertices) processed in 0.65695 s. (24
thread(s)).
[DiscreteGradient] 56 0-cell(s) and 55 interior PL.
[DiscreteGradient] 1854 1-cell(s) and 1245 interior PL.
[DiscreteGradient] 3556 2-cell(s) and 1974 interior PL.
[DiscreteGradient] 1766 3-cell(s) and 629 interior PL.
[DiscreteGradient] Initialization step : 0.0282591 s.
[DiscreteGradient] Ordering of the vpaths : 0.069731945 s.
[DiscreteGradient] Processing of the vpaths : 0.8078969 s.
[DiscreteGradient] Gradient reversal step : 5.19753e-05 s.
[DiscreteGradient] Saddle-Maximum pairs simplified in 0.0724692 s, 24 thread(s).
[DiscreteGradient] Initialization step : 0.258536 s.
[DiscreteGradient] Ordering of the vpaths : 0.080259969 s.
[DiscreteGradient] Processing of the vpaths : 0.8568815 s, 24 thread(s).
[DiscreteGradient] Initialization step : 0.039736 s.
[DiscreteGradient] Ordering of the vpaths : 2.80272e-05 s.
[DiscreteGradient] Processing of the vpaths : 0.92031 s.
[DiscreteGradient] Saddle-Saddle pairs simplified in 0.0978661 s, 24 thread(s).
[DiscreteGradient] Initialization step : 0.0624093 s.
[DiscreteGradient] Ordering of the vpaths : 0.089952086 s.
[DiscreteGradient] Processing of the vpaths : 0.80838112 s.
[DiscreteGradient] Gradient reversal step : 0.08020508 s.
[DiscreteGradient] Saddle-Maximum pairs simplified in 0.0755041 s, 24 thread(s).
[DiscreteGradient] Initialization step : 0.257937 s.
[DiscreteGradient] Ordering of the vpaths : 2.29296e-05 s.
[DiscreteGradient] Processing of the vpaths : 0.872561 s.
[DiscreteGradient] Saddle-Saddle pairs simplified in 0.389896 s, 24 thread(s).
[DiscreteGradient] Initialization step : 0.0597069 s.
[DiscreteGradient] Ordering of the vpaths : 1.3113e-05 s.
[DiscreteGradient] Processing of the vpaths : 0.826727 s.
[DiscreteGradient] Saddle-Saddle pairs simplified in 0.113457 s, 24 thread(s).
[FTM] -----
[FTM] number of threads : 24
[FTM] debug lvl : 0
[FTM] tree type : Contour
[FTM] -----
[FTM] alloc in 0.030821
[FTM] init in 0.0758661
[FTM] sort step in 0.022268
[FTM] leafSearch JT in 0.021368
[FTM] leafGrowth JT in 0.411031
[FTM] trunk JT in 0.0020761
[FTM] leafSearch ST in 0.0202370
[FTM] leafGrowth ST in 0.00806093
[FTM] trunk ST in 0.00227094
[FTM] merge trees in 0.450324
[FTM] combine full in 0.0143611
[FTM] build tree in 0.465250
[FTM] Total in 0.64471
[DiscreteGradient] Initialization step : 0.217854 s.
[DiscreteGradient] Ordering of the vpaths : 0.998084981 s.
[DiscreteGradient] Processing of the vpaths : 1.73724 s.
[DiscreteGradient] Saddle-Saddle pairs simplified in 1.98662 s, 24 thread(s).
[DiscreteGradient] Initialization step : 0.0448911 s.
[DiscreteGradient] Ordering of the vpaths : 0.86844384 s.
[DiscreteGradient] Processing of the vpaths : 0.8243599 s.
[DiscreteGradient] Saddle-Saddle pairs simplified in 0.092867 s, 24 thread(s).
[DiscreteGradient] Gradient reversal in 4.8684 s. (24 thread(s)).
[MorseSmaleComplex3D] Descending 1-separatrices computed in 0.035454 s.
[MorseSmaleComplex3D] Ascending 1-separatrices computed in 0.0279231 s.
[MorseSmaleComplex3D] Saddle connectors computed in 0.79316 s.
[MorseSmaleComplex3D] Ascending 2-separatrices computed in 0.2916 s.
[MorseSmaleComplex3D] Segmentation computed in 0.679845 s.
[DiscreteGradient] 56 0-cell(s).
[DiscreteGradient] 696 1-cell(s).
[DiscreteGradient] 1278 2-cell(s).
[DiscreteGradient] 629 3-cell(s).
[MorseSmaleComplex3D] Data-set (524288 points) processed in 7.10692 s. (24 thread
(s)).
[ttkMorseSmaleComplex] Memory usage: 52.7666 MB.
[ttkSphereFromPoint] Spheres computed in 0.318438 s.
[ttkSphereFromPoint] Memory usage: 45.4062 MB.
[OneSkeleton] Edge-list built in 0.8061099 s. (89953 edges, 1 thread(s)).
[ZeroSkeleton] One-skeleton built in 0.013974 s. (24 thread(s)).
[ScalarFieldSmoother] Data-set (80260 points) smoothed in 0.102752 s. (24 thread
(s)).
[ttkGeometrySmoother] Memory usage: 0 MB.
[OneSkeleton] Edge-list built in 0.017694 s. (220956 edges, 1 thread(s)).
[ZeroSkeleton] One-skeleton built in 0.0328891 s. (24 thread(s)).
[ScalarFieldSmoother] Data-set (73228 points) smoothed in 0.114568 s. (24 thread
(s)).
[ttkGeometrySmoother] Memory usage: 1.12598 MB.

```

File Edit View Sources Filters Tools Catalyst Macros Help

Time: 0

Pipeline Browser

- builtin:
  - pointCloud.csv
  - TableToPoints
  - GaussianResampling1
  - Outline1
  - TTKPersistenceDiagram1
  - Threshold1
  - PersistenceThreshold
  - TTKTopologicalSimplification1
  - TTKTopologicalSimplification1
  - TTKTopologicalSimplification1
  - TTKTopologicalSimplification1
  - TTKMorseSmaleComplex1**
  - Critical Points
  - TTKSphereFromPoint
  - 1-separatrices
  - Threshold2
  - TTKGeometrySmoother1
  - ExtractSurface1
  - Tube1
  - 2-separatrices
  - Tetrahedralize1
  - TTKGeometrySmoother2
  - ExtractSurface2

Output Message... Informa... Properti...

Properties

Apply Reset Delete ?

Search... (use Esc to clear text)

Properties (1)

Input options

Scalar Field SplatterValues

Force Input Offset Field

Output options

- PL-compliant extrema
- PL-compliant saddles
- Critical Points
- Ascending 1-separatrices
- Descending 1-separatrices
- Saddle Connectors
- Ascending 2-separatrices
- Descending 2-separatrices
- Ascending Segmentation
- Descending Segmentation
- Morse-Smale Complex Segmentation
- Return Saddle Connectors

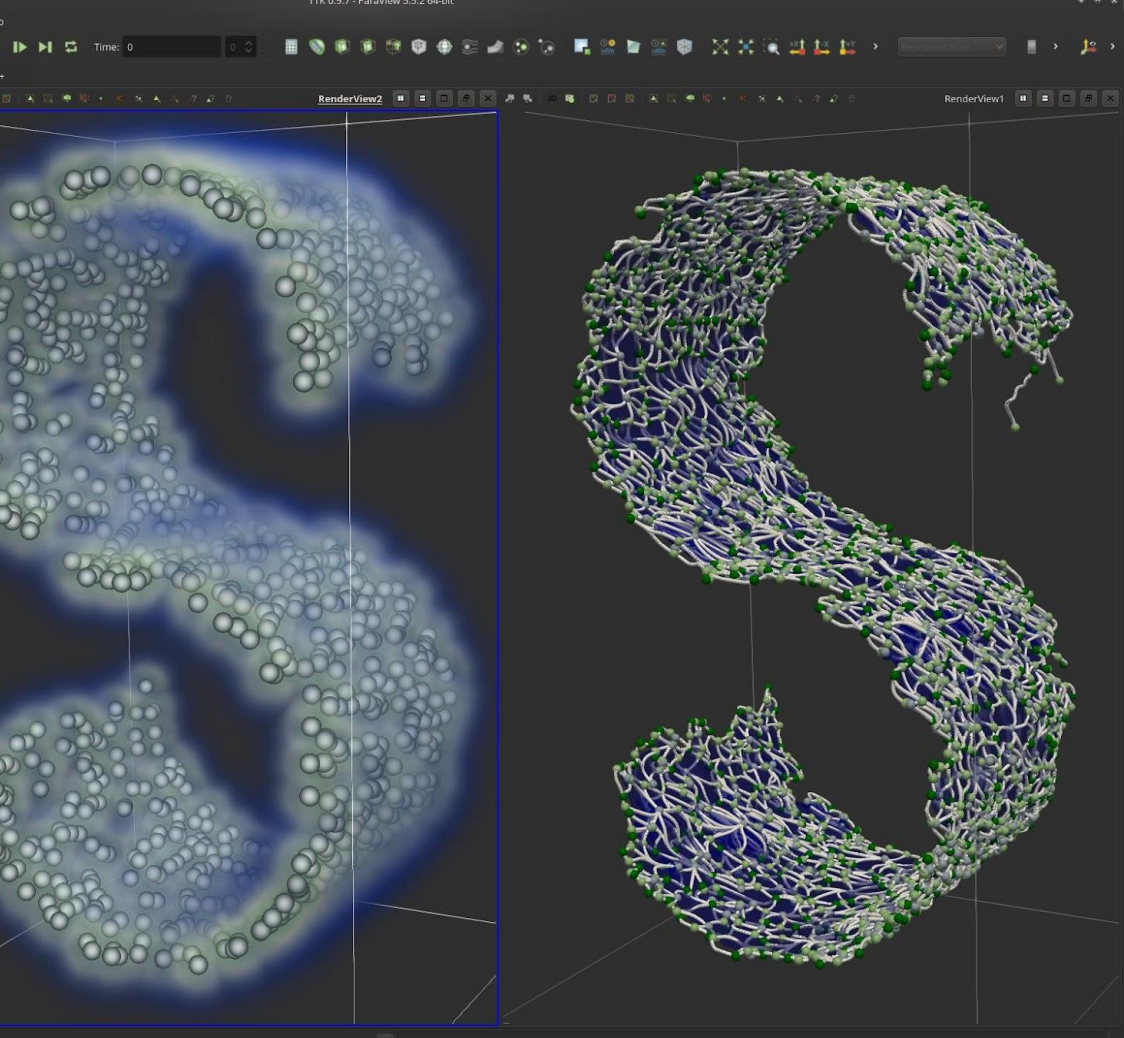
Saddle Connectors

Persistence 0.01

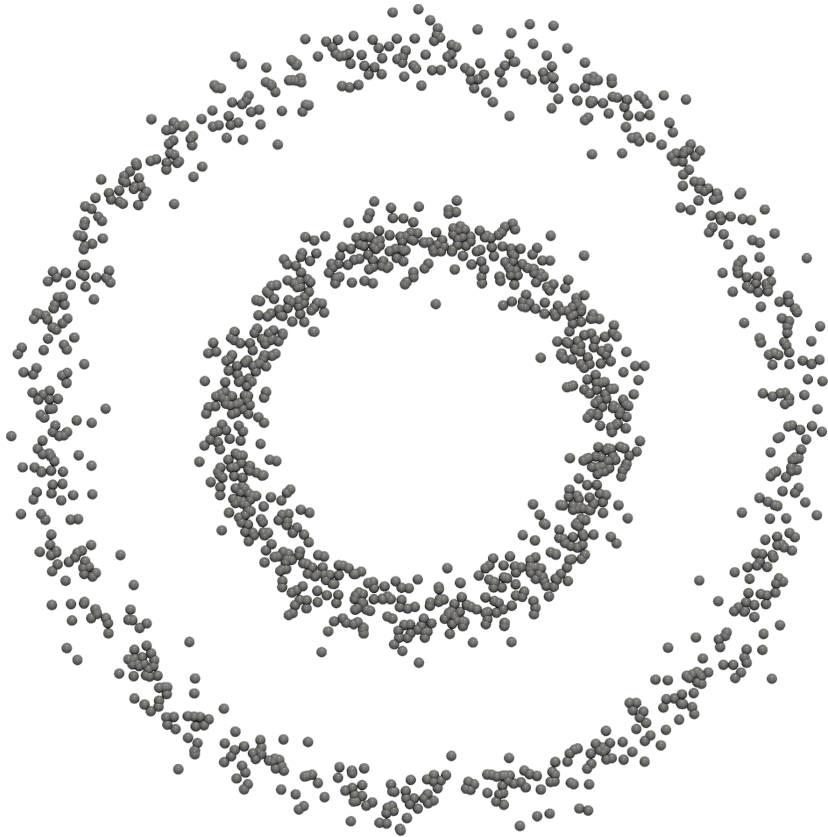
Threshold

Testing

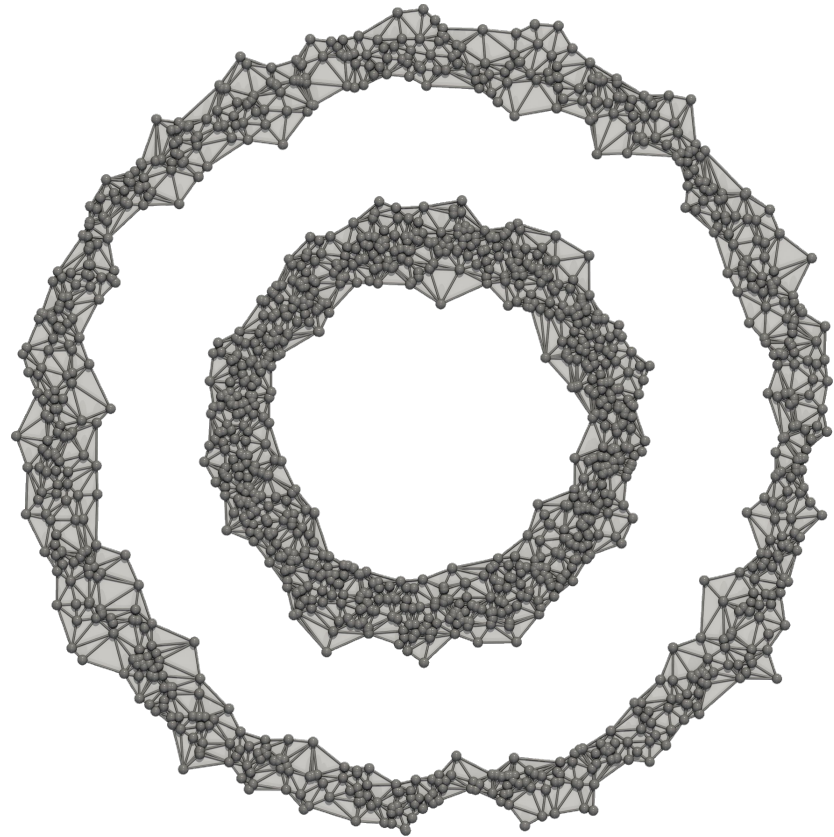
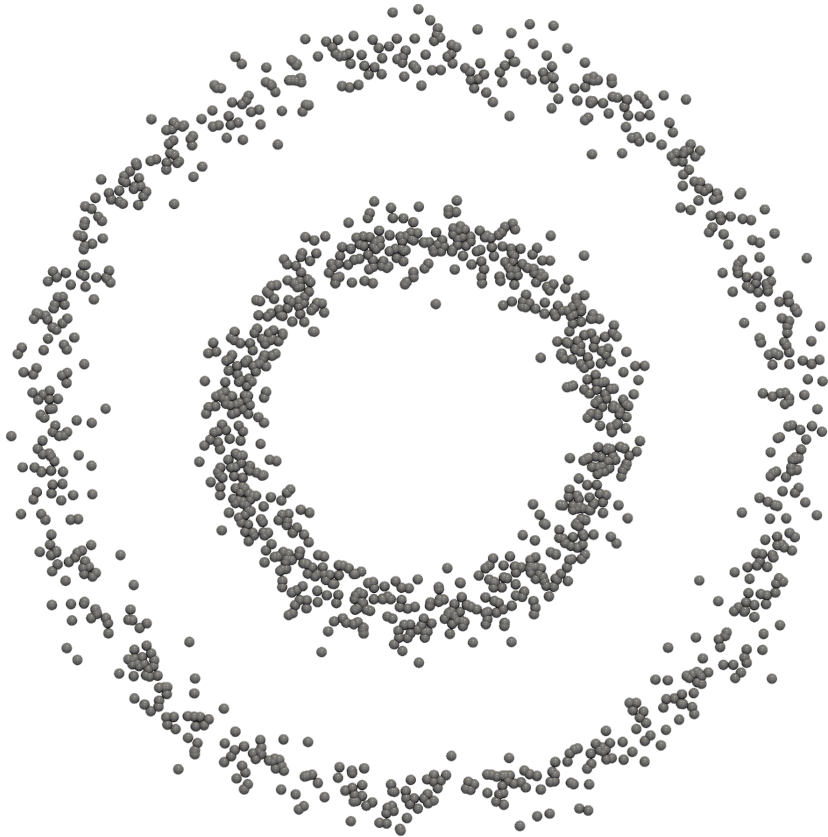
Resurface From One Skeleton



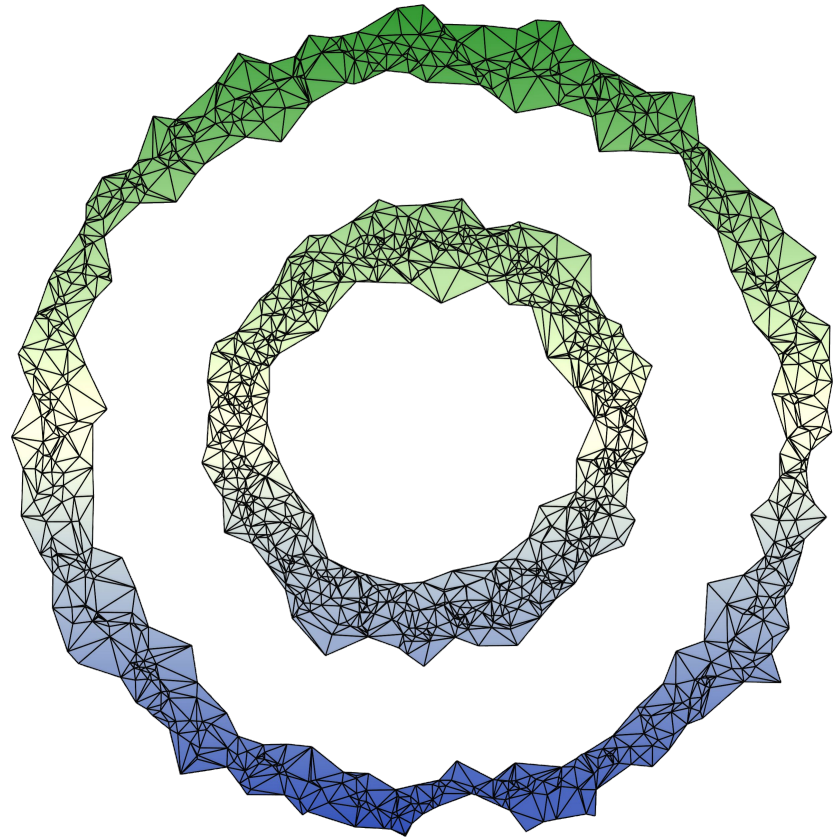
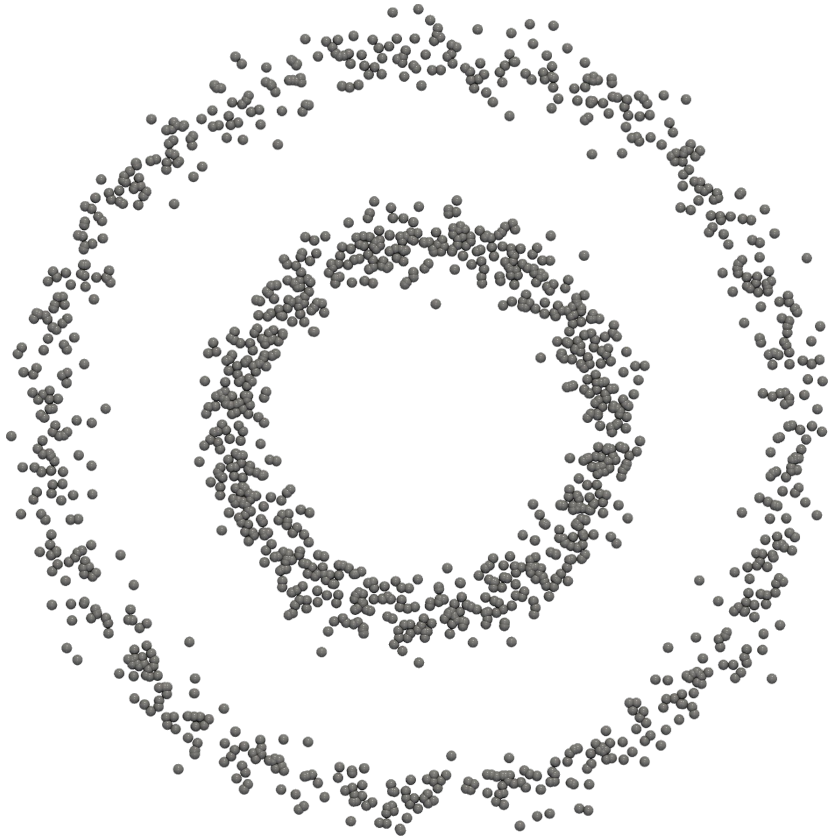
# Mapper



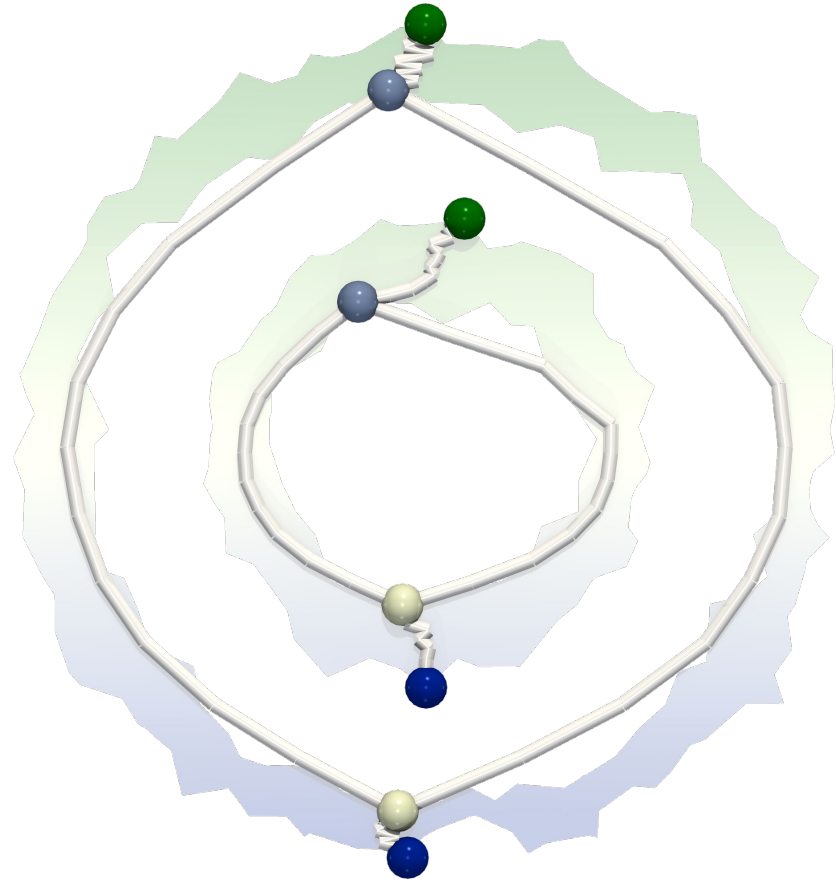
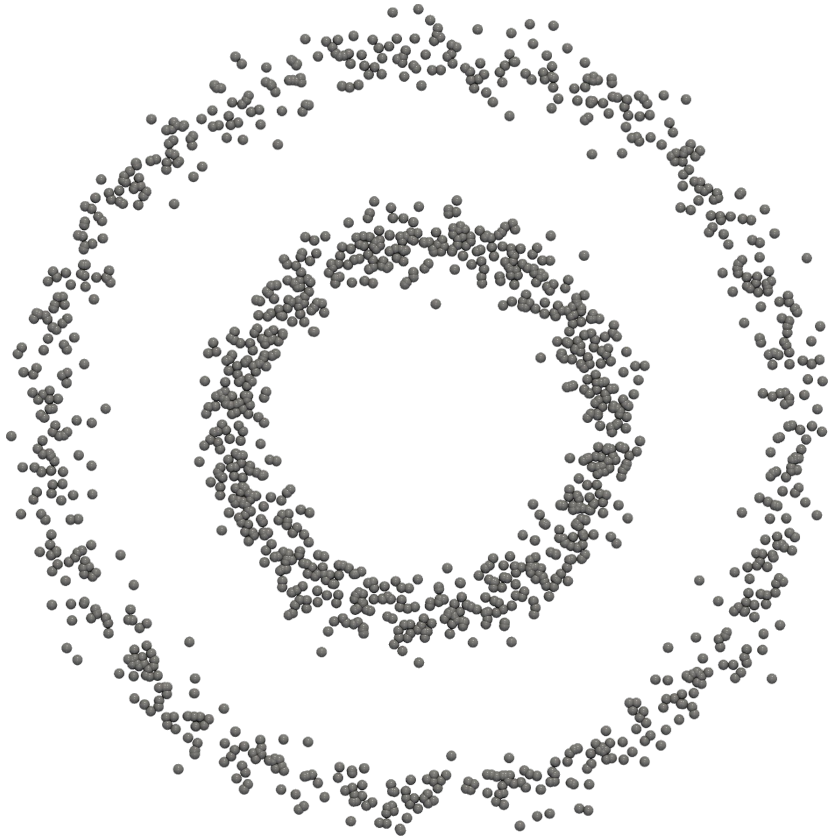
# Mapper



# Mapper

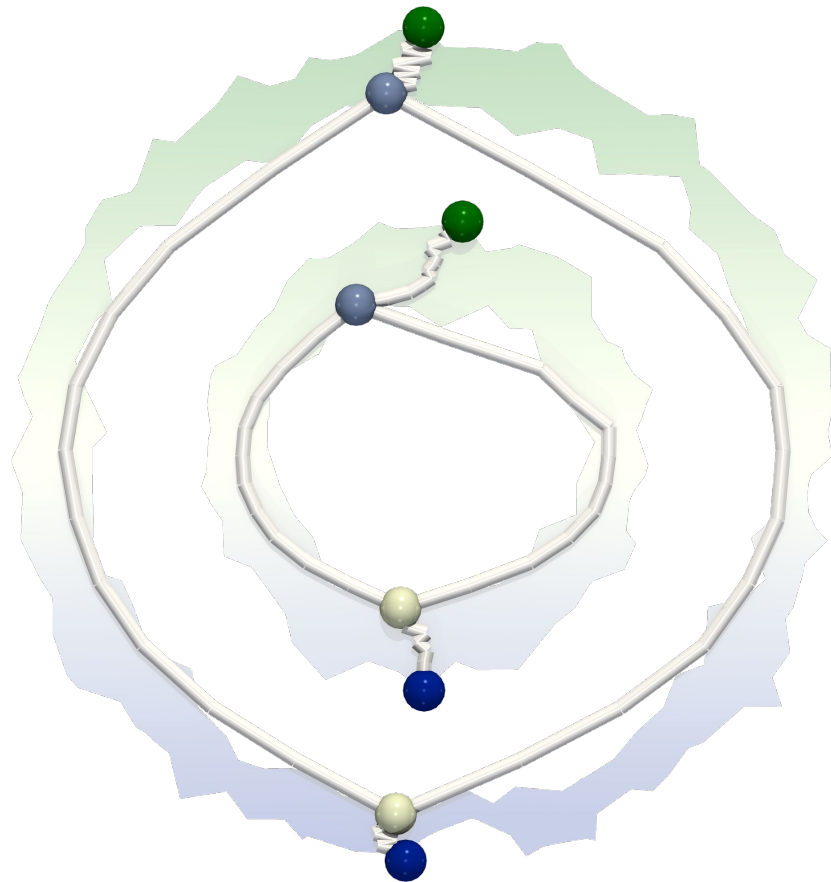
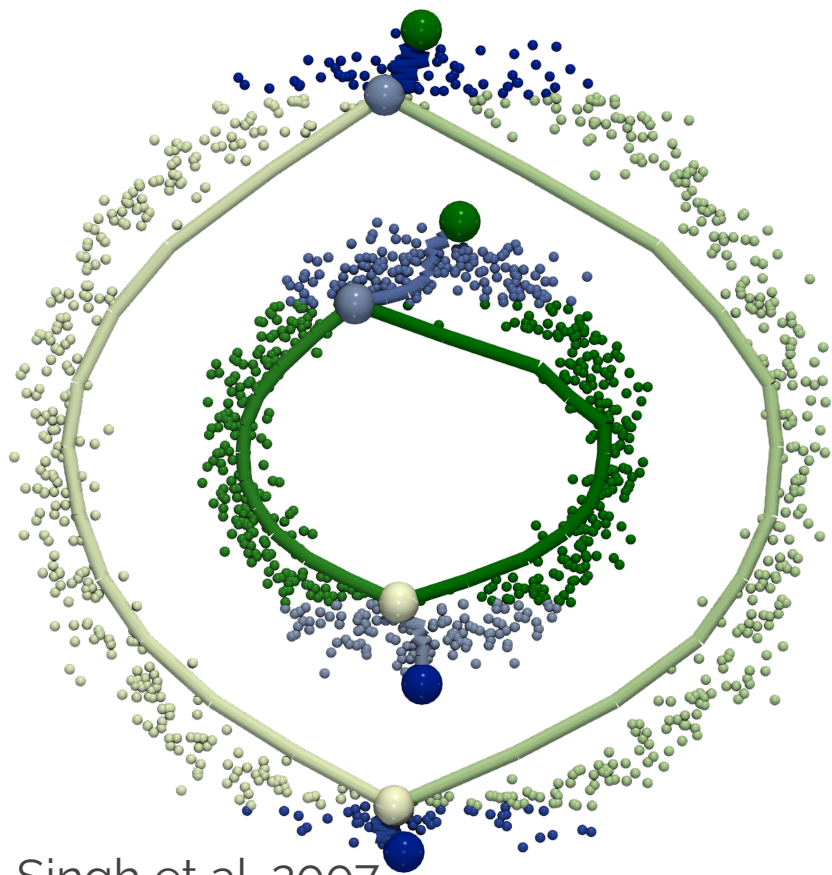


# Mapper





# Mapper



● Singh et al. 2007

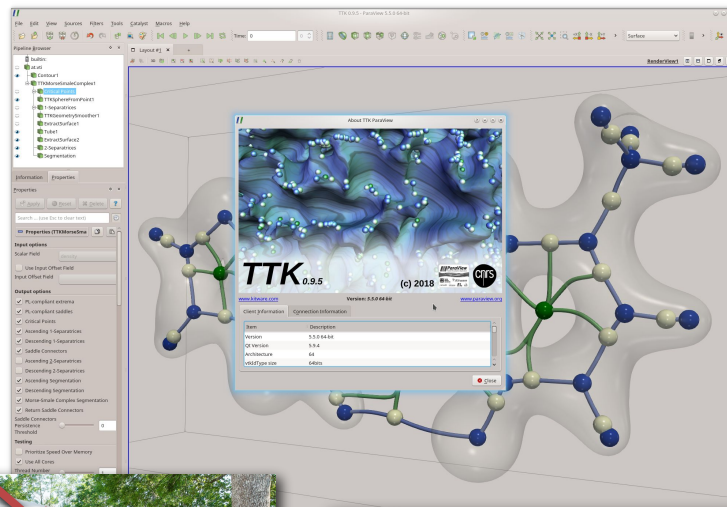
# The Topology ToolKit (TTK)

- **Open-source TDA library**

- ~120k lines in C++, BSD license
- Python bindings, binary packages
  - **Officially integrated in ParaView (5.10)**
- <http://topology-tool-kit.github.io>
- Best paper honorable mention IEEE VIS'17

- **Structuring research receptacle**

- 17 contributing institutions
  - 14 universities, 3 companies
- Mini-symposia:
  - IEEE VIS'18-19-20-21 (2020-1: online)
  - Hackathons



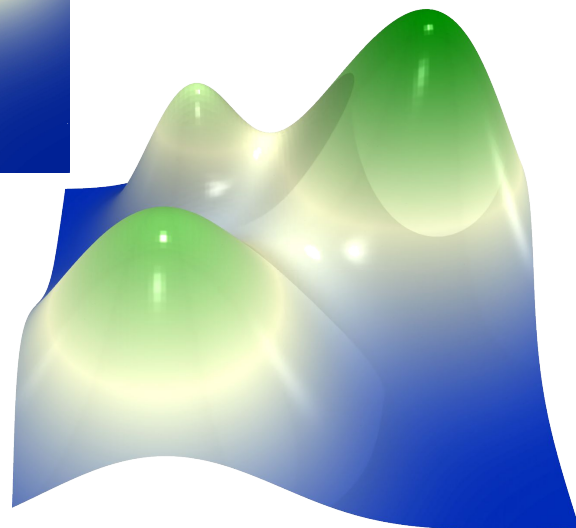
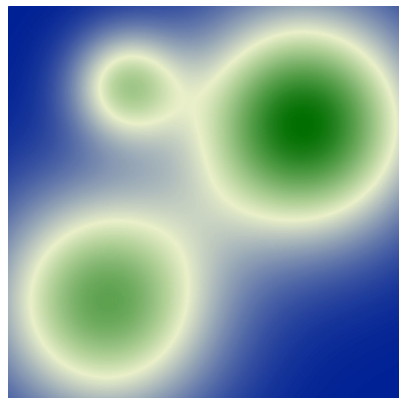
# Why using TTK?

- **What is it good for?**

- Low dimensional data
- Continuous scalar fields
- Science & engineering
  - Astrophysics, biological imaging, quantum chemistry, fluid dynamics, material sciences

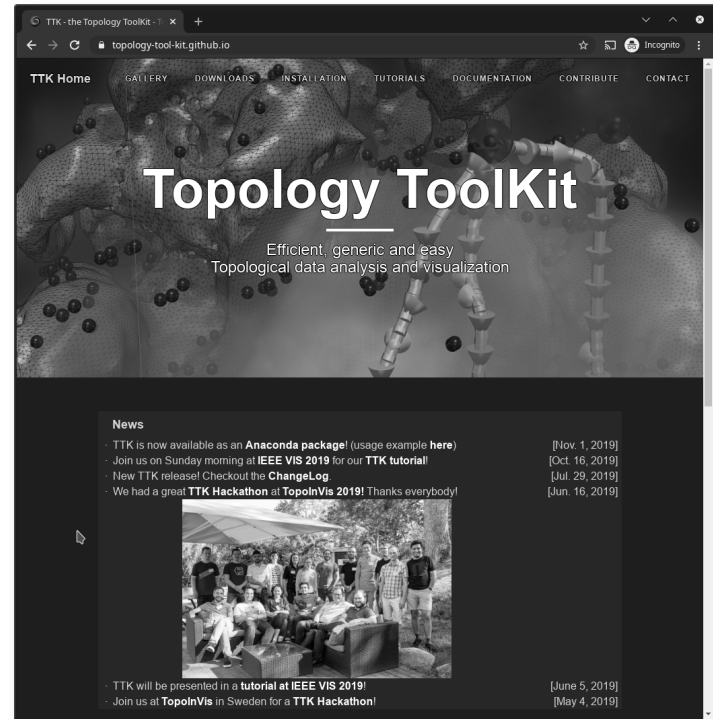
- **What is it not good for? (yet)**

- Vector / tensor data
- High dimensional data



# Online resources

- <https://topology-tool-kit.github.io/>
  - Installation instructions
  - Gallery
  - Data and examples
  - Video tutorials
  - Documentation
  - Exercises
  - Mailing lists
    - [ttk-users@googlegroups.com](mailto:ttk-users@googlegroups.com)



# Installation

- <http://topology-tool-kit.github.io/installation.html>

- **Easy installation**

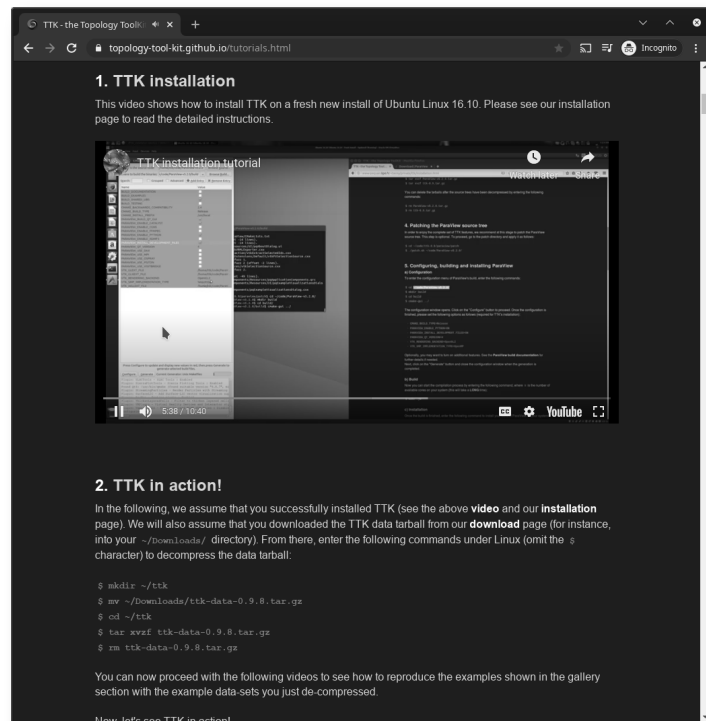
- Officially integrated in ParaView 5.10
  - <http://www.paraview.org/download>
- Binary packages
  - Linux (Ubuntu), MacOS, Windows
- Anaconda package
  - `conda install -c conda-forge topologytoolkit`

- **Virtualization images**

- Virtualbox & docker

- **For the latest features**

- Github repository
  - <https://github.com/topology-tool-kit/ttk>



The screenshot shows a YouTube video player with a dark theme. The video title is "1. TTK installation". The description reads: "This video shows how to install TTK on a fresh new install of Ubuntu Linux 16.10. Please see our installation page to read the detailed instructions." The video content shows a terminal window with the following commands and output:

```
$ mkdir ~/ttk
$ mv ~/Downloads/ttk-data-0.9.8.tar.gz
$ cd ~/ttk
$ tar xvfz ttk-data-0.9.8.tar.gz
$ rm ttk-data-0.9.8.tar.gz
```

Below the video, there is a section titled "2. TTK in action!" with the following text: "In the following, we assume that you successfully installed TTK (see the above video and our installation page). We will also assume that you downloaded the TTK data tarball from our download page (for instance, into your ~/Downloads/ directory). From there, enter the following commands under Linux (omit the \$ character) to decompress the data tarball:"

At the bottom of the page, there is a small text: "Now let's see TTK in action!"

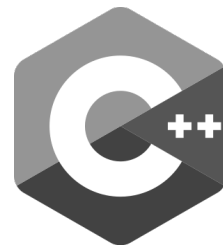
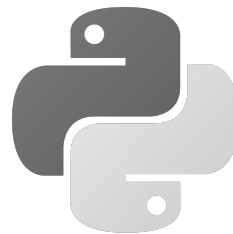
# How should I interact with TTK?

- **End users**

- Plugins for ParaView
  - De-facto standard in scientific computing
- Light Python API
  - Fast scripting
- Inviwo environment
  - <https://inviwo.org/>

- **Developers**

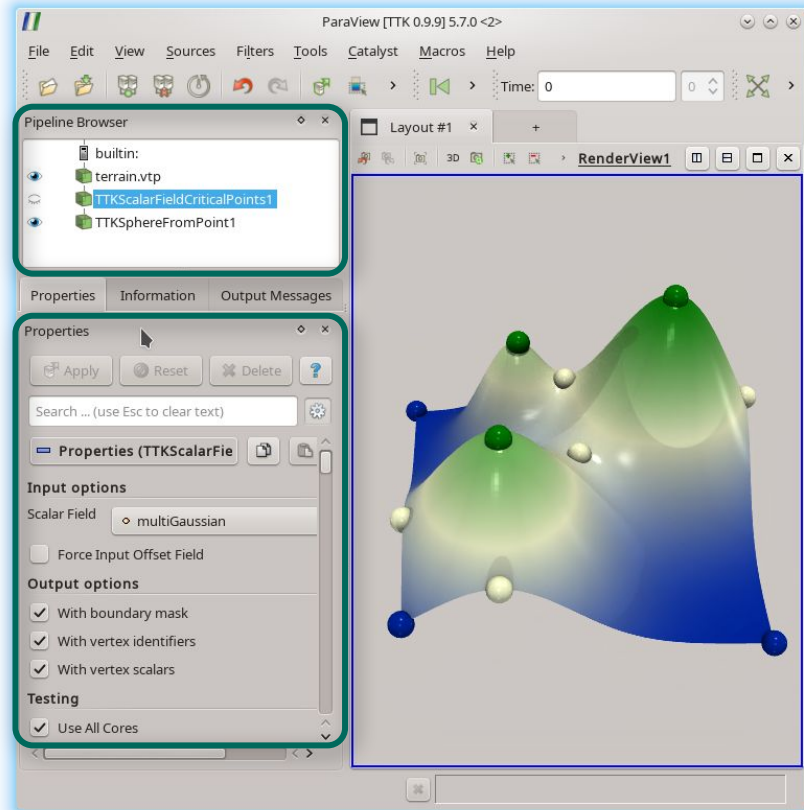
- Recommended
  - VTK-based APIs: C++ & Python
- Advanced
  - Dependency-free plain C++ API



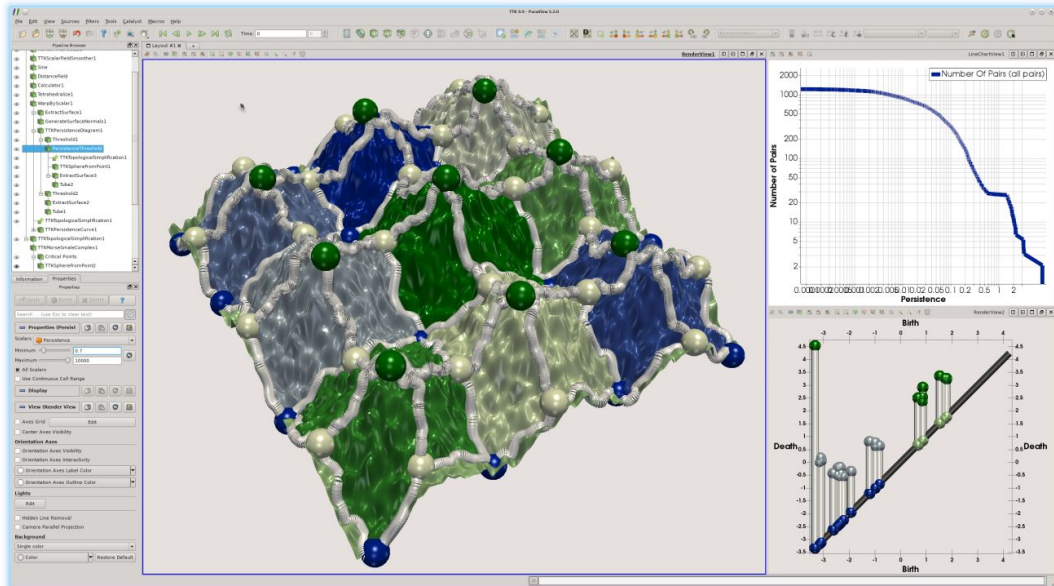
# About ParaView

- **Interesting features**

- Rich IO support
- Modern rendering
  - OpenGL, OSPRay, Optix
- Advanced user interface
- “Visual” programming
  - Pipeline philosophy
- Python scripting

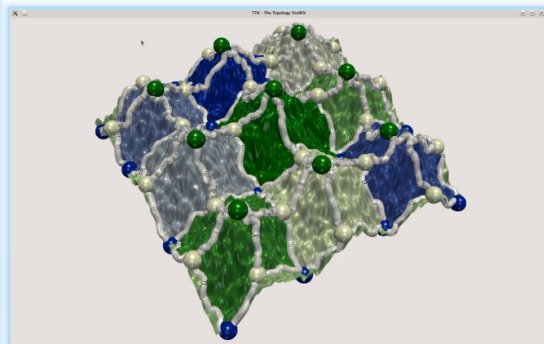
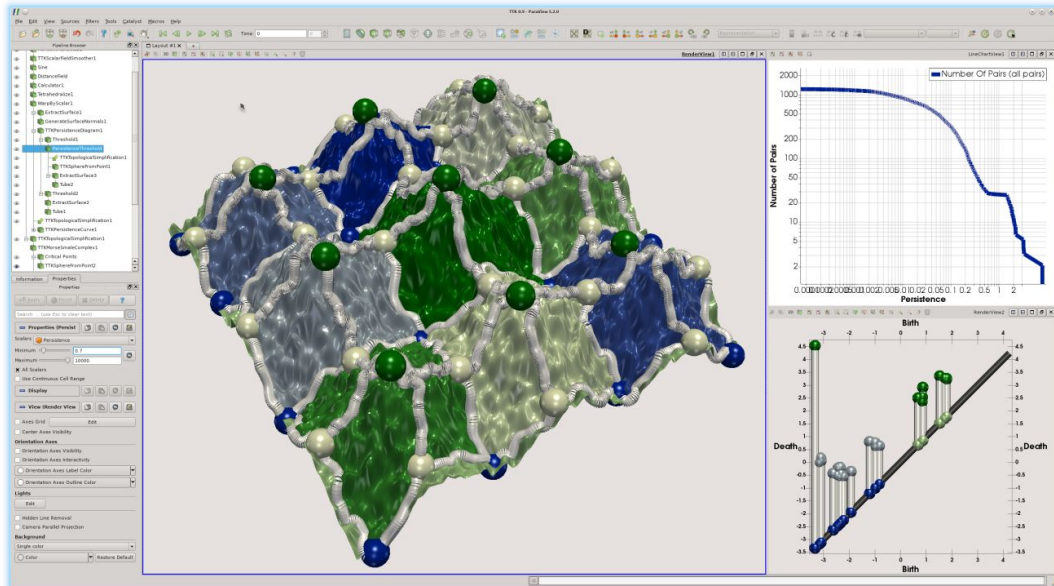


# User experience

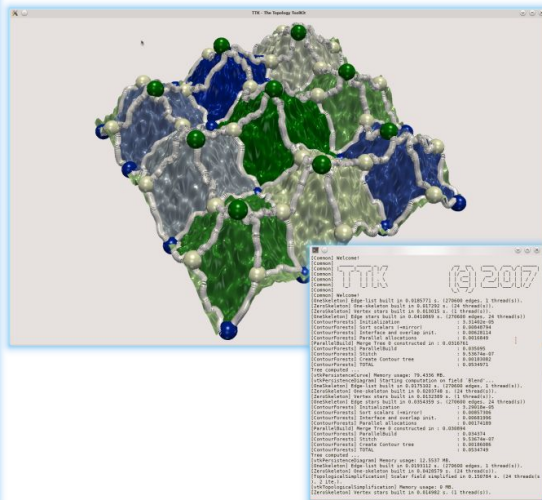
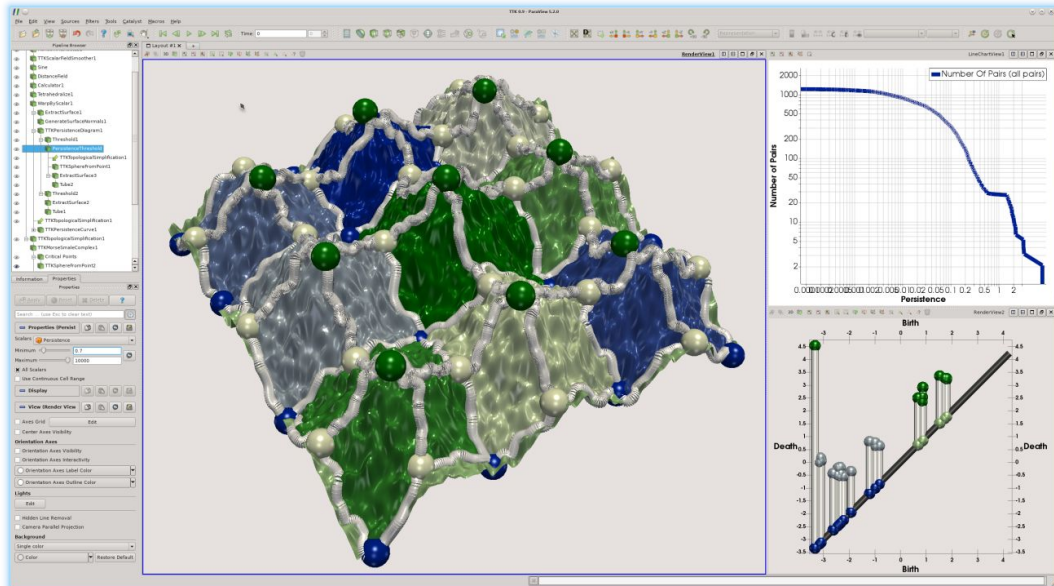




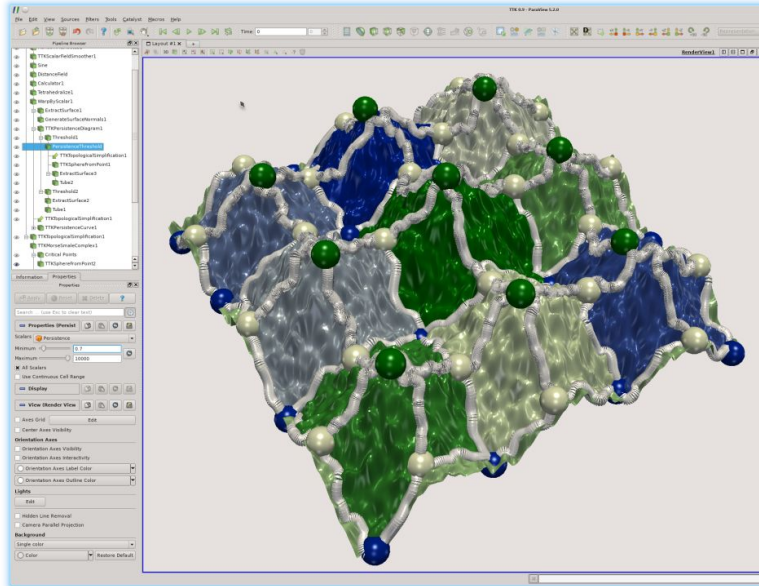
# User experience



# User experience



# User experience

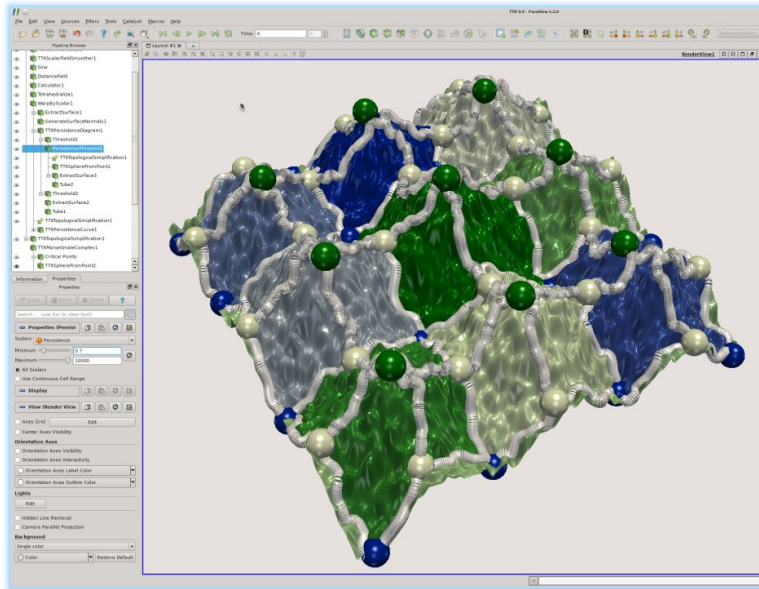


```
File Edit View Build Objects Republish Sessions Tools Settings Help
Python 3.7.4 Shell
File Open Recent Document Next Document Save Save As Close
# Python 3.7.4 Shell
from parview.simple import *
#
# loading the input data
inputData = vtkStructuredGridReaderFromFileName('inputData.vtu')
#
# computing the connectivity data
parastencils = vtkParastencilReader(inputData)
#
# computing the previous degree
parastencilDegree = vtkParastencilDegreeCalculator()
#
# defining the critical point actor
criticalPointActor = vtkCriticalPointActor()
criticalPointActor.SetScalars('CriticalPointScalars')
criticalPointActor.ThresholdRange = [-1, 0.000000]
#
# overlaying the mesh with critical points
parastencilPairs = vtkParastencilPairs()
parastencilPairs.SetScalars('CriticalPointScalars')
parastencilPairs.ThresholdRange = [-1, 0.000000]
#
# computing the connectivity data
connectivityFunction = vtkConnectivityFunction()
connectivityFunction.SetScalars('CriticalPointScalars')
connectivityFunction.ConnectivityThresholdRange = [-1, 0.000000]
#
# applying the Hessian matrix
hessianCalculator = vtkHessianCalculator()
hessianCalculator.SetScalars('CriticalPointScalars')
hessianCalculator.SetOutputFieldId = 0
#
# using the actor actor
hessianActor = vtkHessianActor()
hessianActor.SetScalars('CriticalPointScalars')
hessianActor.SetOutputFieldId = 0
hessianActor.SetRepresentation('vtkHessianActor::Representation::POINTS')
hessianActor.SetRepresentationColor('CriticalPointScalars')
```



Python  
(21 lines)

# User experience



```
1 # vtkPythonShell: simple import *
2
3 # Import the input data
4 inputData = vtkStructuredGridReaderFromFile(InputData.vtu)
5
6 # Read the input data
7 input = inputData.GetOutput()
8
9 # Compute the persistence curve
10 persistenceCurve = vtkPersistenceCurve(inputData)
11
12 # Compute the persistence diagram
13 persistenceDiagram = vtkPersistenceDiagram(persistenceCurve)
14
15 # Selecting the critical point sets
16 criticalPointSets = vtkCriticalPointSets(persistenceDiagram)
17 criticalPointSets.ThresholdRange = [1, 100000]
18
19 # Computing the most persistent points
20 mostPersistentPoints = vtkMostPersistentPoints(criticalPointSets)
21 mostPersistentPoints.ThresholdRange = [1, 100000]
22
23 # Selecting the input data to process non-persistent points
24 topologicalInformation = vtkTopologicalInformation(mostPersistentPoints)
25 nonPersistentComplex = vtkNonPersistentComplex(topologicalInformation)
26 nonPersistentComplex.ThresholdRange = [1, 100000]
27
28 # Computing the normal-tube complex
29 normalTubeComplex = vtkNormalTubeComplex(nonPersistentComplex)
30
31 # Selecting the output data
32 output = normalTubeComplex.GetOutput()
33
34 # Saving the persistence vtu
35 saveFile("persistence.vtu", normalTubeComplex.GetOutput())
36
37 # Saving the persistence vtu
38 saveFile("persistence.vtu", normalTubeComplex.GetOutput())
```

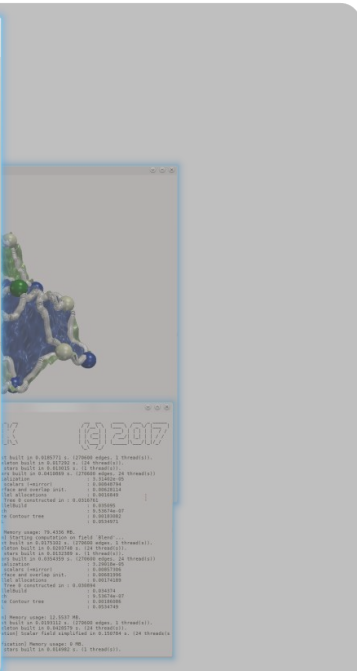
Line 3 of 35 Col 1    Line: VTK NORMAL POKE    script.cpp:50:855:1

Python  
(21 lines)

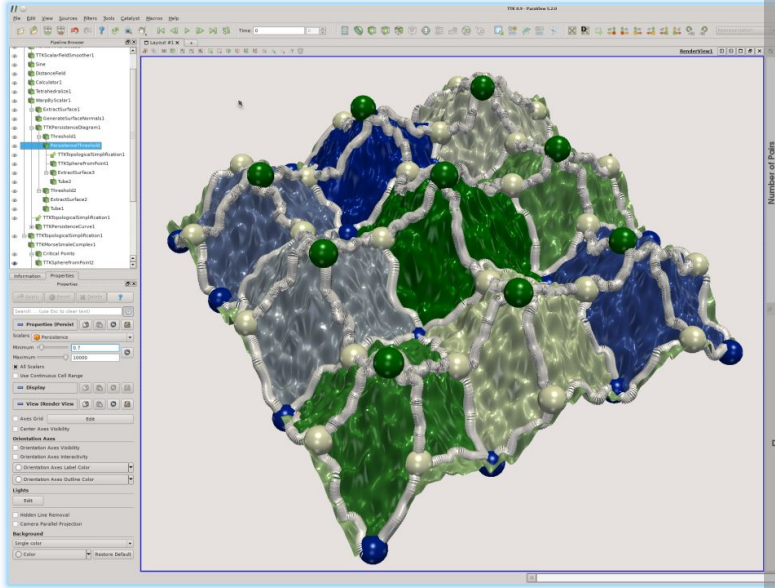
```
1 # vtkPythonShell: simple import *
2
3 # Import the input data
4 inputData = vtkStructuredGridReaderFromFile(InputData.vtu)
5
6 # Read the input data
7 input = inputData.GetOutput()
8
9 # Compute the persistence curve
10 persistenceCurve = vtkPersistenceCurve(inputData)
11
12 # Compute the persistence diagram
13 persistenceDiagram = vtkPersistenceDiagram(persistenceCurve)
14
15 # Selecting the critical point sets
16 criticalPointSets = vtkCriticalPointSets(persistenceDiagram)
17 criticalPointSets.ThresholdRange = [1, 100000]
18
19 # Computing the most persistent points
20 mostPersistentPoints = vtkMostPersistentPoints(criticalPointSets)
21 mostPersistentPoints.ThresholdRange = [1, 100000]
22
23 # Selecting the input data to process non-persistent points
24 topologicalInformation = vtkTopologicalInformation(mostPersistentPoints)
25 nonPersistentComplex = vtkNonPersistentComplex(topologicalInformation)
26 nonPersistentComplex.ThresholdRange = [1, 100000]
27
28 # Computing the normal-tube complex
29 normalTubeComplex = vtkNormalTubeComplex(nonPersistentComplex)
30
31 # Selecting the output data
32 output = normalTubeComplex.GetOutput()
33
34 # Saving the persistence vtu
35 saveFile("persistence.vtu", normalTubeComplex.GetOutput())
36
37 # Saving the persistence vtu
38 saveFile("persistence.vtu", normalTubeComplex.GetOutput())
```

Line 3 of 77 Col 1    Line: VTK NORMAL POKE    main.cpp:52:855:1

VTK/C++  
(37 lines)



# User experience



```
1 # Python code to generate a VTK point cloud of the protein structure
2 # based on the PDB structure file 1D55A.pdb.
3
4 # Step 1: Read the PDB file and extract the atom coordinates
5 import sys
6 from pathlib import Path
7
8 # Step 2: Create a VTK point cloud
9 from vtkmodules.vtkCommonDataModel import vtkPoints, vtkPointSource
10 from vtkmodules.vtkCommonMath import vtkMath
11
12 # Step 3: Write the point cloud to a file
13 import numpy as np
14
15 # Step 4: Visualize the point cloud
16 from vtkmodules.vtkRenderingCore import vtkRenderer, vtkRenderWindow,
17 vtkRenderWindowInteractor
18 from vtkmodules.vtkFiltersSources import vtkPolyDataNormals
19
20 # Step 5: Clean up
21 del sys
22 del Path
23 del vtkPoints
24 del vtkPointSource
25 del vtkMath
26 del np
27 del vtkRenderer
28 del vtkRenderWindow
29 del vtkRenderWindowInteractor
30 del vtkPolyDataNormals
31
32 # Step 6: End of script
33
```

Line 3 of 35 Col 1    LINE VTK NORMAL MODE    main.cpp (2) /SD/05501  
Q Search and Replace Current Project Documents Projects Terminal Build Output

Python  
(21 lines)

```
1 # C++ code to generate a VTK point cloud of the protein structure
2 # based on the PDB structure file 1D55A.pdb.
3
4 # Step 1: Read the PDB file and extract the atom coordinates
5 # Step 2: Create a VTK point cloud
6 # Step 3: Write the point cloud to a file
7 # Step 4: Visualize the point cloud
8 # Step 5: Clean up
9 # Step 6: End of script
10
```

Line 3 of 77 Col 1    LINE VTK NORMAL MODE    main.cpp (2) /SD/05501  
Q Search and Replace Current Project Documents Projects Terminal Build Output

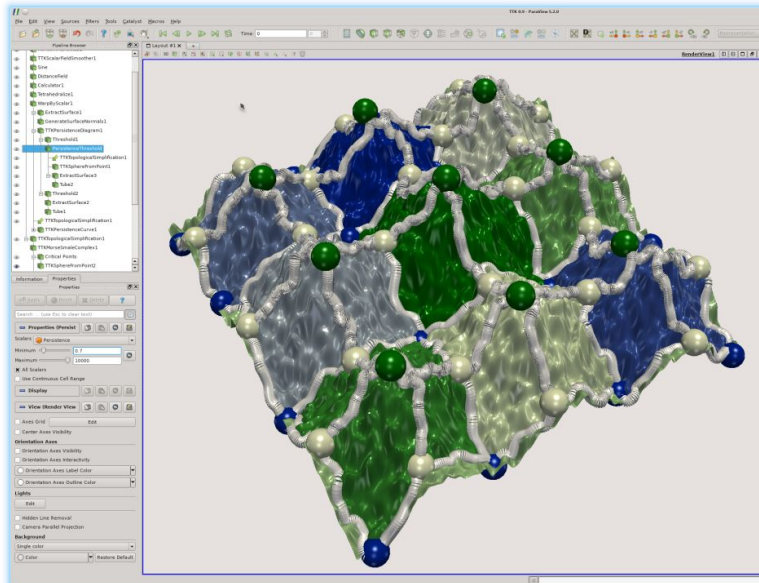
VTK/C++  
(37 lines)

```
1 # Pure C++ code to generate a VTK point cloud of the protein structure
2 # based on the PDB structure file 1D55A.pdb.
3
4 # Step 1: Read the PDB file and extract the atom coordinates
5 # Step 2: Create a VTK point cloud
6 # Step 3: Write the point cloud to a file
7 # Step 4: Visualize the point cloud
8 # Step 5: Clean up
9 # Step 6: End of script
10
```

Line 33 of 288 Col 7    LINE VTK NORMAL MODE    main.cpp (2) /SD/05501  
Q Search and Replace Current Project Documents Projects Terminal Build Output

Pure C++  
(101 lines)

# User experience



Number of Points

Vtk Normal Node

Python (21 lines)

VTK/C++ (37 lines)

Pure C++ (101 lines)

```
1 # Python script to generate a VTK normal node from a parameter file.
2 #
3 # Parameters:
4 #   - input: A file containing the input data.
5 #   - output: A file containing the output data.
6 #
7 # Usage:
8 #   python script.py input output
9 #
10 #
11 #
12 #
13 #
14 #
15 #
16 #
17 #
18 #
19 #
20 #
21 #
22 #
23 #
24 #
25 #
26 #
27 #
28 #
29 #
30 #
31 #
32 #
33 #
34 #
35 #
36 #
37 #
38 #
39 #
40 #
41 #
42 #
43 #
44 #
45 #
46 #
47 #
48 #
49 #
50 #
51 #
52 #
53 #
54 #
55 #
56 #
57 #
58 #
59 #
60 #
61 #
62 #
63 #
64 #
65 #
66 #
67 #
68 #
69 #
70 #
71 #
72 #
73 #
74 #
75 #
76 #
77 #
78 #
79 #
80 #
81 #
82 #
83 #
84 #
85 #
86 #
87 #
88 #
89 #
90 #
91 #
92 #
93 #
94 #
95 #
96 #
97 #
98 #
99 #
100 #
101 #
102 #
103 #
104 #
105 #
106 #
107 #
108 #
109 #
110 #
111 #
112 #
113 #
114 #
115 #
116 #
117 #
118 #
119 #
120 #
121 #
122 #
123 #
124 #
125 #
126 #
127 #
128 #
129 #
130 #
131 #
132 #
133 #
134 #
135 #
136 #
137 #
138 #
139 #
140 #
141 #
142 #
143 #
144 #
145 #
146 #
147 #
148 #
149 #
150 #
151 #
152 #
153 #
154 #
155 #
156 #
157 #
158 #
159 #
160 #
161 #
162 #
163 #
164 #
165 #
166 #
167 #
168 #
169 #
170 #
171 #
172 #
173 #
174 #
175 #
176 #
177 #
178 #
179 #
180 #
181 #
182 #
183 #
184 #
185 #
186 #
187 #
188 #
189 #
190 #
191 #
192 #
193 #
194 #
195 #
196 #
197 #
198 #
199 #
200 #
201 #
202 #
203 #
204 #
205 #
206 #
207 #
208 #
209 #
210 #
211 #
212 #
213 #
214 #
215 #
216 #
217 #
218 #
219 #
220 #
221 #
222 #
223 #
224 #
225 #
226 #
227 #
228 #
229 #
230 #
231 #
232 #
233 #
234 #
235 #
236 #
237 #
238 #
239 #
240 #
241 #
242 #
243 #
244 #
245 #
246 #
247 #
248 #
249 #
250 #
251 #
252 #
253 #
254 #
255 #
256 #
257 #
258 #
259 #
260 #
261 #
262 #
263 #
264 #
265 #
266 #
267 #
268 #
269 #
270 #
271 #
272 #
273 #
274 #
275 #
276 #
277 #
278 #
279 #
280 #
281 #
282 #
283 #
284 #
285 #
286 #
287 #
288 #
289 #
290 #
291 #
292 #
293 #
294 #
295 #
296 #
297 #
298 #
299 #
300 #
301 #
302 #
303 #
304 #
305 #
306 #
307 #
308 #
309 #
310 #
311 #
312 #
313 #
314 #
315 #
316 #
317 #
318 #
319 #
320 #
321 #
322 #
323 #
324 #
325 #
326 #
327 #
328 #
329 #
330 #
331 #
332 #
333 #
334 #
335 #
336 #
337 #
338 #
339 #
340 #
341 #
342 #
343 #
344 #
345 #
346 #
347 #
348 #
349 #
350 #
351 #
352 #
353 #
354 #
355 #
356 #
357 #
358 #
359 #
360 #
361 #
362 #
363 #
364 #
365 #
366 #
367 #
368 #
369 #
370 #
371 #
372 #
373 #
374 #
375 #
376 #
377 #
378 #
379 #
380 #
381 #
382 #
383 #
384 #
385 #
386 #
387 #
388 #
389 #
390 #
391 #
392 #
393 #
394 #
395 #
396 #
397 #
398 #
399 #
400 #
401 #
402 #
403 #
404 #
405 #
406 #
407 #
408 #
409 #
410 #
411 #
412 #
413 #
414 #
415 #
416 #
417 #
418 #
419 #
420 #
421 #
422 #
423 #
424 #
425 #
426 #
427 #
428 #
429 #
430 #
431 #
432 #
433 #
434 #
435 #
436 #
437 #
438 #
439 #
440 #
441 #
442 #
443 #
444 #
445 #
446 #
447 #
448 #
449 #
450 #
451 #
452 #
453 #
454 #
455 #
456 #
457 #
458 #
459 #
460 #
461 #
462 #
463 #
464 #
465 #
466 #
467 #
468 #
469 #
470 #
471 #
472 #
473 #
474 #
475 #
476 #
477 #
478 #
479 #
480 #
481 #
482 #
483 #
484 #
485 #
486 #
487 #
488 #
489 #
490 #
491 #
492 #
493 #
494 #
495 #
496 #
497 #
498 #
499 #
500 #
501 #
502 #
503 #
504 #
505 #
506 #
507 #
508 #
509 #
510 #
511 #
512 #
513 #
514 #
515 #
516 #
517 #
518 #
519 #
520 #
521 #
522 #
523 #
524 #
525 #
526 #
527 #
528 #
529 #
530 #
531 #
532 #
533 #
534 #
535 #
536 #
537 #
538 #
539 #
540 #
541 #
542 #
543 #
544 #
545 #
546 #
547 #
548 #
549 #
550 #
551 #
552 #
553 #
554 #
555 #
556 #
557 #
558 #
559 #
560 #
561 #
562 #
563 #
564 #
565 #
566 #
567 #
568 #
569 #
570 #
571 #
572 #
573 #
574 #
575 #
576 #
577 #
578 #
579 #
580 #
581 #
582 #
583 #
584 #
585 #
586 #
587 #
588 #
589 #
590 #
591 #
592 #
593 #
594 #
595 #
596 #
597 #
598 #
599 #
600 #
601 #
602 #
603 #
604 #
605 #
606 #
607 #
608 #
609 #
610 #
611 #
612 #
613 #
614 #
615 #
616 #
617 #
618 #
619 #
620 #
621 #
622 #
623 #
624 #
625 #
626 #
627 #
628 #
629 #
630 #
631 #
632 #
633 #
634 #
635 #
636 #
637 #
638 #
639 #
640 #
641 #
642 #
643 #
644 #
645 #
646 #
647 #
648 #
649 #
650 #
651 #
652 #
653 #
654 #
655 #
656 #
657 #
658 #
659 #
660 #
661 #
662 #
663 #
664 #
665 #
666 #
667 #
668 #
669 #
670 #
671 #
672 #
673 #
674 #
675 #
676 #
677 #
678 #
679 #
680 #
681 #
682 #
683 #
684 #
685 #
686 #
687 #
688 #
689 #
690 #
691 #
692 #
693 #
694 #
695 #
696 #
697 #
698 #
699 #
700 #
701 #
702 #
703 #
704 #
705 #
706 #
707 #
708 #
709 #
710 #
711 #
712 #
713 #
714 #
715 #
716 #
717 #
718 #
719 #
720 #
721 #
722 #
723 #
724 #
725 #
726 #
727 #
728 #
729 #
730 #
731 #
732 #
733 #
734 #
735 #
736 #
737 #
738 #
739 #
740 #
741 #
742 #
743 #
744 #
745 #
746 #
747 #
748 #
749 #
750 #
751 #
752 #
753 #
754 #
755 #
756 #
757 #
758 #
759 #
760 #
761 #
762 #
763 #
764 #
765 #
766 #
767 #
768 #
769 #
770 #
771 #
772 #
773 #
774 #
775 #
776 #
777 #
778 #
779 #
780 #
781 #
782 #
783 #
784 #
785 #
786 #
787 #
788 #
789 #
790 #
791 #
792 #
793 #
794 #
795 #
796 #
797 #
798 #
799 #
800 #
801 #
802 #
803 #
804 #
805 #
806 #
807 #
808 #
809 #
810 #
811 #
812 #
813 #
814 #
815 #
816 #
817 #
818 #
819 #
820 #
821 #
822 #
823 #
824 #
825 #
826 #
827 #
828 #
829 #
830 #
831 #
832 #
833 #
834 #
835 #
836 #
837 #
838 #
839 #
840 #
841 #
842 #
843 #
844 #
845 #
846 #
847 #
848 #
849 #
850 #
851 #
852 #
853 #
854 #
855 #
856 #
857 #
858 #
859 #
860 #
861 #
862 #
863 #
864 #
865 #
866 #
867 #
868 #
869 #
870 #
871 #
872 #
873 #
874 #
875 #
876 #
877 #
878 #
879 #
880 #
881 #
882 #
883 #
884 #
885 #
886 #
887 #
888 #
889 #
890 #
891 #
892 #
893 #
894 #
895 #
896 #
897 #
898 #
899 #
900 #
901 #
902 #
903 #
904 #
905 #
906 #
907 #
908 #
909 #
910 #
911 #
912 #
913 #
914 #
915 #
916 #
917 #
918 #
919 #
920 #
921 #
922 #
923 #
924 #
925 #
926 #
927 #
928 #
929 #
930 #
931 #
932 #
933 #
934 #
935 #
936 #
937 #
938 #
939 #
940 #
941 #
942 #
943 #
944 #
945 #
946 #
947 #
948 #
949 #
950 #
951 #
952 #
953 #
954 #
955 #
956 #
957 #
958 #
959 #
960 #
961 #
962 #
963 #
964 #
965 #
966 #
967 #
968 #
969 #
970 #
971 #
972 #
973 #
974 #
975 #
976 #
977 #
978 #
979 #
980 #
981 #
982 #
983 #
984 #
985 #
986 #
987 #
988 #
989 #
990 #
991 #
992 #
993 #
994 #
995 #
996 #
997 #
998 #
999 #
1000 #
```

• <https://github.com/topology-tool-kit/ttk/tree/dev/examples>

# TTK data & examples

The screenshot shows a web browser window with the URL `topology-tool-kit.github.io/examples/index.html`. The page title is "TTK Examples".

**TTK Examples**  
Welcome to the TTK Examples!

- 1-Manifold Learning
- 1-Manifold Learning Circles
- 2-Manifold Learning
- Builtin example 1
- Builtin example 2
- CinemaIO
- Contour Tree Alignment
- CT bones
- Dragon
- Geometry Approximation
- Harmonic Skeleton
- Image Processing
- Interaction sites
- Karhunen-Love Digits 64-Dimensions
- Manifold Check
- Merge Tree Clustering
- Merge Tree Temporal Reduction
- Morse molecule
- Morse persistence
- Morse-Smale Quadrangulation
- Nested Tracking From Overlap
- Persistence Clustering 0
- Persistence Clustering 1
- Persistence Clustering 2
- Persistence Clustering 3
- Persistence Clustering 4

Welcome to the TTK Examples!

This website hosts a list of data analysis pipelines exemplifying the usage of TTK with ParaView and its Python API `pvpython`.

This website is targeting novice users who are not power users of ParaView but who would like to get started with topological data analysis with TTK in Python.

Each example includes:

- a screenshot (or a tutorial video)
- a short description
- the command line to reproduce the example with ParaView
- the corresponding Python code, to:
  - load the input data
  - execute the analysis pipeline
  - store the output to disk (for later analysis or visualization, e.g. with ParaView)
- a description of the inputs and outputs
- pointers to the corresponding C++/Python documentation

This documentation assumes a default TTK installation (with the `pvpython` API support enabled) and that the repository `ttk-data` has been downloaded locally.

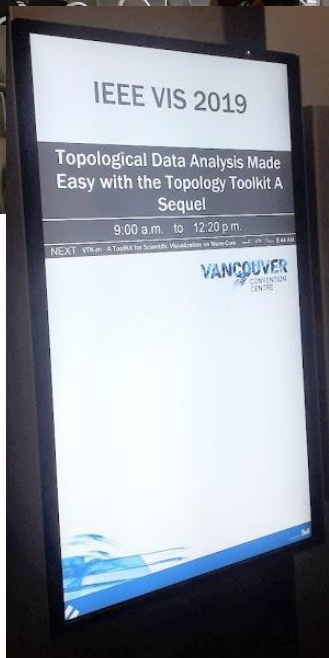
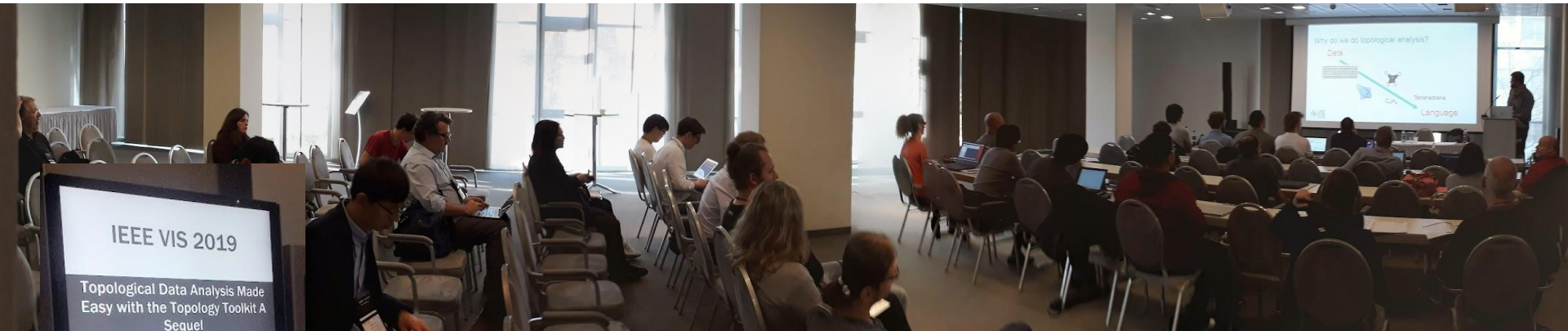
If you have any questions regarding these examples, please let us know by sending an email to the [TTK user mailing list](#)!

Scalar data

**Table of contents**

- Scalar data
- Bivariate scalar data
- Uncertain scalar data
- Time-varying scalar data
- Ensemble scalar data
- High-dimensional / point cloud data
- In-situ features
- Misc features

# IEEE VIS Tutorials 2018-2021



- **Overview**

- Presentations + hands-on
- ~10 speakers, attendance: ~50
- <https://topology-tool-kit.github.io/ieeeVisTutorial.html>
- Slides, data, examples, pre-installed virtual machines, more



- **Large-scale data reduction**
  - Until 2025
  - High-Performance TDA
  - Statistical framework for ensembles
  
- **We're hiring!**
  - 4 1 Ph.D. offers
  - 2 1 Post-docs
  - 2 1 engineers
  
- **Website**
  - <http://erc-tori.github.io/>



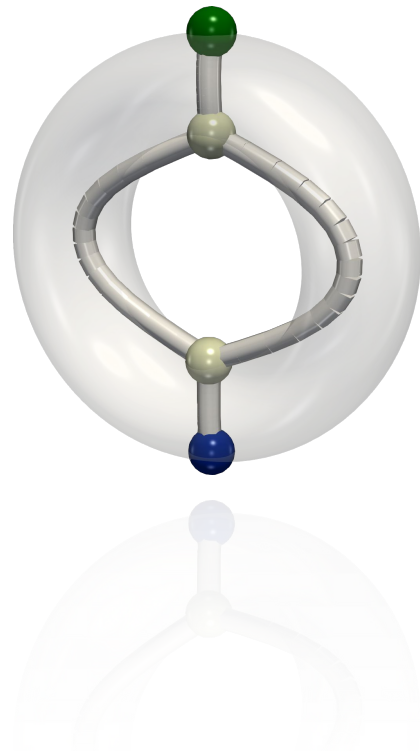
# Take-home messages

- **Data on meshes, or meshable data?**

- Features of interest?
- Recover 'structural' information

- ⇒ Topological Data Analysis

- Robust, multiscale
- Successful in applications
- Software available



# Thanks!

- Papers, video, code, teaching material, exercise packages, tutorials...

- <http://lip6.fr/Julien.Tierny>, <https://twitter.com/JulienTierny>
- <http://topology-tool-kit.github.io>

- We are hiring!

- 1 Ph.D. student, 1 post-doc, 1 engineer
- <http://erc-tori.github.io/>



**WE WANT YOU!**

- Thanks to all my co-authors!

- Roberto Alvarez-Boto, John Bell, Timo Bremer, Bert Buchholz, Joseph Budin, Hamish Carr, Amit Chattopadhyay, Fang Chen, Bruno Conche, Julia Contreras, Joel Daniels, Mohamed Daoudi, Marcus Day, Julie Delon, Harish Doraiswamy, Florent Dupont, Tiago Etienne, Noura Faraj, Guillaume Favelier, Tarik Filali, Pierre Fortin, Juliana Freire, Mariem Gargouri, Christoph Garth, Zhao Geng, Brad Grimm, Charles Gueunet, Pierre Guillou, Attila Gyulassy, Hans Hagen, Bernd Hamann, Mike Kirby, Pavol Klacansky, Scott Klasky, Aaron Knoll, Erwan Jolivet, Julien Jomier, Ayla Khan, Guillaume Lavoue, Josh Levine, Lauro Lins, Jonas Lukasczyk, Pooran Memari, Michael Michaux, Gustavo Nonato, Małgorzata Olejniczak, Valerio Pascucci, Philippe Petit, Sujin Philip, Jean-Philip Piquemal, Melanie Plainchault, Norbet Podhorszki, Mathieu Pont, Daisuke Sakurai, Joseph Salmon, Emanuele Santos, Carlos Scheidegger, Claudio Silva, Eddie Simon, Maxime Soler, Brian Summa, Roselyne Tchoua, Jean-Marc Thiery, Will Usher, Jean-Philippe Vandeborre, Jules Vidal, Ana Vintescu, Gunther Weber, Qi Wu

- Questions?

# Main publications

- **J. Tierny**, "Topological Data Analysis for Scientific Visualization", Springer 2018.
- V. Pascucci, X. Tricoche, H. Hagen, **J. Tierny**, "Topological methods in data analysis and visualization, Springer 2010.
- Jules Vidal, Joseph Budin, Julien Tierny, "Progressive Wasserstein Barycenters of Persistence Diagrams", IEEE Trans. On Visualization and Computer Graphics (Proc. of IEEE VIS 2019), **Best paper honorable mention**.
- M. Soler, M. Petitfrere, G. Darche, M. Plainchault, B. Conche, **J. Tierny**, "Ranking Viscous Finger Simulations to an Acquired Ground Truth with Topology-Aware Matchings", IEEE LDAV 2019, **Best paper award**.
- T. Bridel-Bertomeu, B. Fovet, **J. Tierny**, F. Vivodtzev, "Topological Analysis of High Velocity Turbulent Flow", Proc. of IEEE LDAV 2019 (shorts).
- C. Gueunet, P. Fortin, J. Jomier, **J. Tierny**, "Task-based Augmented Reeb Graphs with Dynamic ST-Trees", Proc. of EGPGV 2019.
- C. Gueunet, P. Fortin, J. Jomier, **J. Tierny**, "Task-based Augmented Contour Trees with Fibonacci Heaps", IEEE Trans. On Parallel and Distributed Systems, 2019.
- G. Favelier, N. Faraj, B. Summa, **J. Tierny**, "Persistence Atlas for Critical Point Variability in Ensembles", IEEE Trans. On Visualization and Computer Graphics (Proc. of IEEE VIS 2018)
- M. Soler, M. Plainchault, B. Conche, **J. Tierny**, "Lifted Wasserstein Matcher for Fast and Robust Topology Tracking", Proc. of IEEE LDAV 2018. **Best paper honorable mention**.
- M. Soler, M. Plainchault, B. Conche, **J. Tierny**, "Topologically Controlled Lossy Compression", Proc. of IEEE PacificViz 2018.
- **J. Tierny**, G. Favelier, J. Levine, C. Gueunet, M. Michaux, "The Topology Toolkit", IEEE Trans. On Visualization and Computer Graphics (Proc. of IEEE VIS 2017), **Best paper honorable mention**.
- B. Summa, **J. Tierny**, V. Pascucci, "Visualizing the Uncertainty of Graph-based 2D Segmentation with Min-path Stability", Computer Graphics Forum Journal (Proc. of EuroVis 2017)
- C. Gueunet, P. Fortin, J. Jomier, **J. Tierny**, "Task-based Augmented Merge Trees with Fibonacci Heaps", Proc. of IEEE LDAV 2017
- **J. Tierny**, H. Carr, "Jacobi Fiber Surfaces for Bivariate Reeb Space Computation", IEEE Trans. On Visualization and Computer Graphics (Proc. of IEEE VIS 2016). **Best paper award**.
- P. Klacansky, **J. Tierny**, H. Carr, Z. Geng, "Fast and Exact Fiber Surfaces for Tetrahedral Meshes", IEEE Trans. On Visualization and Computer Graphics 2016
- C. Gueunet, P. Fortin, J. Jomier, **J. Tierny**, "Contour Forests: Fast Multi-threaded Augmented Contour Trees", Proc. of IEEE LDAV 2016
- S. Philip, B. Summa, **J. Tierny**, T. Bremer, V. Pascucci, "Distributed Seams for Gigapixel Panoramas", IEEE Trans. On Visualization and Computer Graphics 2015
- H. Carr, Z. Geng, **J. Tierny**, A. Chattopadhyay, A. Knoll, "Fiber Surfaces: Generalizing Isosurfaces to Bivariate Data", Computer Graphics Forum Journal (Proc. of EuroVis 2015)
- A. Gyulassy, D. Guenther, J. Levine, **J. Tierny**, V. Pascucci, "Conforming Morse-Smale Complexes", IEEE Trans. On Visualization and Computer Graphics (Proc. of IEEE VIS 2014)
- D. Guenther, R. Alvarez-Boto, J. Contreras, J.P. Piquemal, **J. Tierny**, "Characterizing Molecular Interactions in Chemical Systems", IEEE Trans. On Visualization and Computer Graphics (Proc. of IEEE VIS 2014)
- D. Guenther, J. Salmon, **J. Tierny**, "Mandatory Critical Points of 2D Uncertain Scalar Fields", Computer Graphics Forum Journal (Proc. of EuroVis 2014)
- **J. Tierny** and V. Pasucci, "Generalized Topological Simplification of Scalar Fields on Surfaces, IEEE Trans. On Visualization and Computer Graphics (Proc. of IEEE VIS 2012)
- T. Etienne, G. Nonato, C. Scheidegger, **J. Tierny**, T. Peters, V. Pascucci, M. Kirby, C. Silva, "Topology verification for isosurface extraction", IEEE Trans. On Visualization and Computer Graphics 2012
- **J. Tierny**, J. Daniels, G. Nonato, V. Pascucci, C. Silva, "Interactive quadrangulation with Reeb atlases and connectivity textures", IEEE Trans. On Visualization and Computer Graphics 2012
- T. Bremer, G. Weber, **J. Tierny**, V. Pascucci, M. Day, J. Bell, "Interactive exploration and analysis of large scale simulations using topology-based data segmentation", IEEE Trans. On Visualization and Computer Graphics 2011
- **J. Tierny**, A. Gyulassy, E. Simon, V. Pascucci, "Loop surgery for volumetric meshes: Reeb graphs reduced to contour trees", IEEE Trans. On Visualization and Computer Graphics (Proc. of IEEE VIS 2009)