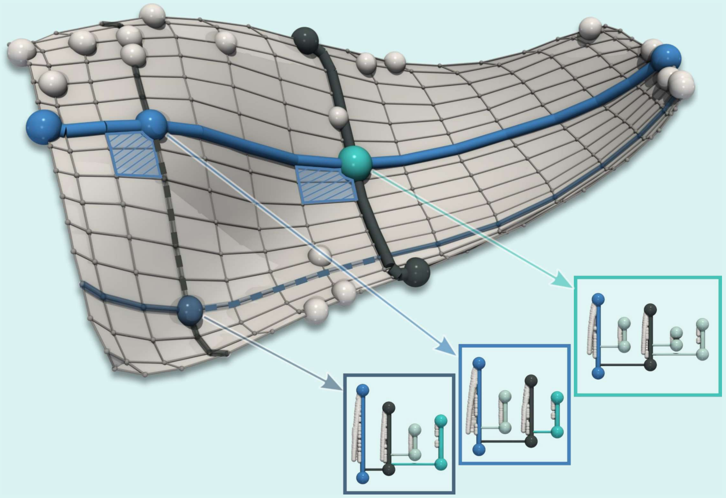
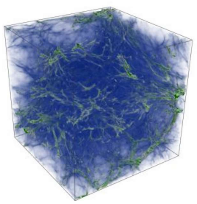


Principal Geodesic Analysis Of Merge Trees (and Persistence Diagrams)

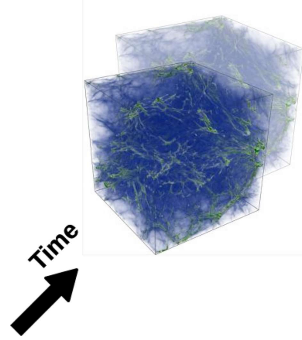
Mathieu Pont, Jules Vidal, Julien Tierny

SORBONNE UNIVERSITÉ
CRÉATEURS DE FUTURS DEPUIS 1257

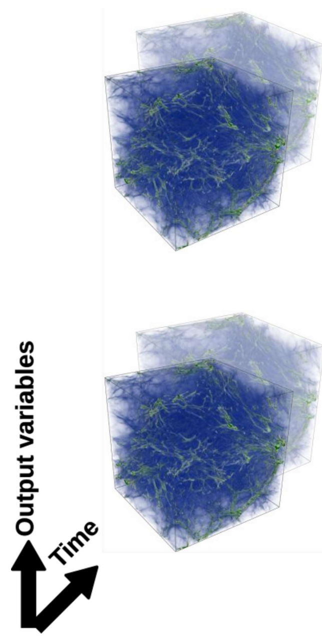
Topological data reduction



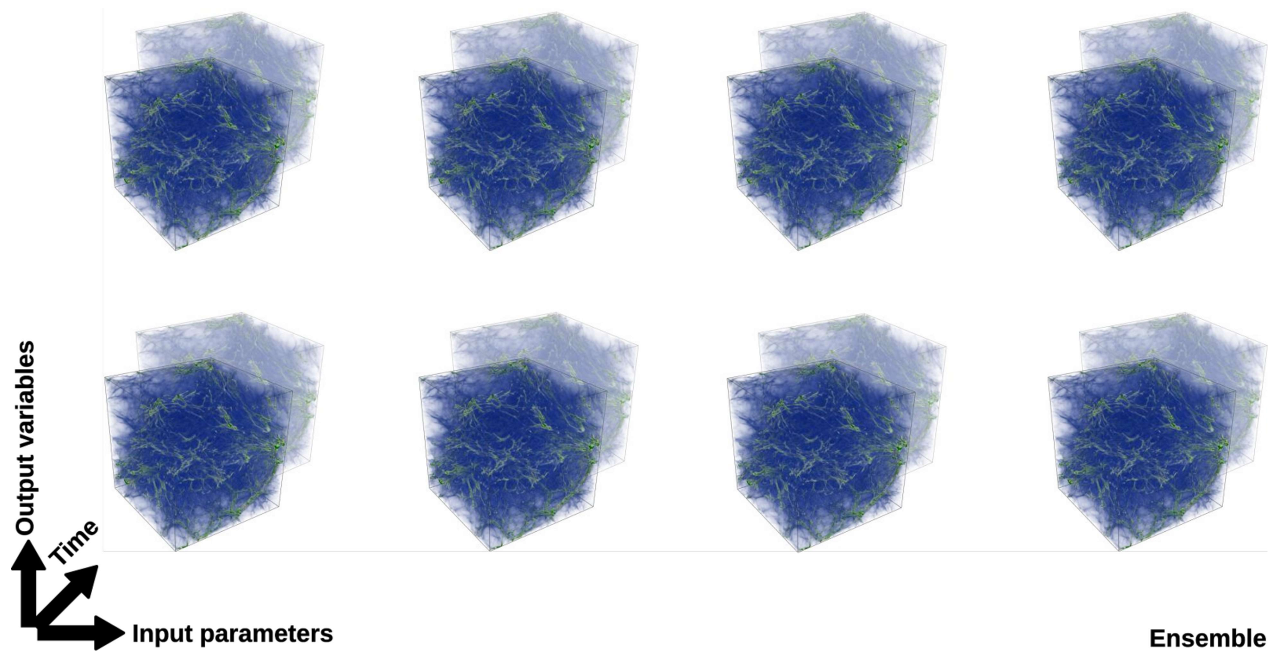
Topological data reduction



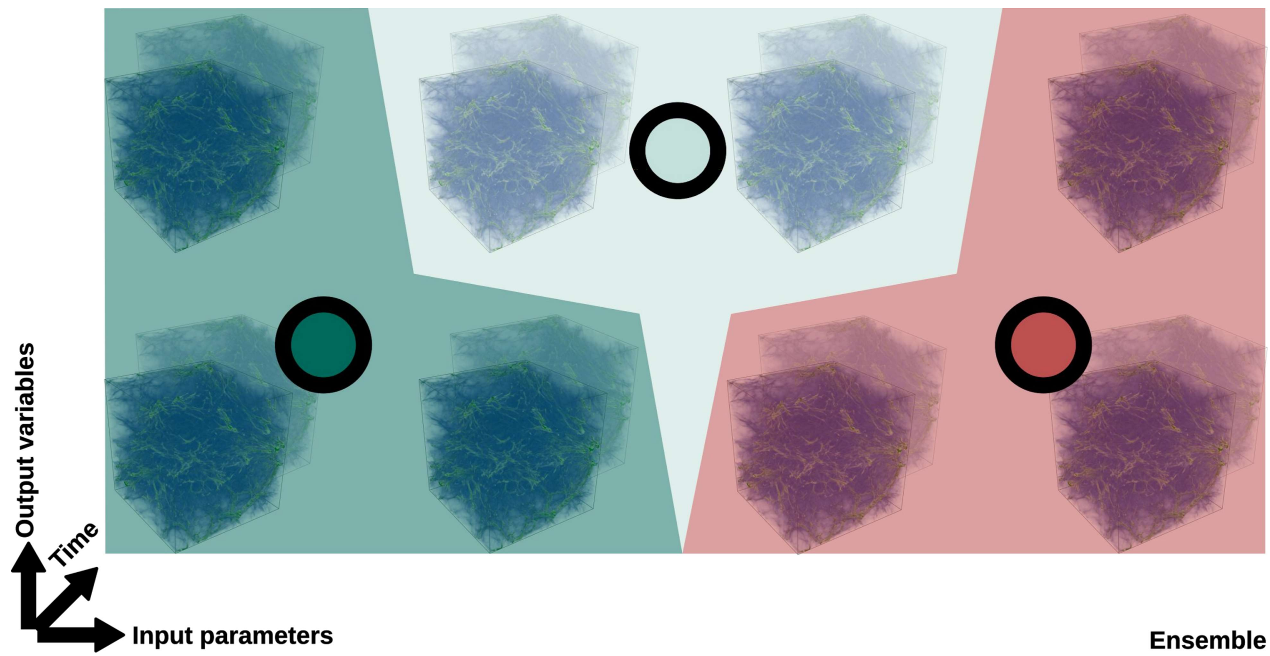
Topological data reduction



Topological data reduction

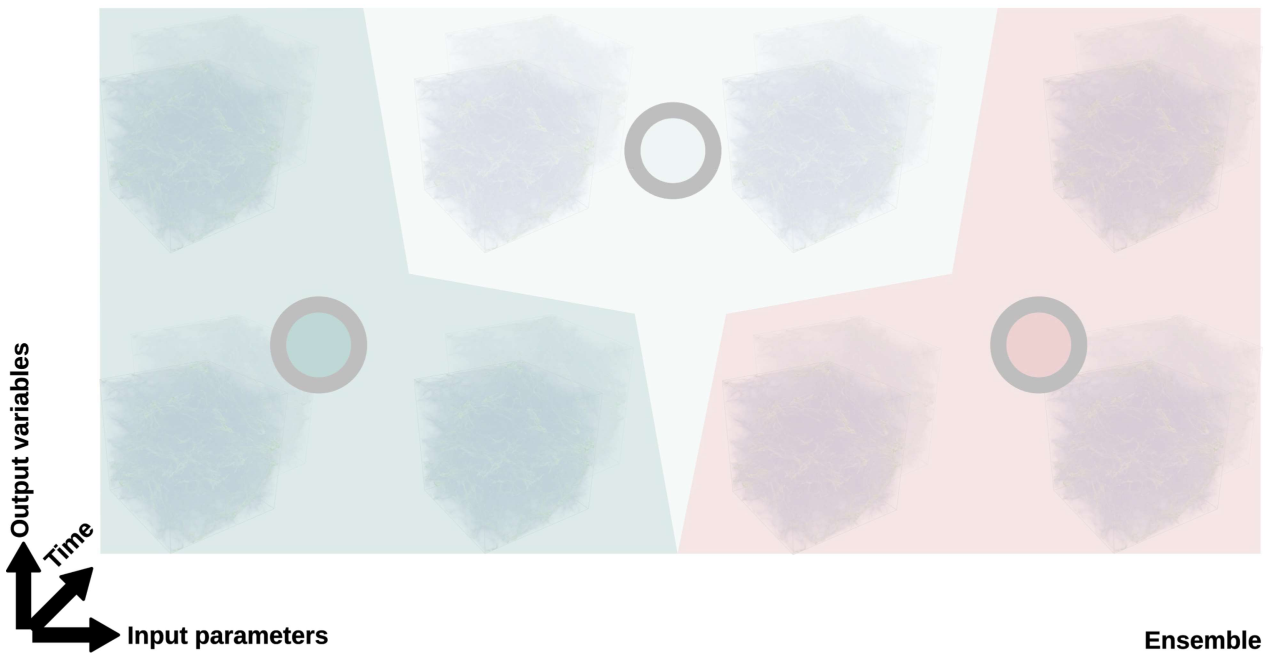


Topological data reduction



Topological data reduction

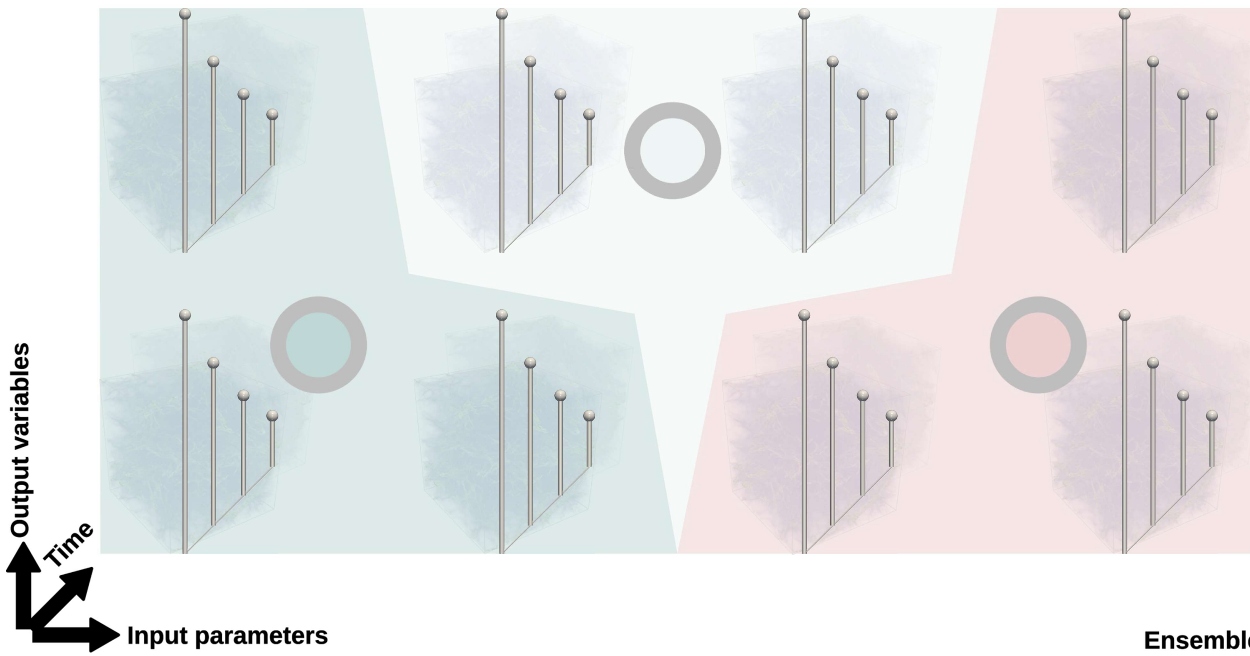
- **Challenges**
  - Significance
  - Data size



Topological data reduction

- **Challenges**
  - Significance
  - Data size

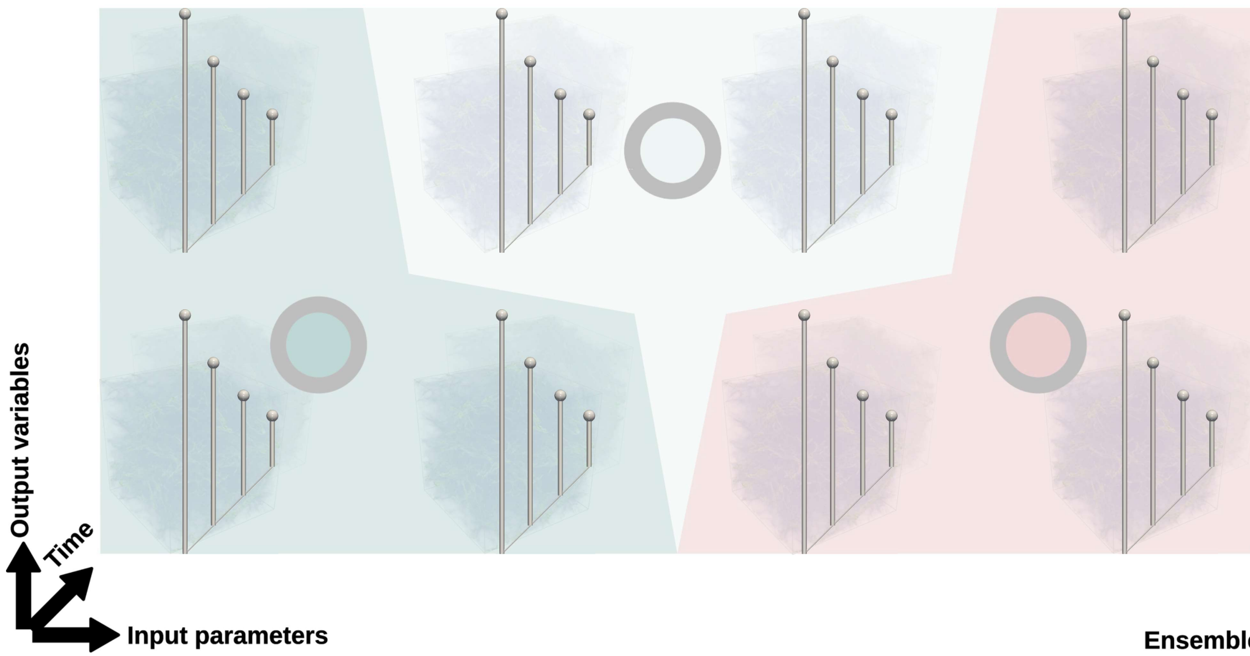
- **Strategy**
  - Topological reduction



Topological data reduction

- **Challenges**
  - Significance
  - Data size

- **Strategy**
  - Topological reduction



erc **http://erc-tori.github.io/**

Metric spaces for Persistence diagrams

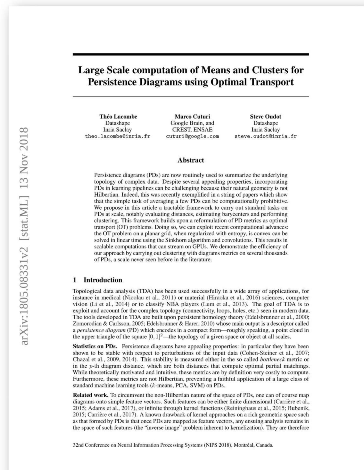
- Vast literature!


Stability of Persistence Diagrams
David Cohen-Steiner · Herbert Edelsbrunner · John Harer

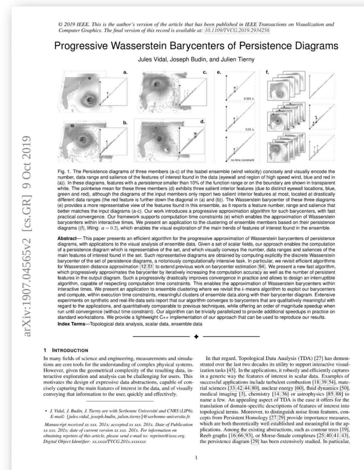
[Cohen-Steiner '05]


Fréchet Means for Distributions of Persistence Diagrams
Katharine Turner · Yuriy Mileyko · Sayan Mukherjee · John Harer

[Turner '14]


Large Scale computation of Means and Clusters for Persistence Diagrams using Optimal Transport
Théo Lacombe · Marco Cuturi · Steve Oudot

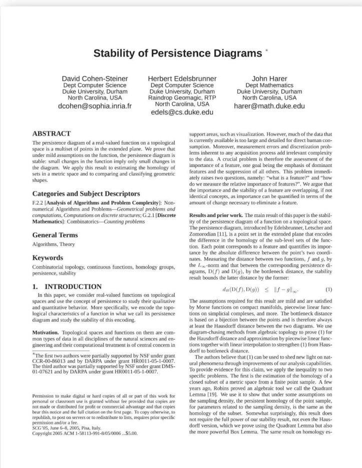
[Lacombe '18]


Progressive Wasserstein Barycenters of Persistence Diagrams
Jules Vidal, Joseph Budin, and Julien Tierny

[Vidal '19]

Metric spaces for Persistence diagrams

- Vast literature!



Stability of Persistence Diagrams

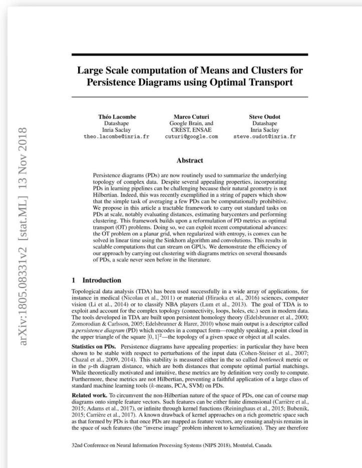
David Cohen-Steiner, Herbert Edelsbrunner, John Harer

[Cohen-Steiner '05]

Fréchet Means for Distributions of Persistence Diagrams

Katharine Turner · Yuriy Mileyko · Sayan Mukherjee · John Harer

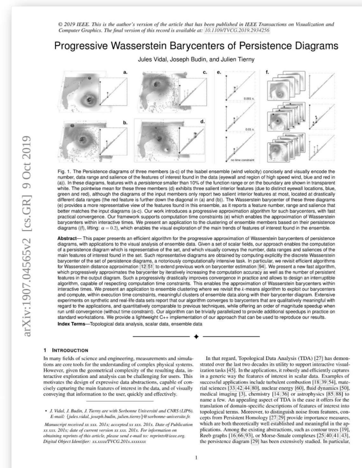
[Turner '14]



Large Scale computation of Means and Clusters for Persistence Diagrams using Optimal Transport

Théo Lacombe, Marco Cuturi, Steve Oudot

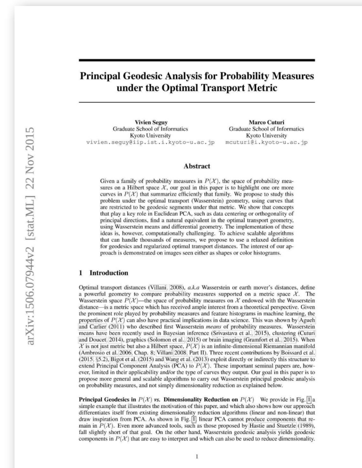
[Lacombe '18]



Progressive Wasserstein Barycenters of Persistence Diagrams

Jules Vidal, Joseph Budin, and Julien Tierny

[Vidal '19]



Principal Geodesic Analysis for Probability Measures under the Optimal Transport Metric

Vivien Seguy, Marco Cuturi

[Seguy '15]

Contributions

- **Principal Geodesic Analysis (PGA) of Merge Trees (and Persistence diagrams)**

Contributions

- **Principal Geodesic Analysis (PGA) of Merge Trees (and Persistence diagrams)**
  - Approach
  - Algorithm
  - Implementation
  - Applications

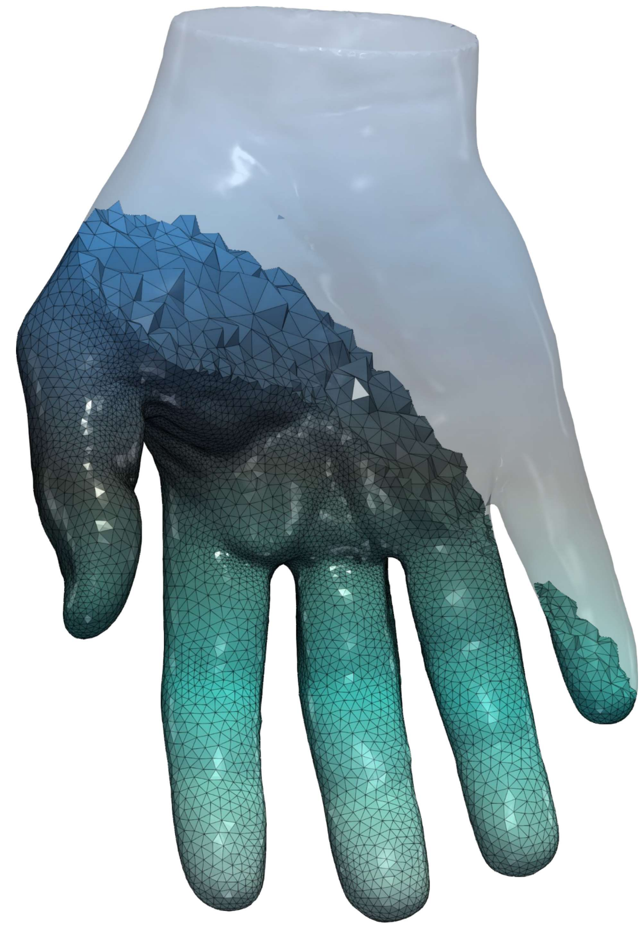
Background

Input data

- **Ensemble of scalar fields**
  - $f_i : \mathscr{M} \rightarrow \mathbb{R}$
  - $i \in \{1, \ldots, N\}$

Input data

- **Ensemble of scalar fields**
  - $f_i : \mathcal{M} \rightarrow \mathbb{R}$
  - $i \in \{1, \ldots, N\}$
  - $\mathcal{M}$: simplicial complex



Persistence diagrams (PDs)

- **Sublevel set filtration**
  - Topology changes at critical points
  - Elder rule [Edelsbrunner&Harer '10]

Persistence diagrams (PDs)

- **Sublevel set filtration**
  - Topology changes at critical points
  - Elder rule [Edelsbrunner&Harer '10]

Persistence diagrams (PDs)

- **Sublevel set filtration**
  - Topology changes at critical points
  - Elder rule [Edelsbrunner&Harer '10]

Persistence diagrams (PDs)

- **Sublevel set filtration**
  - Topology changes at critical points
  - Elder rule [Edelsbrunner&Harer '10]

Persistence diagrams (PDs)

- **Sublevel set filtration**
  - Topology changes at critical points
  - Elder rule [Edelsbrunner&Harer '10]

- **Persistence diagram**
  - $\mathscr{D}(f_i)$
  - Bar code in birth/death space

Merge trees

- **1D simplicial complex**
  - $\mathcal{T}^-(f_i) = \mathcal{M} / \sim$
  - $p_1 \sim p_2$

Merge trees

- **1D simplicial complex**
  - $\mathcal{T}^{-}(f_i) = \mathcal{M} / \sim$
  - $p_1 \sim p_2$
    - $f_i(p_1) = f_i(p_2)$
    - Same connected component of sub-level set

Merge trees

- **1D simplicial complex**
  - $\mathcal{T}^{-}(f_i) = \mathcal{M} / \sim$
  - $p_1 \sim p_2$
    - $f_i(p_1) = f_i(p_2)$
    - Same connected component of sub-level set

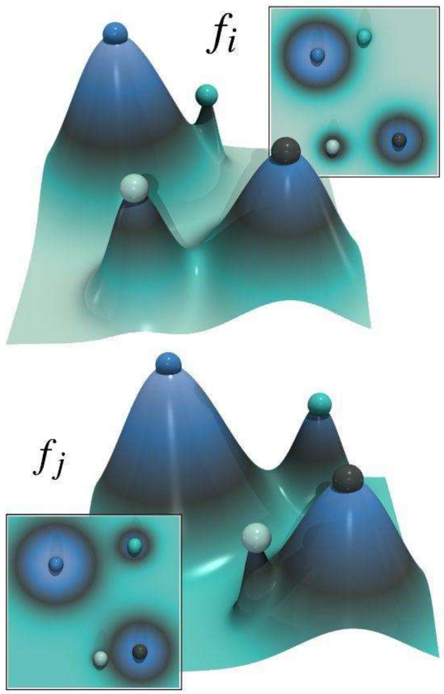
Merge trees

- **1D simplicial complex**
  - $\mathscr{T}^{-}(f_i) = \mathscr{M} / \sim$
  - $p_1 \sim p_2$
    - $f_i(p_1) = f_i(p_2)$
    - Same connected component of sub-level set

Merge trees

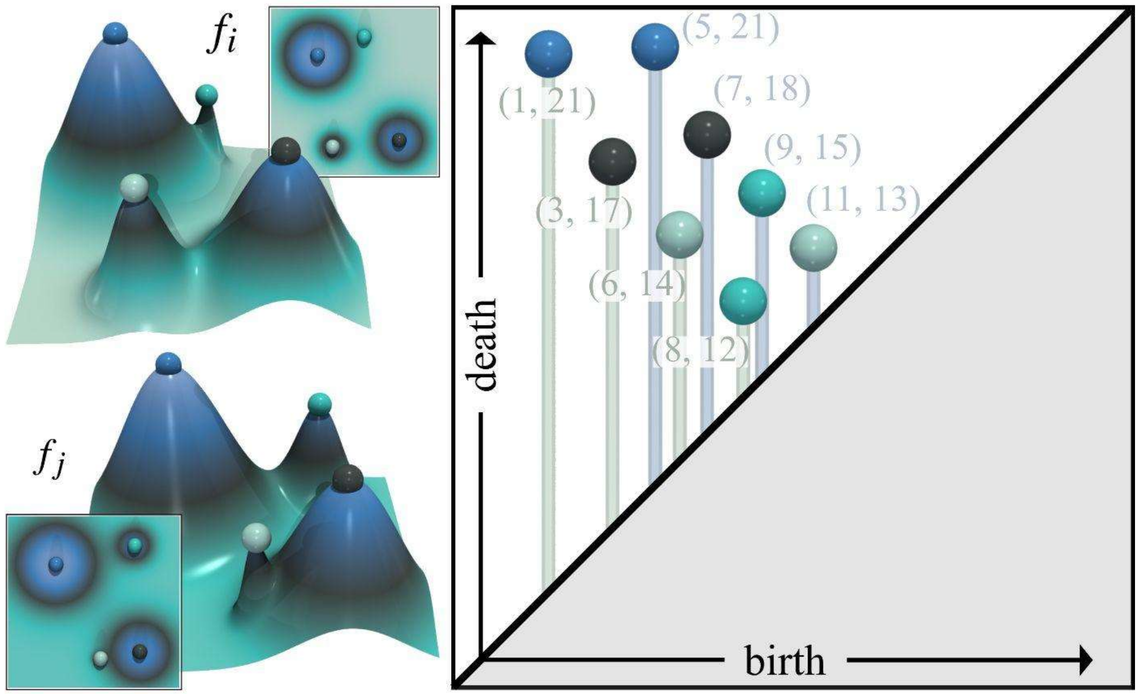
- **1D simplicial complex**
  - $\mathscr{T}^{-}(f_i) = \mathscr{M} / \sim$
  - $p_1 \sim p_2$
    - $f_i(p_1) = f_i(p_2)$
    - Same connected component of sub-level set

Wasserstein distance



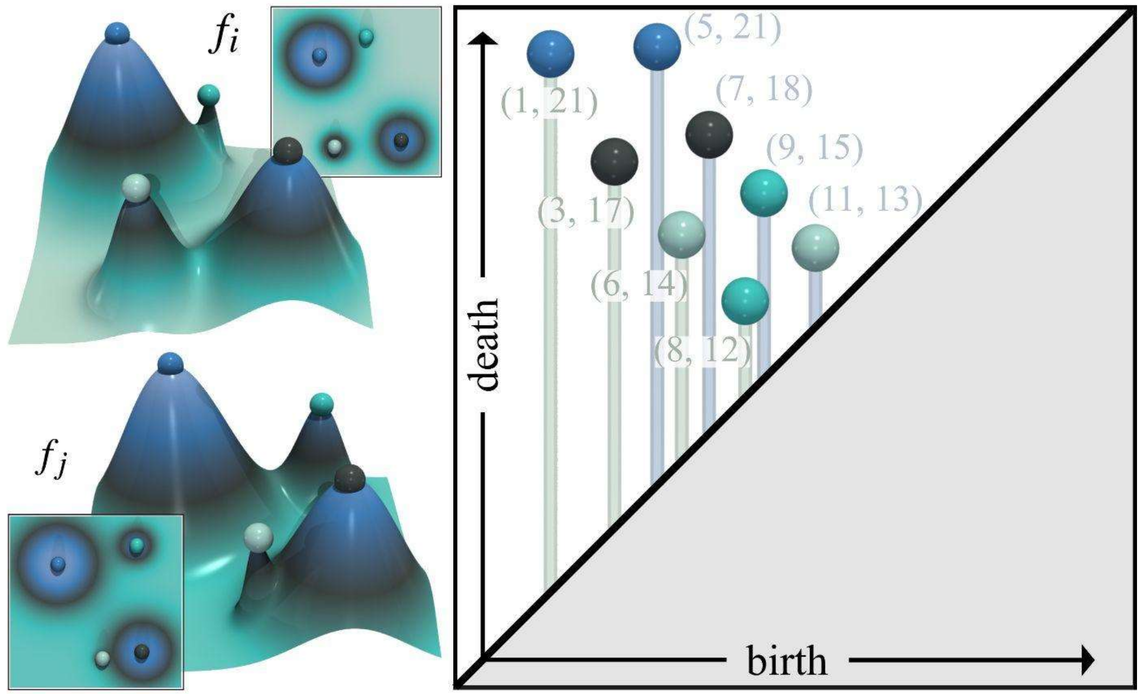
- Persistence diagrams

Wasserstein distance



- **Persistence diagrams**

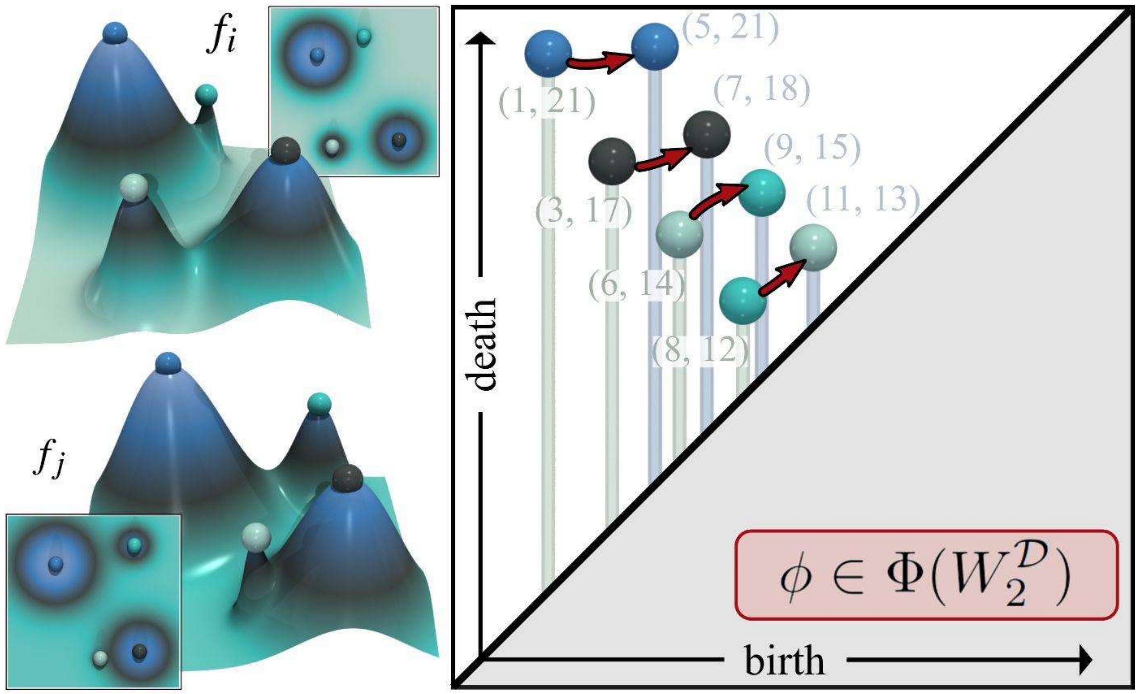
Wasserstein distance



- **Persistence diagrams**
  - Bipartite assignment

$$W_2^{\mathcal{D}}\big(\mathcal{D}(f_i), \mathcal{D}(f_j)\big) = \min_{\phi \in \Phi} \Big(\sum_{p_i \in \mathcal{D}(f_i)} d_2\big(p_i, \phi(p_i)\big)^2 \Big)^{1/2}$$

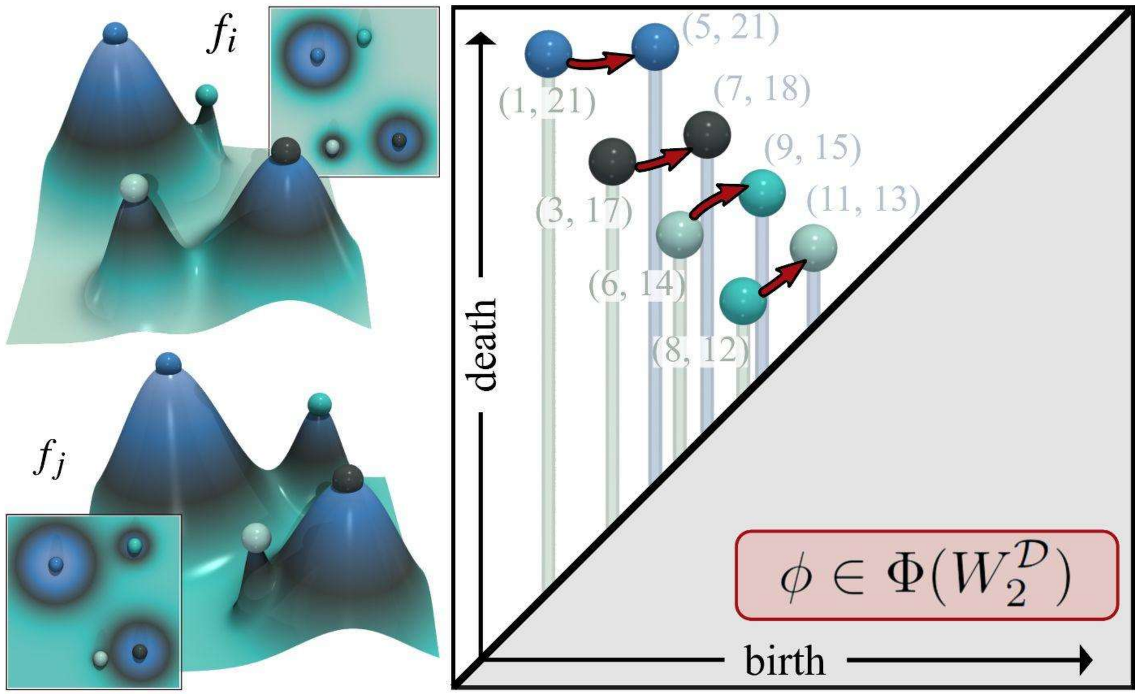
Wasserstein distance



- **Persistence diagrams**
  - Bipartite assignment

$$W_2^{\mathcal{D}}\big(\mathcal{D}(f_i), \mathcal{D}(f_j)\big) = \min_{\phi \in \Phi} \Big(\sum_{p_i \in \mathcal{D}(f_i)} d_2\big(p_i, \phi(p_i)\big)^2 \Big)^{1/2}$$

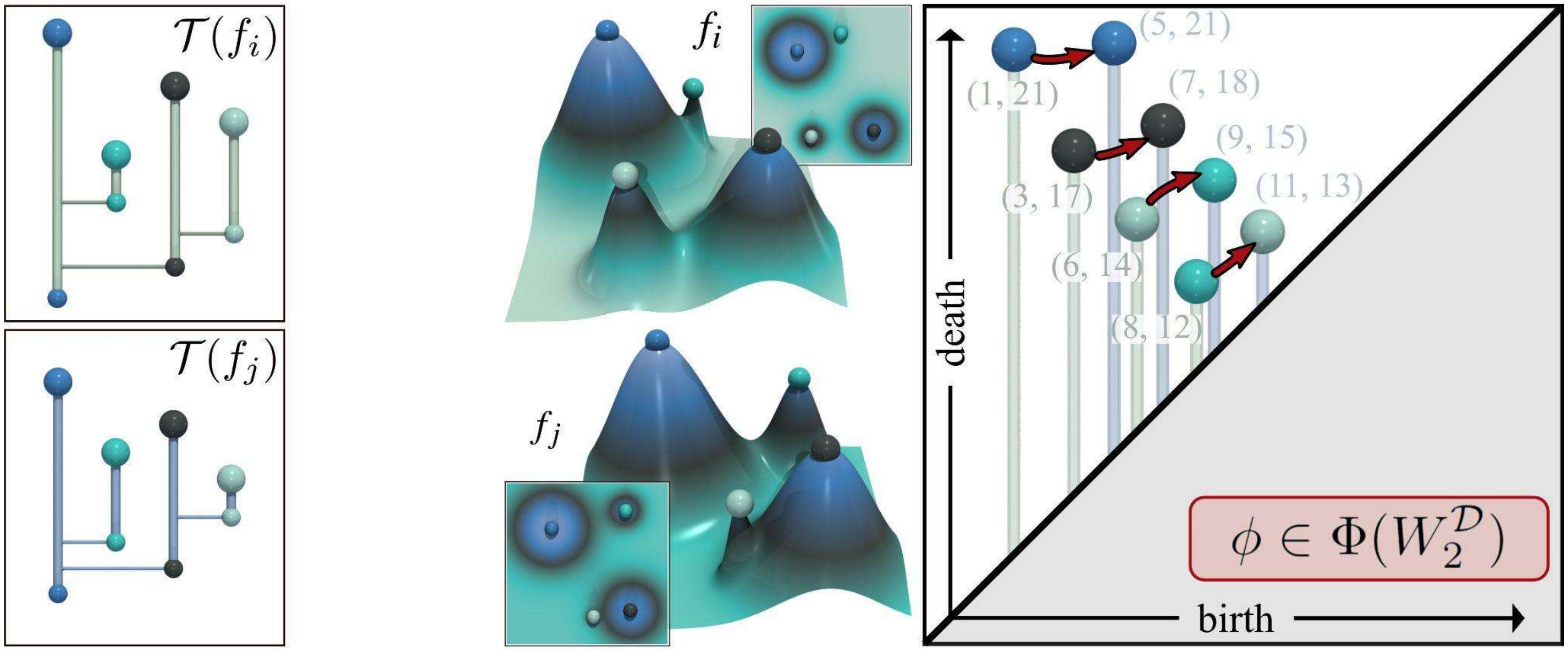
Wasserstein distance



- **Persistence diagrams**
  - Bipartite assignment
  - Features treated independently

$$W_2^{\mathcal{D}}\big(\mathcal{D}(f_i), \mathcal{D}(f_j)\big) = \min_{\phi \in \Phi} \Big(\sum_{p_i \in \mathcal{D}(f_i)} d_2\big(p_i, \phi(p_i)\big)^2 \Big)^{1/2}$$

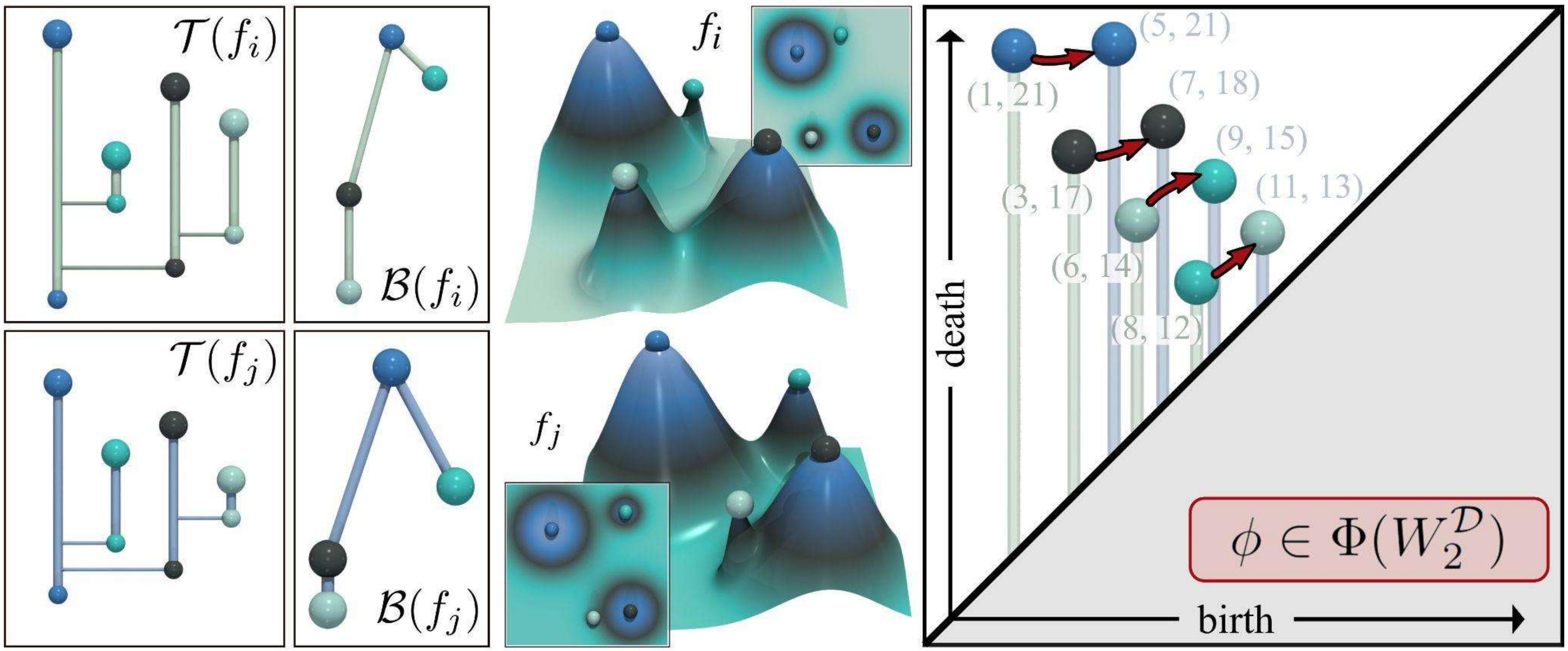
Wasserstein distance



- **Merge trees** [Pont '21]

$$W_2^{\mathcal{D}}\big(\mathcal{D}(f_i), \mathcal{D}(f_j)\big) = \min_{\phi \in \Phi} \Big(\sum_{p_i \in \mathcal{D}(f_i)} d_2\big(p_i, \phi(p_i)\big)^2 \Big)^{1/2}$$

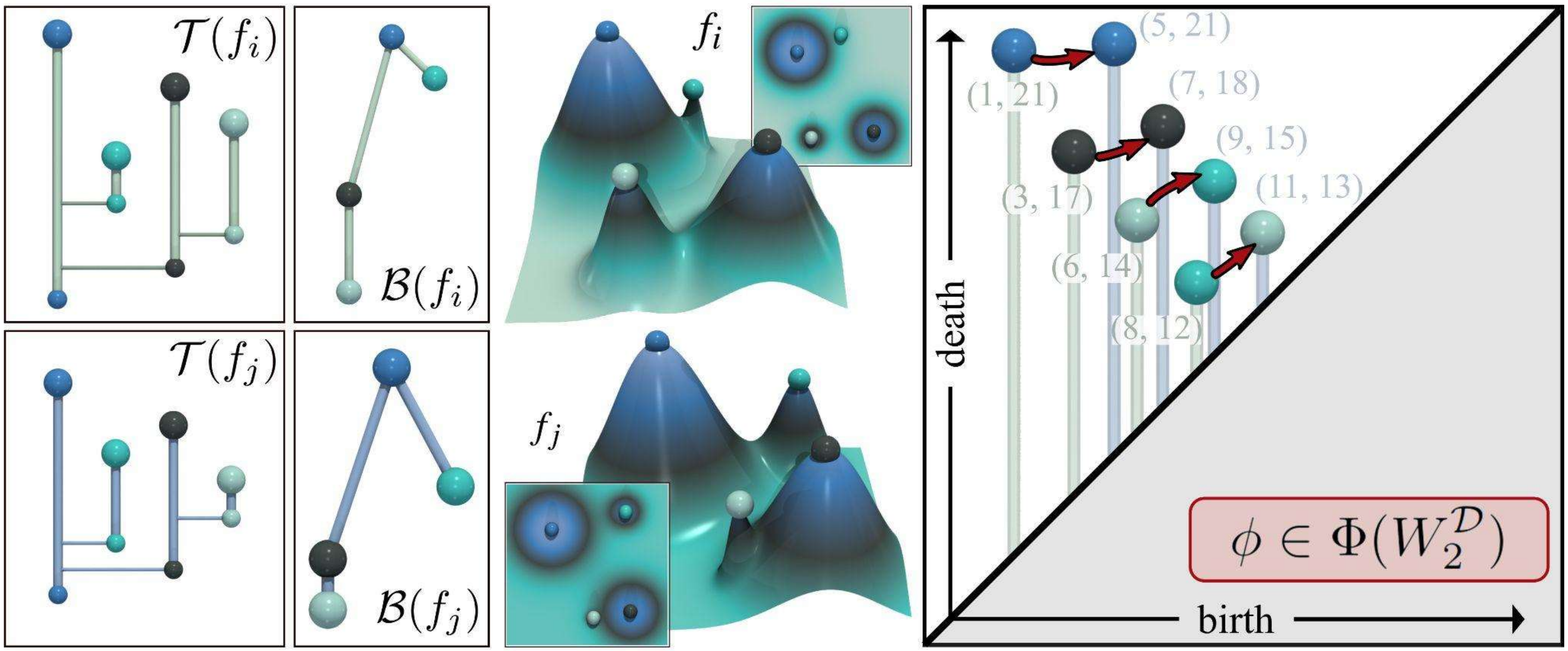
Wasserstein distance



- **Merge trees** [Pont '21]

$$W_2^{\mathcal{D}}\big(\mathcal{D}(f_i), \mathcal{D}(f_j)\big) = \min_{\phi \in \Phi} \Big(\sum_{p_i \in \mathcal{D}(f_i)} d_2\big(p_i, \phi(p_i)\big)^2 \Big)^{1/2}$$

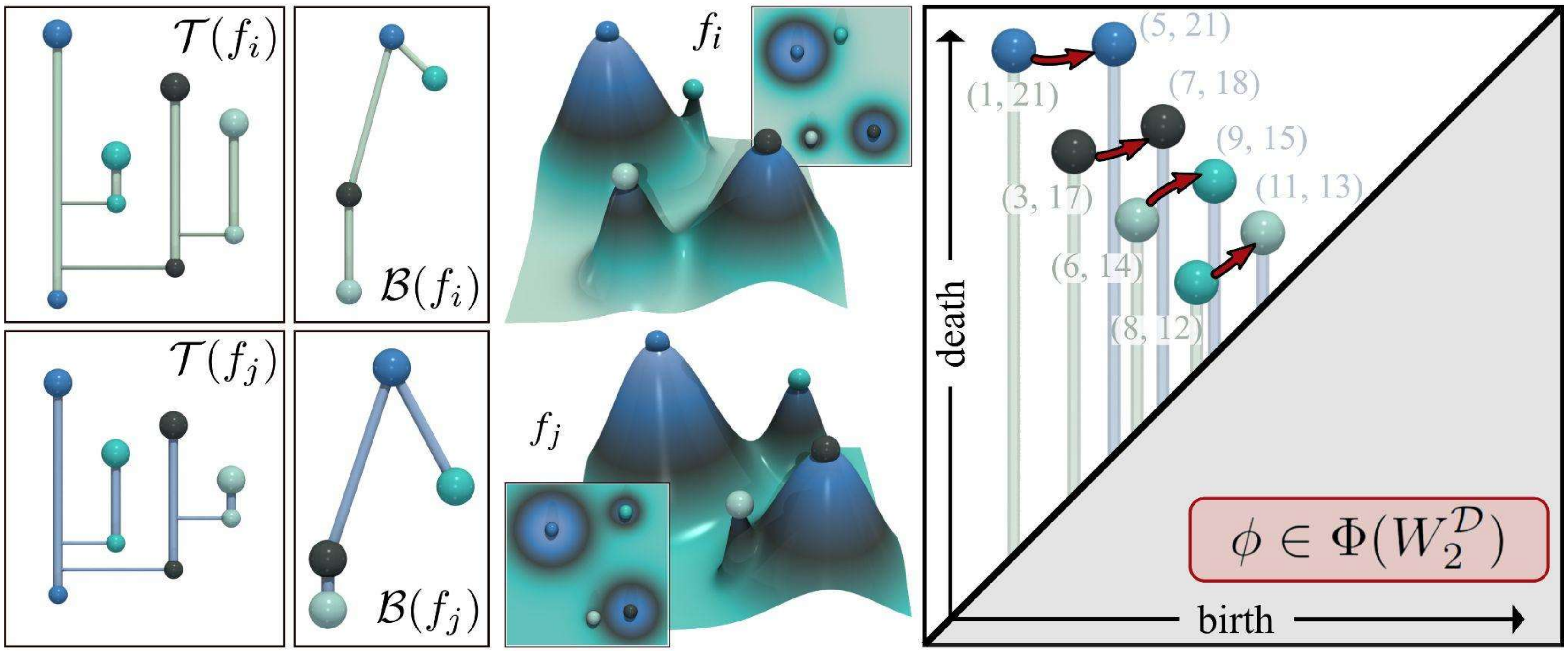
Wasserstein distance



- **Merge trees** [Pont '21]
  - Partial (rooted) isomorphisms

$$W_2^{\mathcal{D}}\big(\mathcal{D}(f_i), \mathcal{D}(f_j)\big) = \min_{\phi \in \Phi} \Big(\sum_{p_i \in \mathcal{D}(f_i)} d_2\big(p_i, \phi(p_i)\big)^2 \Big)^{1/2}$$

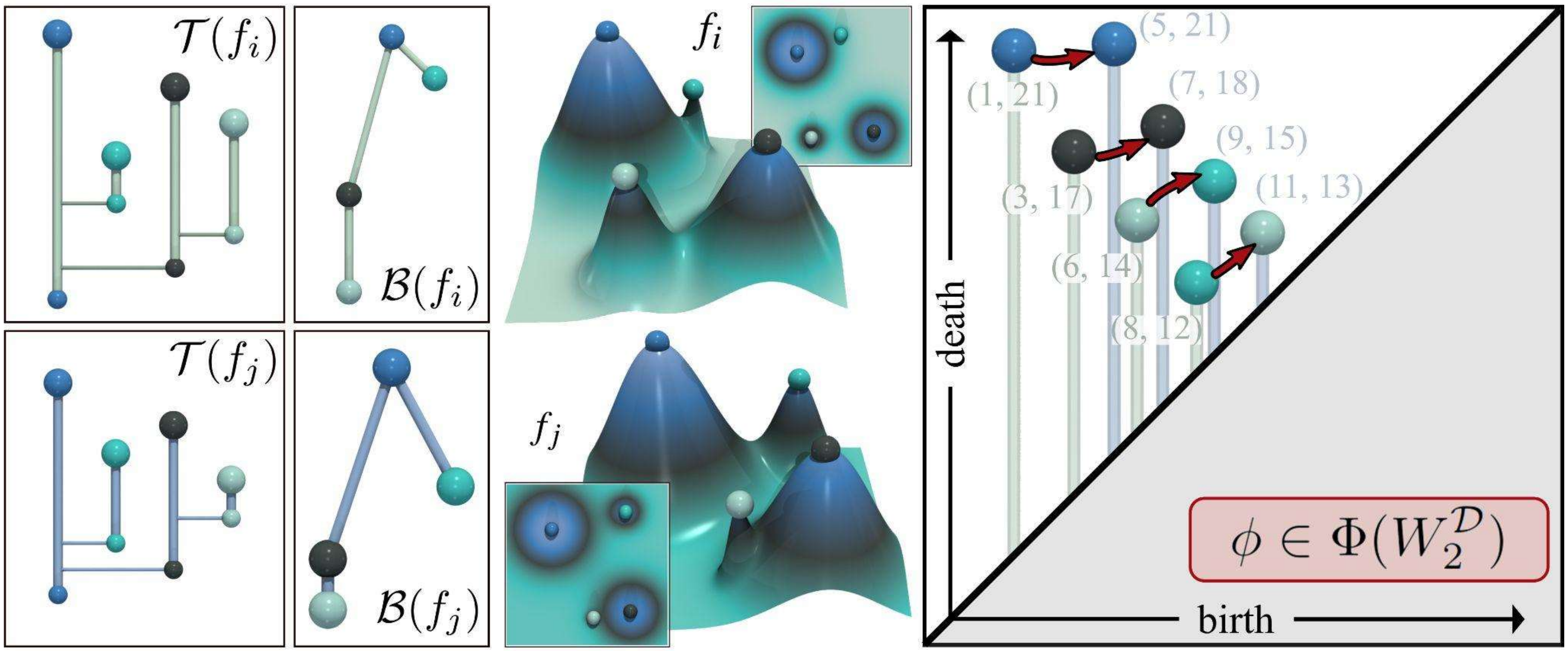
Wasserstein distance



- **Merge trees** [Pont '21]
  - Partial (rooted) isomorphisms

$$W_2^{\mathcal{T}}\left(\mathcal{B}(f_i), \mathcal{B}(f_j)\right) = \min_{\phi' \in \Phi'} \left(\sum_{p_i \in \mathcal{B}(f_i)} d_2\left(p_i, \phi'(p_i)\right)^2 \right)^{1/2}$$

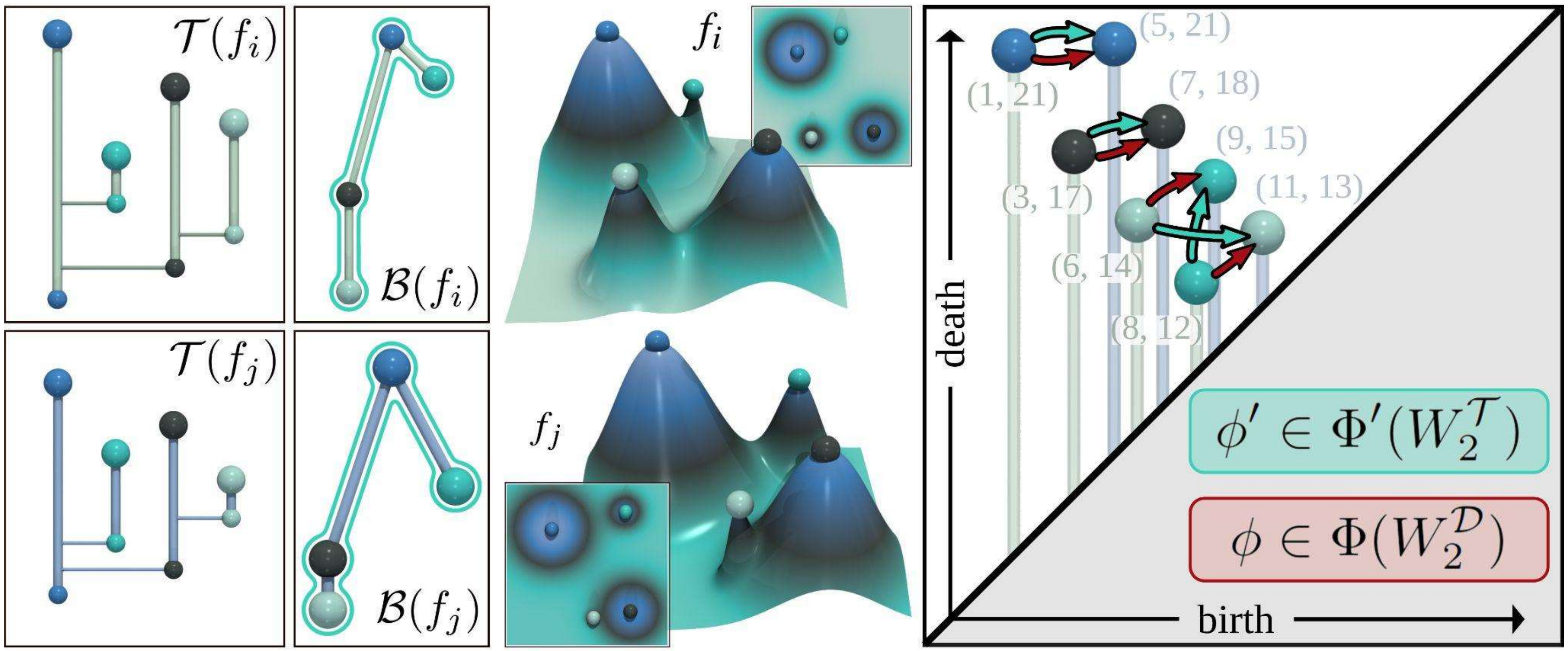
Wasserstein distance



- **Merge trees** [Pont '21]
  - Partial (rooted) isomorphisms

$$W_2^{\mathcal{T}}\left(\mathcal{B}(f_i), \mathcal{B}(f_j)\right) = \min_{\phi' \in \Phi'} \left(\sum_{p_i \in \mathcal{B}(f_i)} d_2\left(p_i, \phi'(p_i)\right)^2 \right)^{1/2}$$

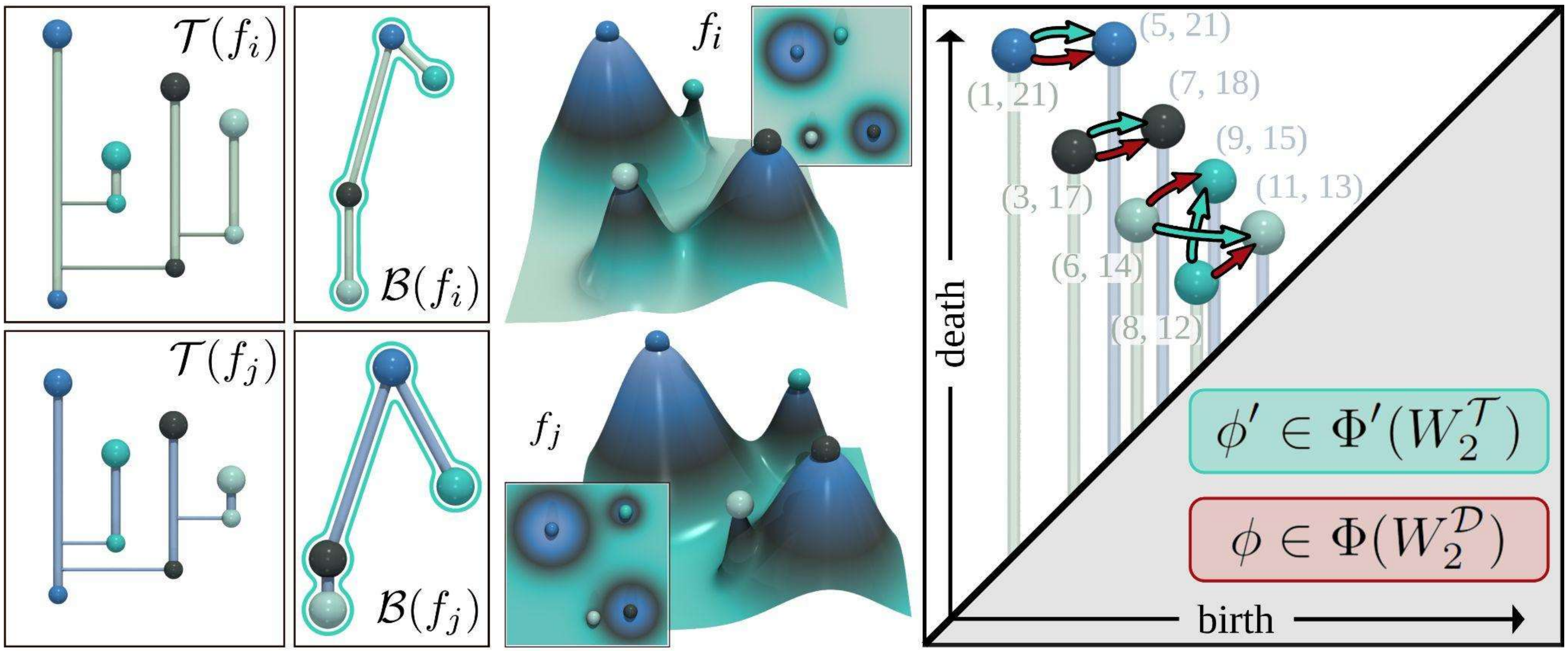
Wasserstein distance



- **Merge trees** [Pont '21]
  - Partial (rooted) isomorphisms

$$W_2^{\mathcal{T}}\left(\mathcal{B}(f_i), \mathcal{B}(f_j)\right) = \min_{\phi' \in \Phi'} \left(\sum_{p_i \in \mathcal{B}(f_i)} d_2\left(p_i, \phi'(p_i)\right)^2 \right)^{1/2}$$

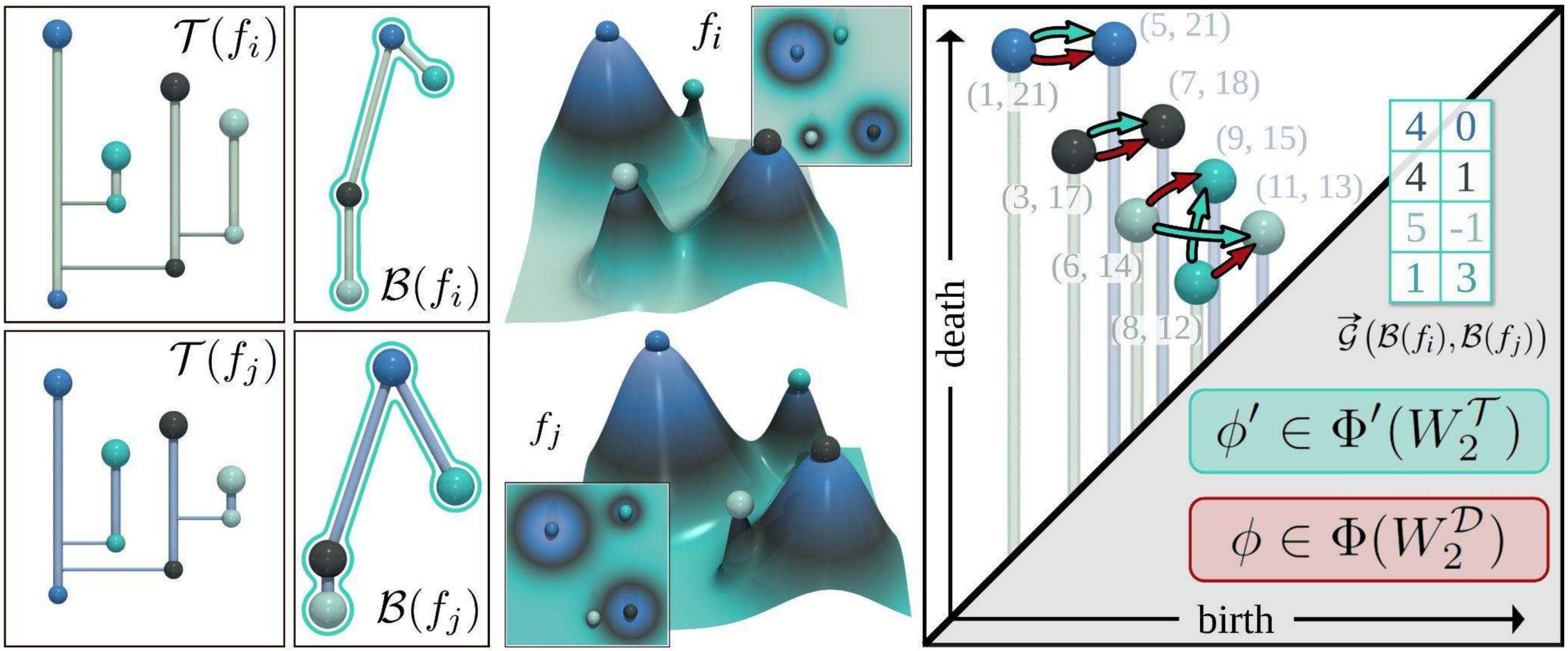
Wasserstein distance



- **Merge trees** [Pont '21]
  - Partial (rooted) isomorphisms
  - Capture global structure

$$W_2^{\mathcal{T}}\left(\mathcal{B}(f_i), \mathcal{B}(f_j)\right) = \min_{\phi' \in \Phi'} \left(\sum_{p_i \in \mathcal{B}(f_i)} d_2\left(p_i, \phi'(p_i)\right)^2 \right)^{1/2}$$

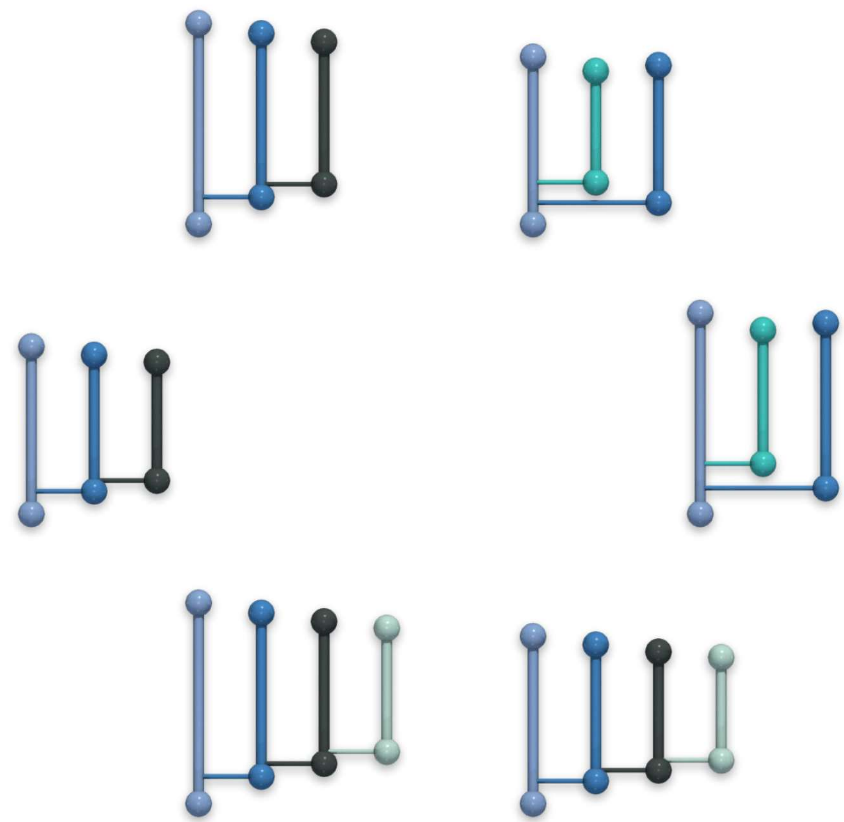
Wasserstein distance



- **Merge trees** [Pont '21]
  - Partial (rooted) isomorphisms
  - Capture global structure
  - **Geodesic**: vector in $\mathbb{R}^{2\times|\mathcal{B}(f_i)|}$

$$W_2^{\mathcal{T}}\left(\mathcal{B}(f_i), \mathcal{B}(f_j)\right) = \min_{\phi' \in \Phi'} \left(\sum_{p_i \in \mathcal{B}(f_i)} d_2\left(p_i, \phi'(p_i)\right)^2 \right)^{1/2}$$

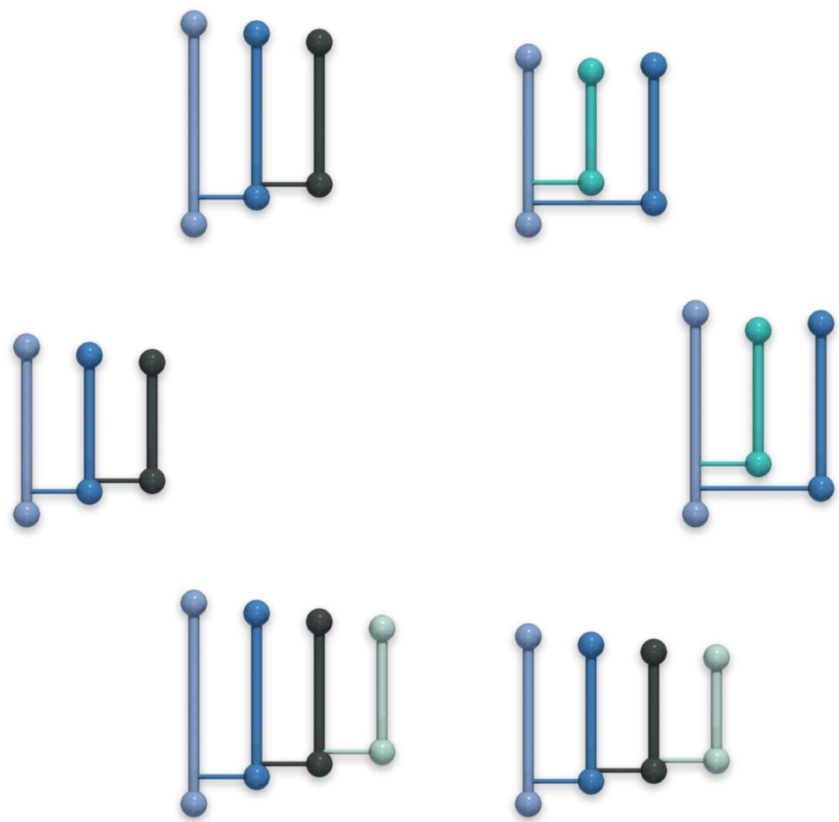
Wasserstein barycenters



Wasserstein barycenters

- **Fréchet energy**

$$E_F(\mathcal{B}) = \sum_{i=1}^{N} W_2^{\mathcal{T}}\left(\mathcal{B}, \mathcal{B}(f_i)\right)^2$$

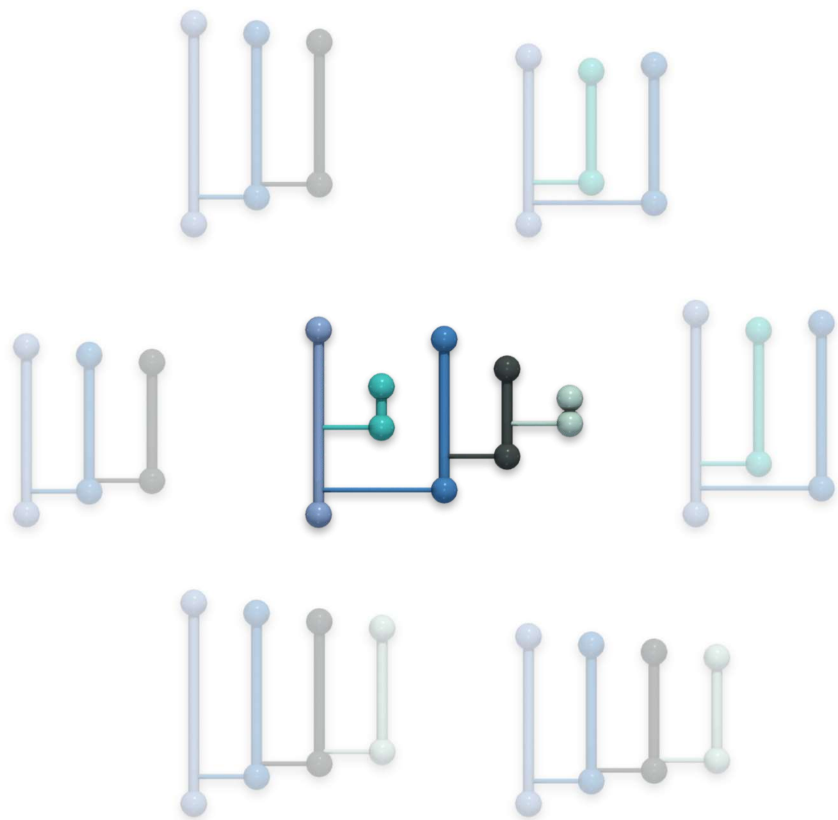


Wasserstein barycenters

- **Fréchet energy**

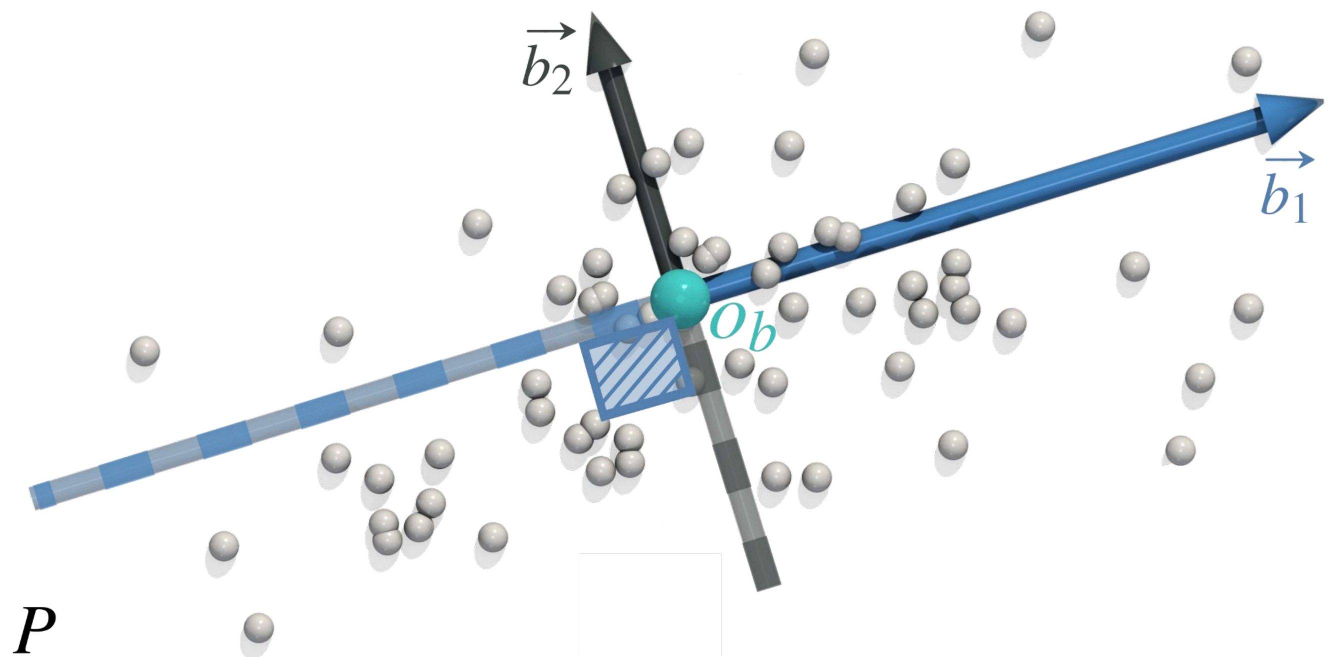
$$E_F(\mathcal{B}) = \sum_{i=1}^{N} W_2^{\mathcal{T}}\left(\mathcal{B}, \mathcal{B}(f_i)\right)^2$$

- **Algorithm** [Pont' 21]
  - Iterative Assignment/Update



Approach

Geometric interpretation of PCA



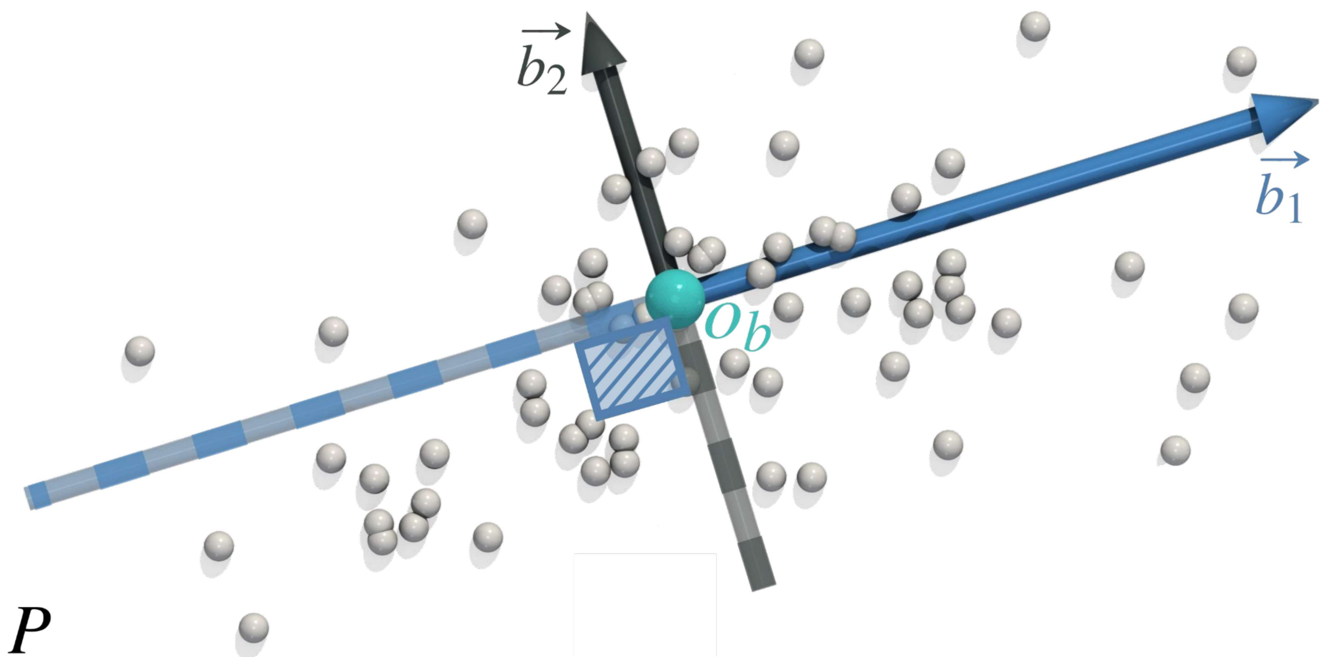
- **Input**

$P = \{p_1, p_2, \ldots, p_N\}$ in $\mathbb{R}^d$

- **PCA**

$B_{\mathbb{R}^d} = \{\overrightarrow{b_1}, \overrightarrow{b_2}, \ldots, \overrightarrow{b_d}\}$

Geometric interpretation of PCA



- **Input**

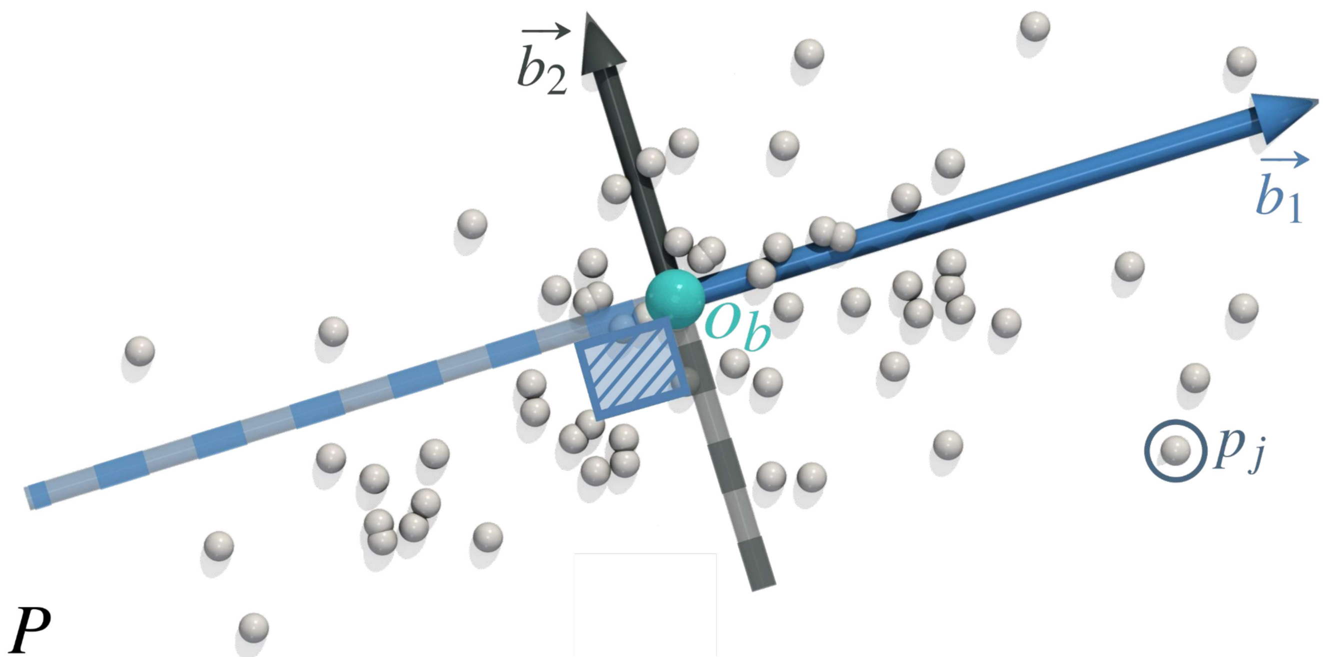
$P = \{p_1, p_2, \ldots, p_N\}$ in $\mathbb{R}^d$

- **PCA**

$B_{\mathbb{R}^d} = \{\vec{b_1}, \vec{b_2}, \ldots, \vec{b_d}\}$

$$E_{L_2}(B_{\mathbb{R}^d}) = \sum_{j=1}^{N} ||p_j - (o_b + \sum_{i=1}^{d'} \alpha_i^j \vec{b_i})||_2^2$$

Geometric interpretation of PCA



- **Input**

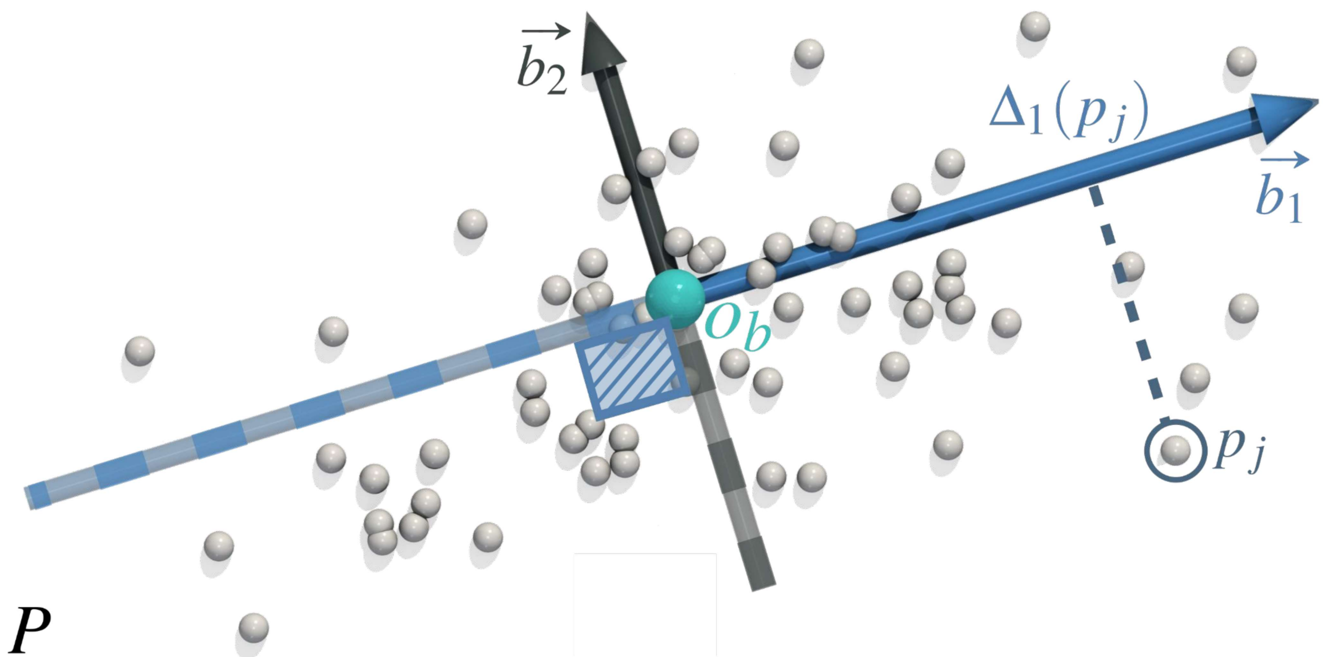
$P = \{p_1, p_2, \ldots, p_N\}$ in $\mathbb{R}^d$

- **PCA**

$B_{\mathbb{R}^d} = \{\vec{b_1}, \vec{b_2}, \ldots, \vec{b_d}\}$

$$E_{L_2}(B_{\mathbb{R}^d}) = \sum_{j=1}^{N} ||p_j - (o_b + \sum_{i=1}^{d'} \alpha_i^j \vec{b_i})||_2^2$$

Geometric interpretation of PCA



- **Input**

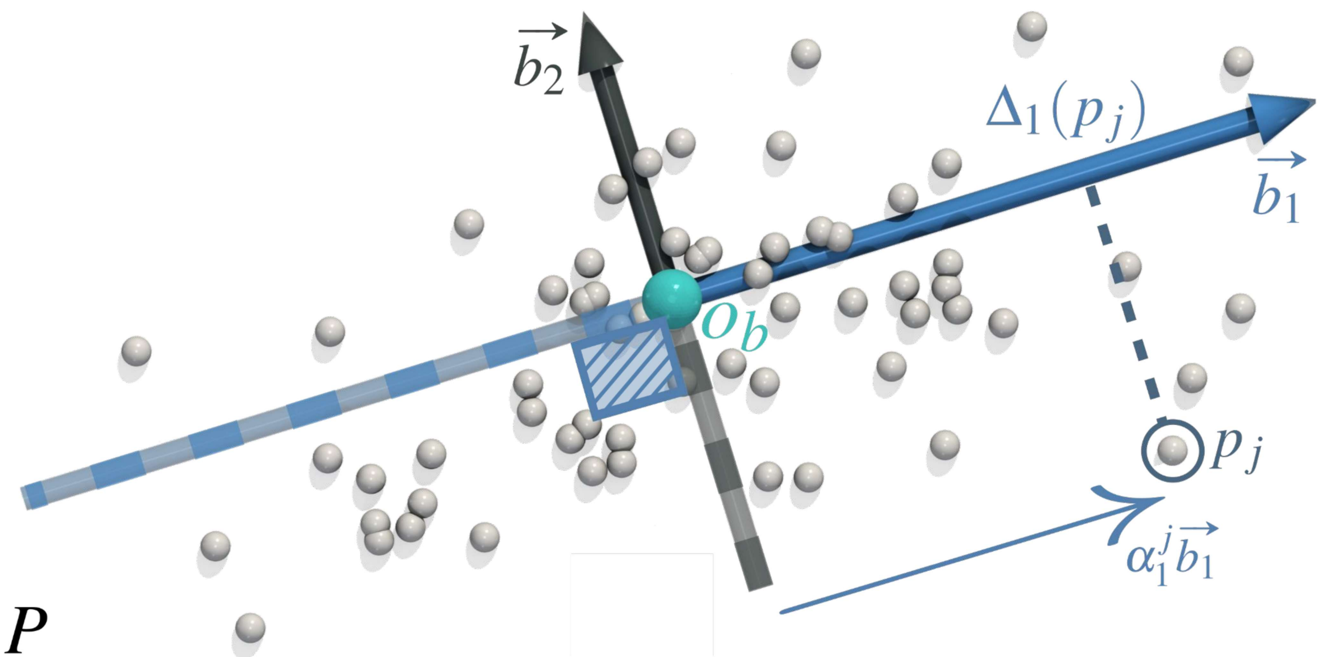
  $P = \{p_1, p_2, \ldots, p_N\}$ in $\mathbb{R}^d$

- **PCA**

  $B_{\mathbb{R}^d} = \{\overrightarrow{b_1}, \overrightarrow{b_2}, \ldots, \overrightarrow{b_d}\}$

$$E_{L_2}(B_{\mathbb{R}^d}) = \sum_{j=1}^{N} ||p_j - (o_b + \sum_{i=1}^{d'} \alpha_i^j \overrightarrow{b_i})||_2^2$$

Geometric interpretation of PCA



- **Input**

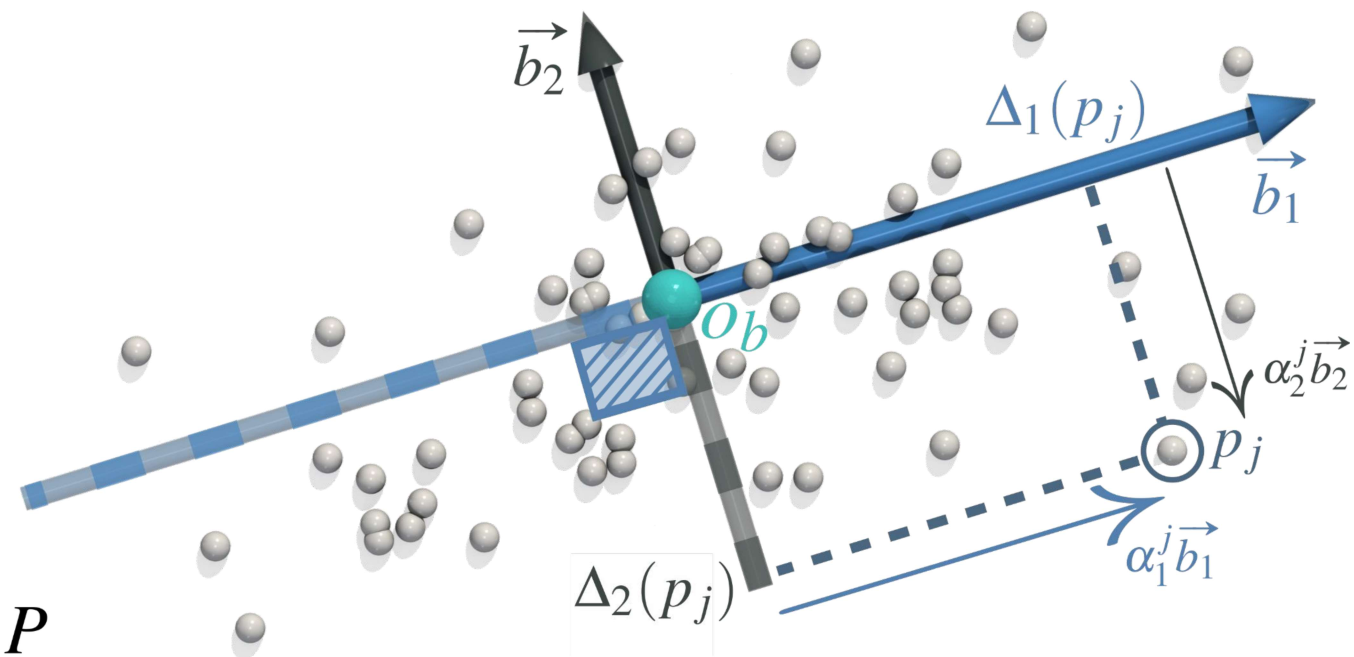
$P = \{p_1, p_2, \ldots, p_N\}$ in $\mathbb{R}^d$

- **PCA**

$B_{\mathbb{R}^d} = \{\vec{b_1}, \vec{b_2}, \ldots, \vec{b_d}\}$

$$E_{L_2}(B_{\mathbb{R}^d}) = \sum_{j=1}^{N} ||p_j - (o_b + \sum_{i=1}^{d'} \alpha_i^j \vec{b_i})||_2^2$$

Geometric interpretation of PCA



- **Input**

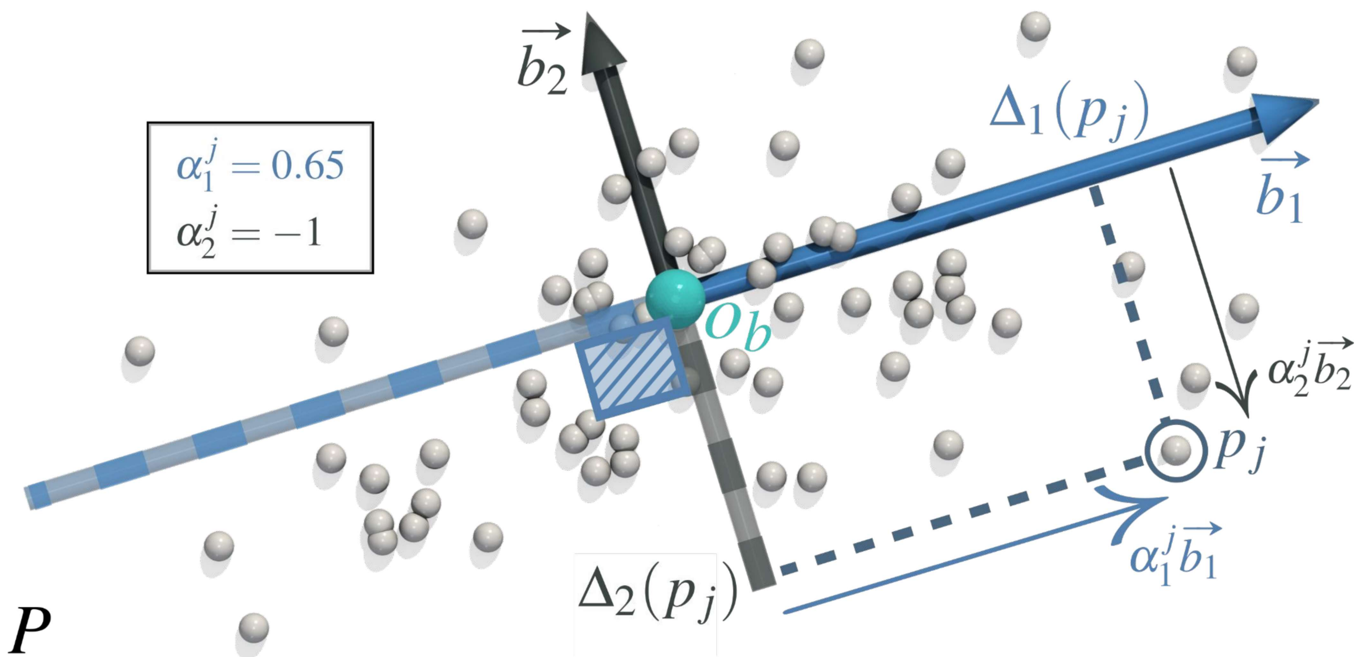
$P = \{p_1, p_2, \ldots, p_N\}$ in $\mathbb{R}^d$

- **PCA**

$B_{\mathbb{R}^d} = \{\overrightarrow{b_1}, \overrightarrow{b_2}, \ldots, \overrightarrow{b_d}\}$

$$E_{L_2}(B_{\mathbb{R}^d}) = \sum_{j=1}^{N} ||p_j - (o_b + \sum_{i=1}^{d'} \alpha_i^j \overrightarrow{b_i})||_2^2$$

Geometric interpretation of PCA



- **Input**

$P = \{p_1, p_2, \ldots, p_N\}$ in $\mathbb{R}^d$

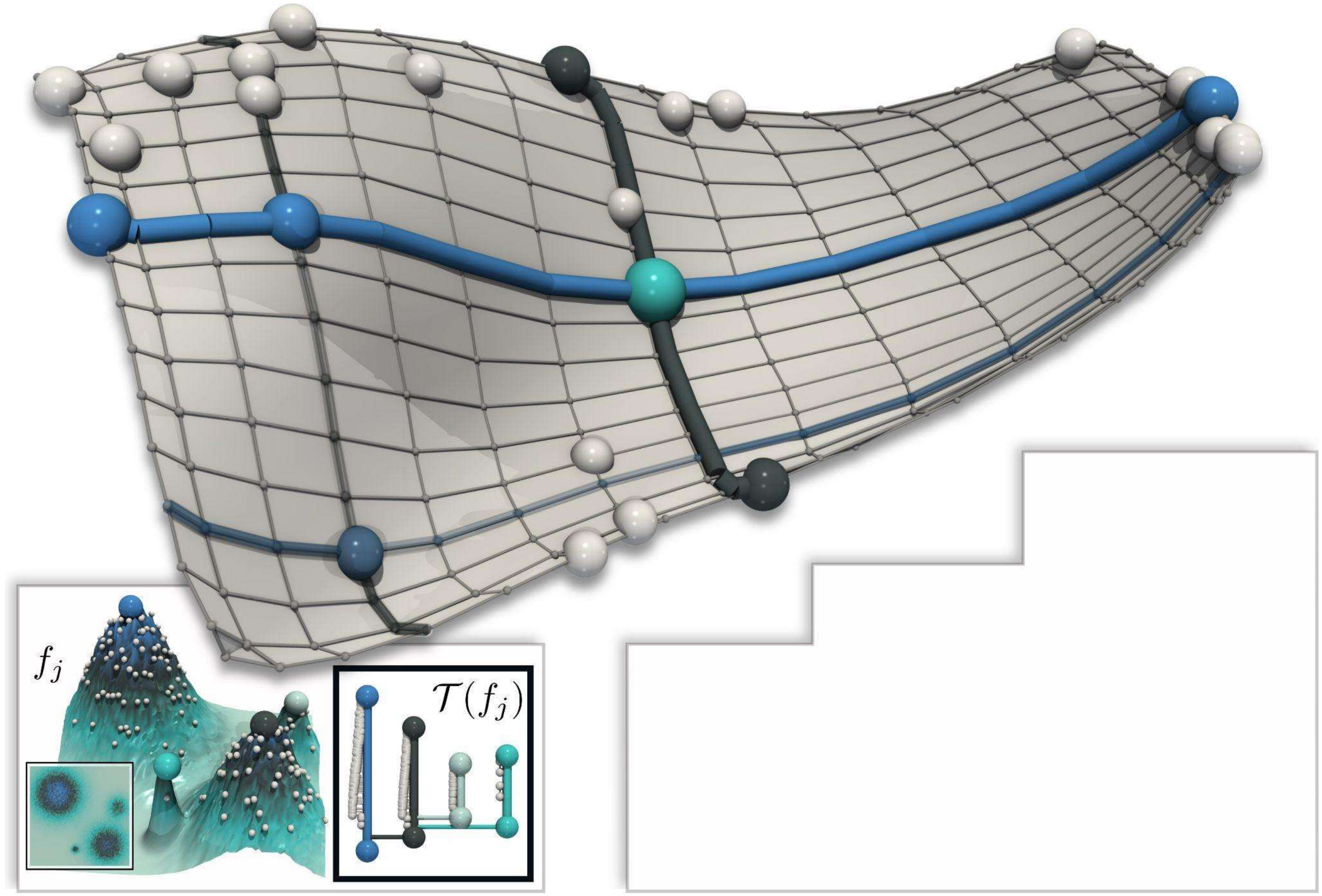
- **PCA**

$B_{\mathbb{R}^d} = \{\overrightarrow{b_1}, \overrightarrow{b_2}, \ldots, \overrightarrow{b_d}\}$

$$E_{L_2}(B_{\mathbb{R}^d}) = \sum_{j=1}^{N} ||p_j - (o_b + \sum_{i=1}^{d'} \alpha_i^j \overrightarrow{b_i})||_2^2$$

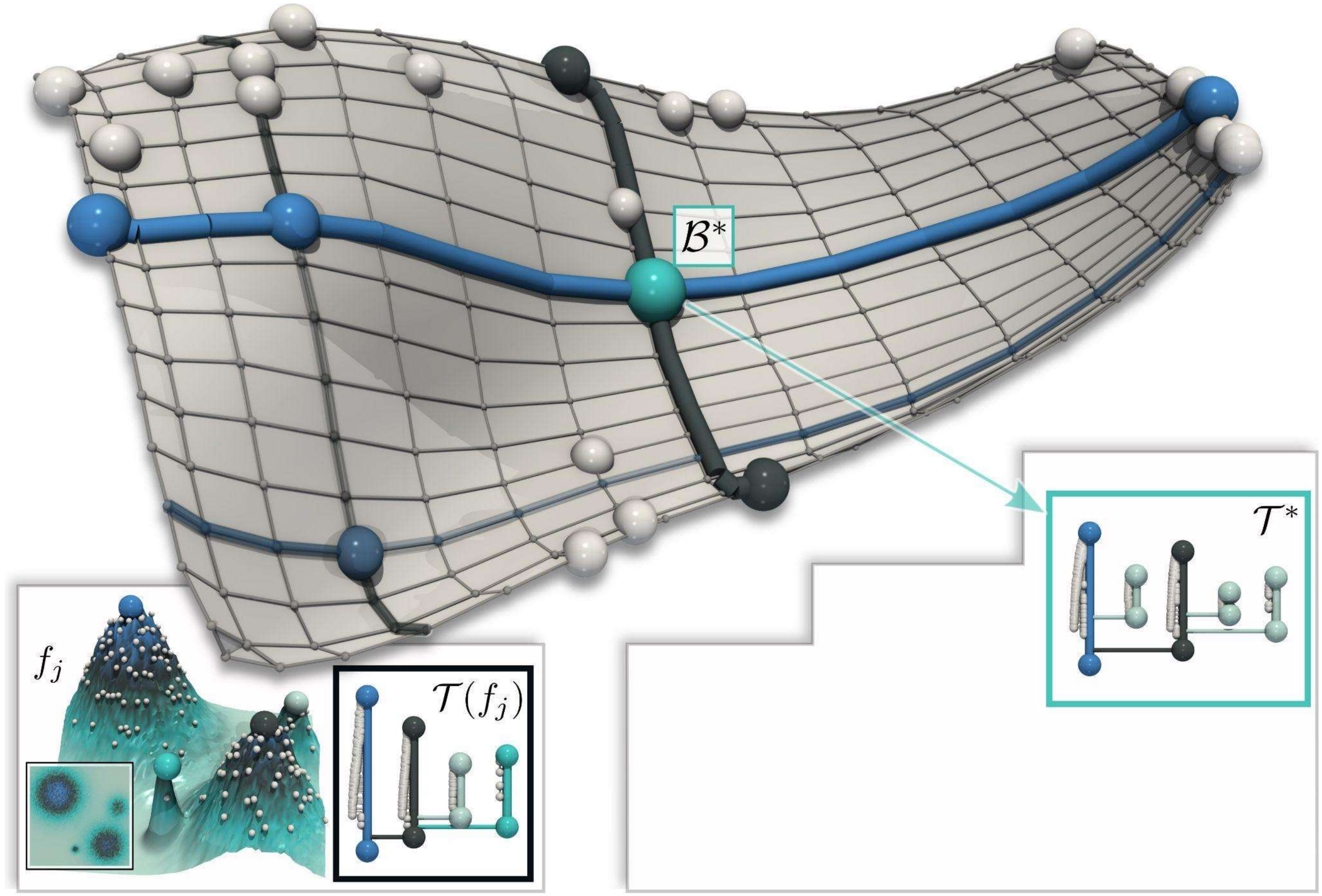
Overview

- **BDT Basis**



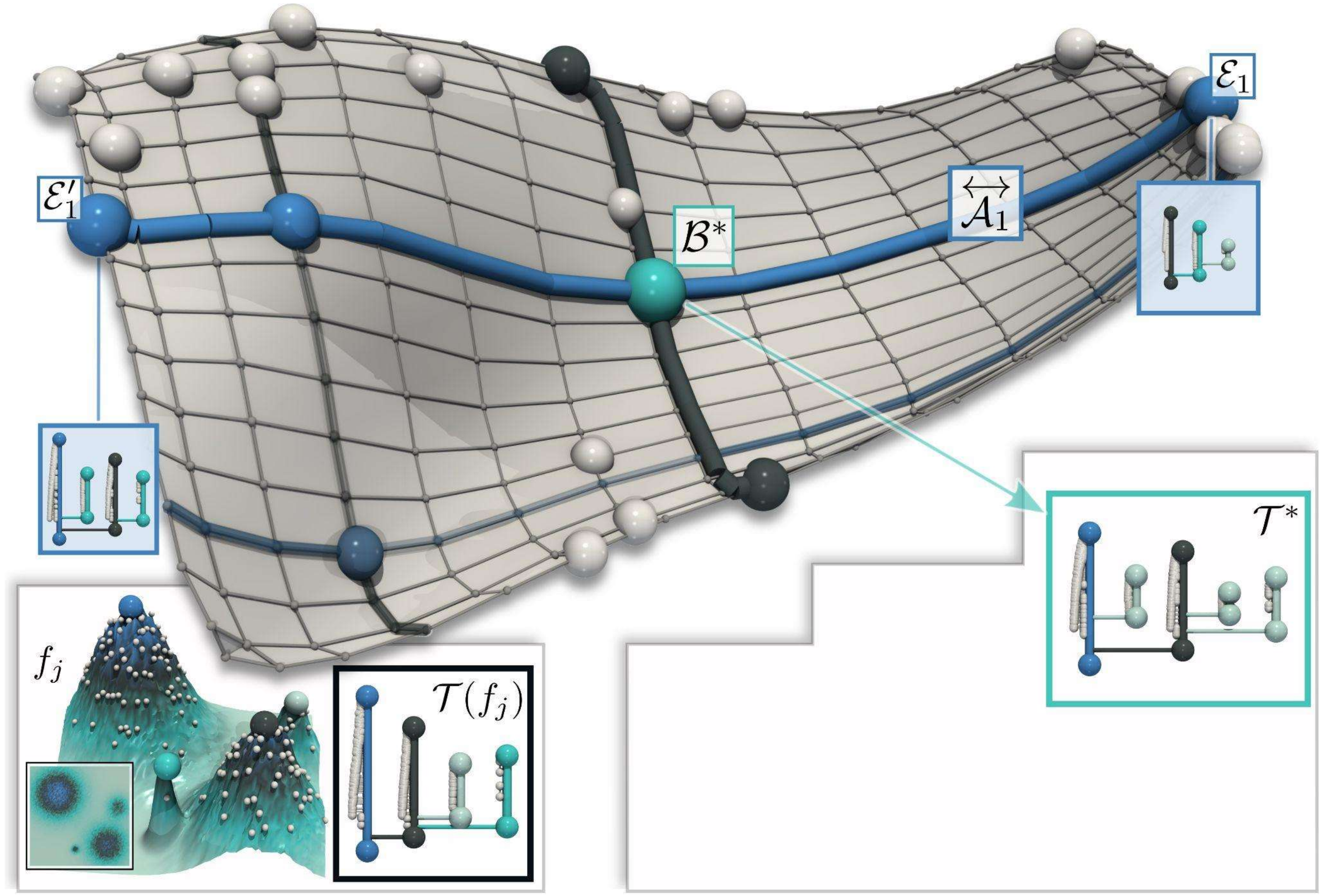
Overview

- **BDT Basis**
  - Barycenter



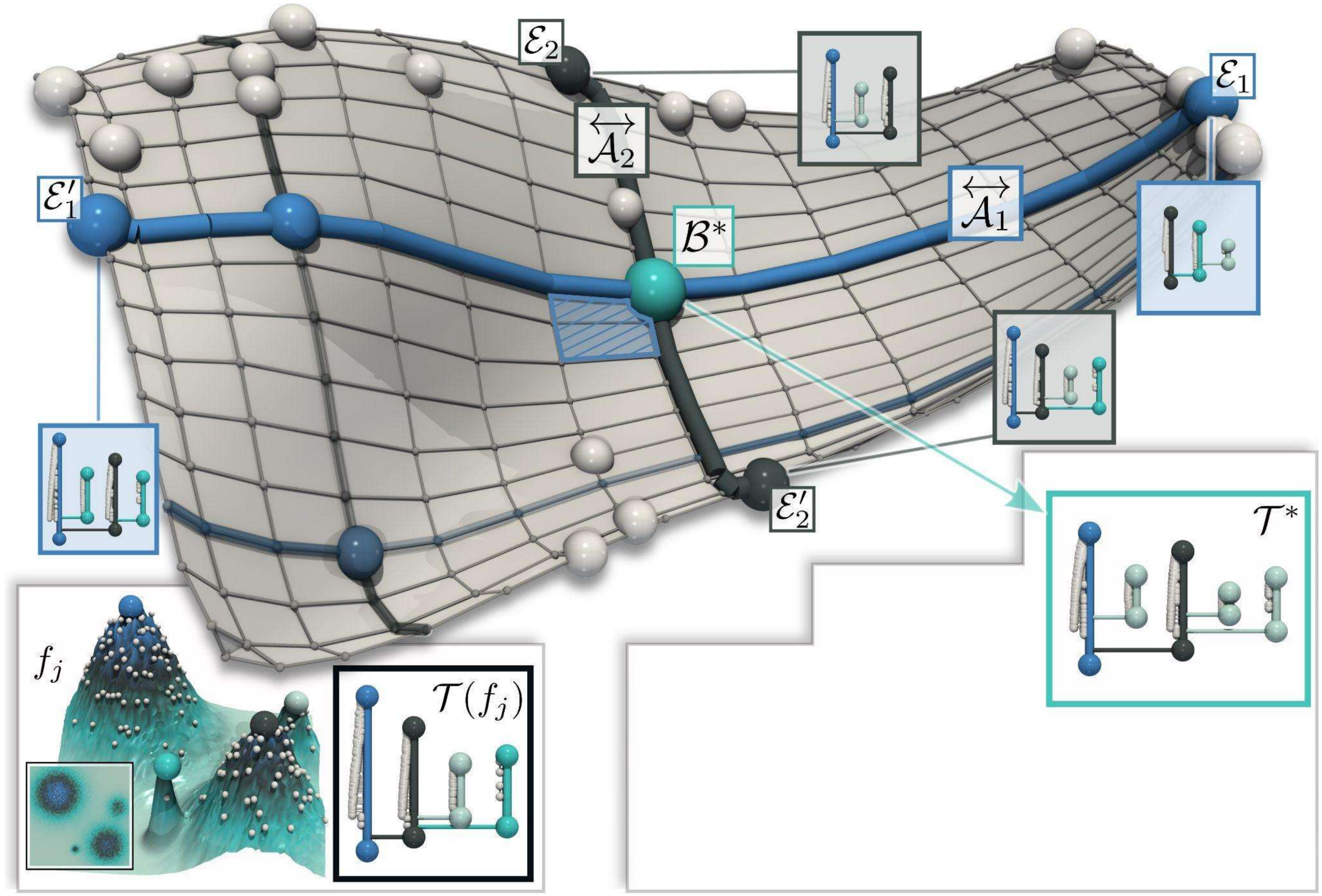
Overview

- **BDT Basis**
  - Barycenter
  - Geodesic axis



Overview

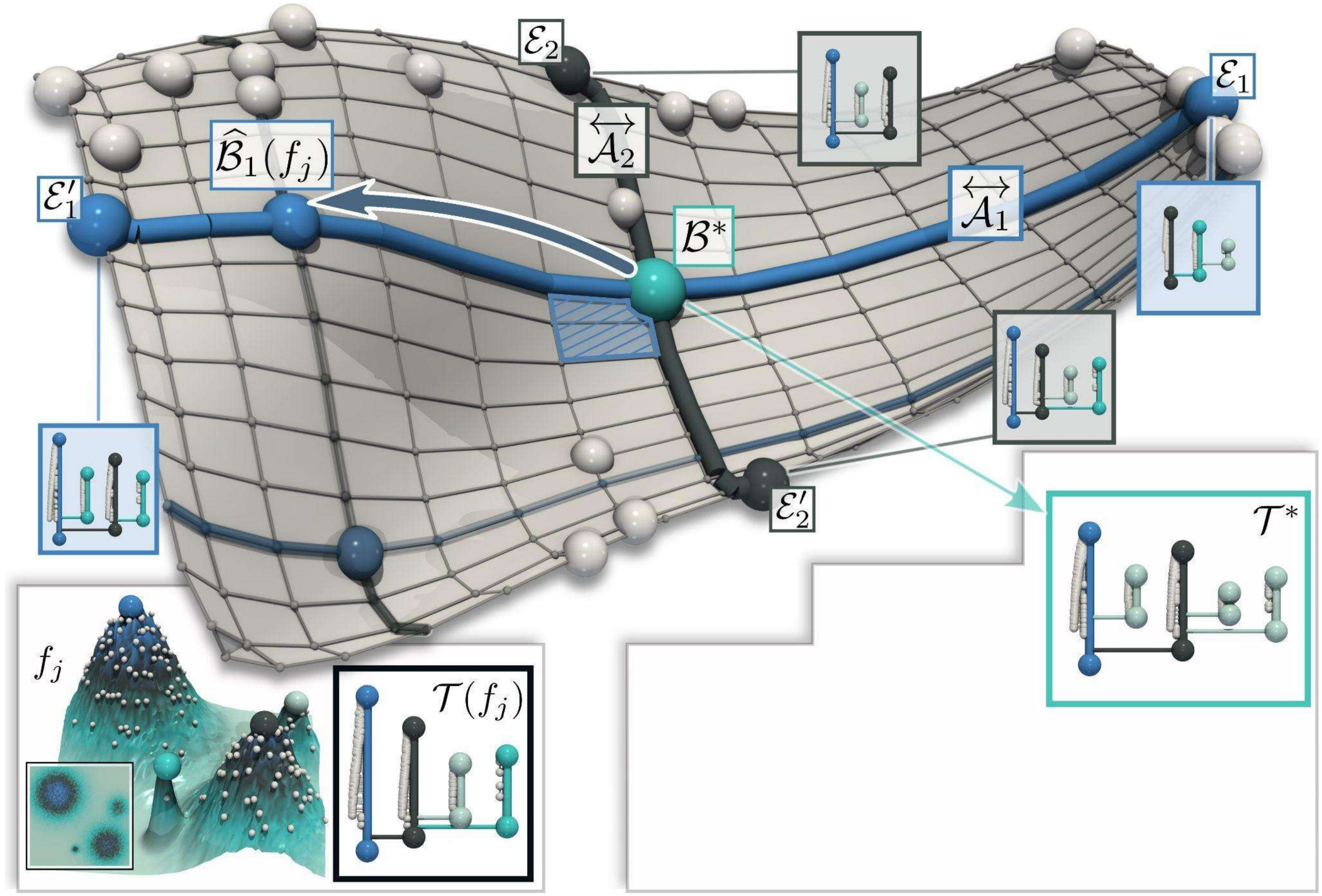
- **BDT Basis**
  - Barycenter
  - Geodesic axis
  - Orthogonal axes



Overview

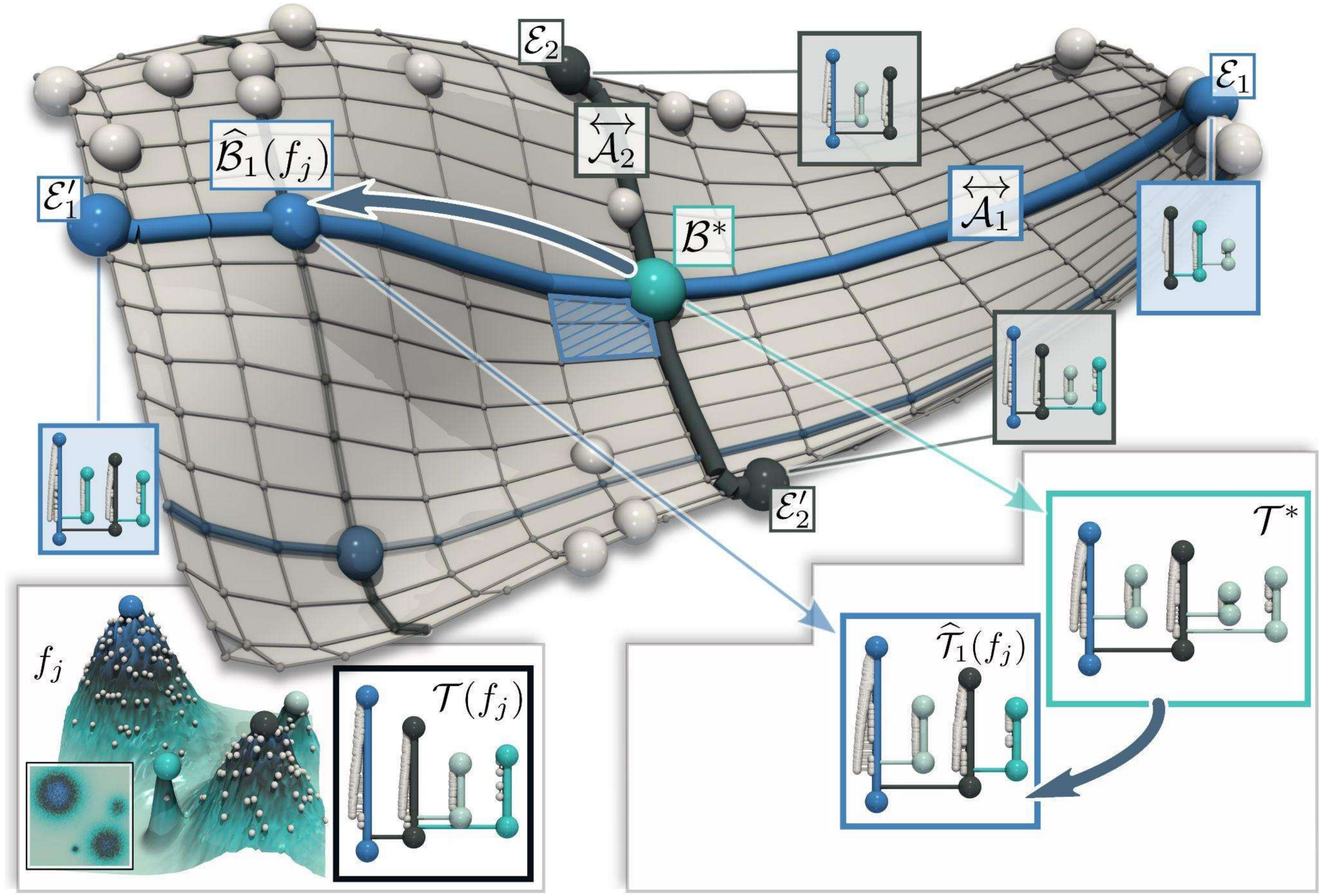
- **BDT Basis**
  - Barycenter
  - Geodesic axis
  - Orthogonal axes

- **Estimation**
  - Axis projection



Overview

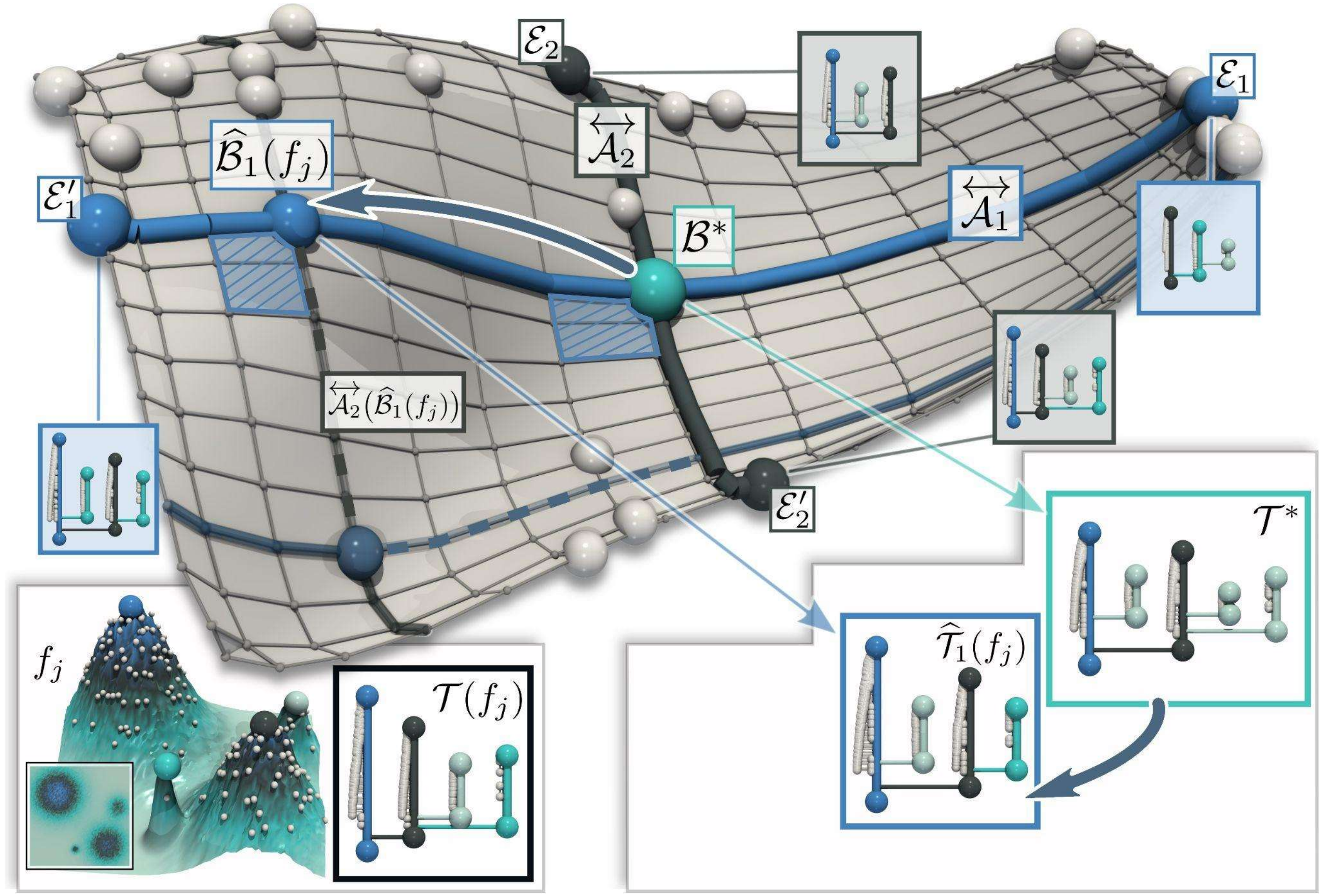
- **BDT Basis**
  - Barycenter
  - Geodesic axis
  - Orthogonal axes
- **Estimation**
  - Axis projection



Overview

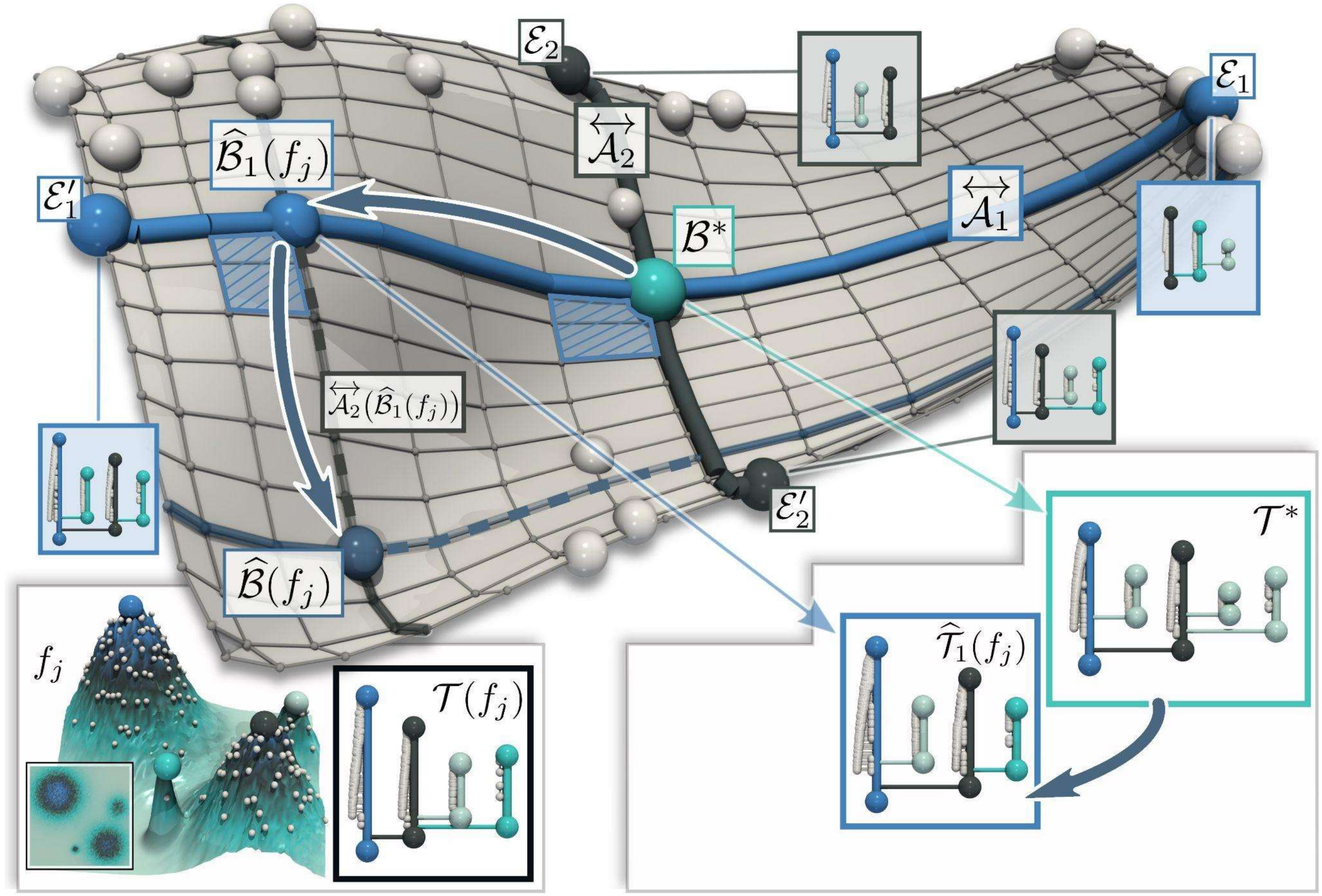
- **BDT Basis**
  - Barycenter
  - Geodesic axis
  - Orthogonal axes

- **Estimation**
  - Axis projection
  - Translated axis



Overview

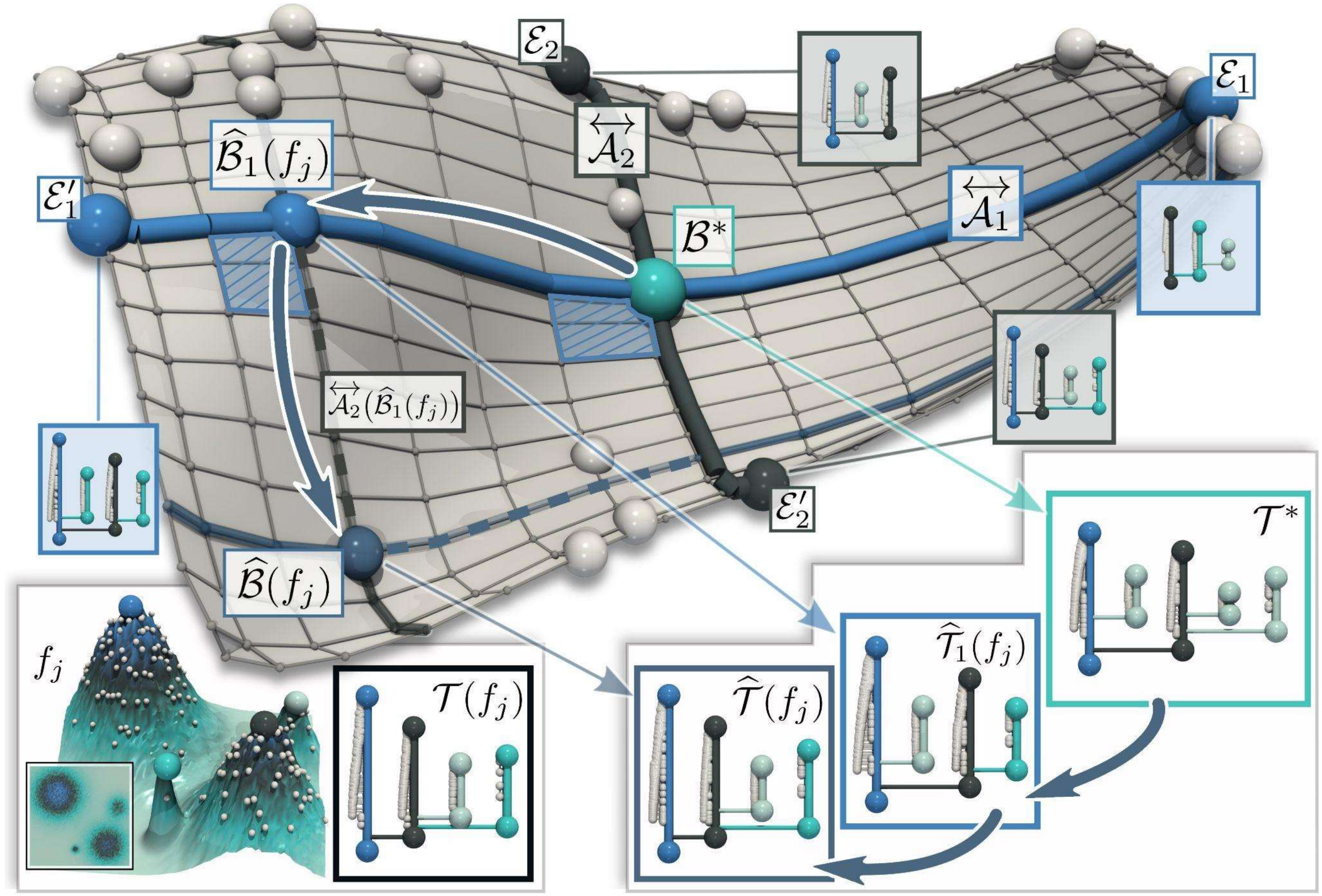
- **BDT Basis**
  - Barycenter
  - Geodesic axis
  - Orthogonal axes
- **Estimation**
  - Axis projection
  - Translated axis



Overview

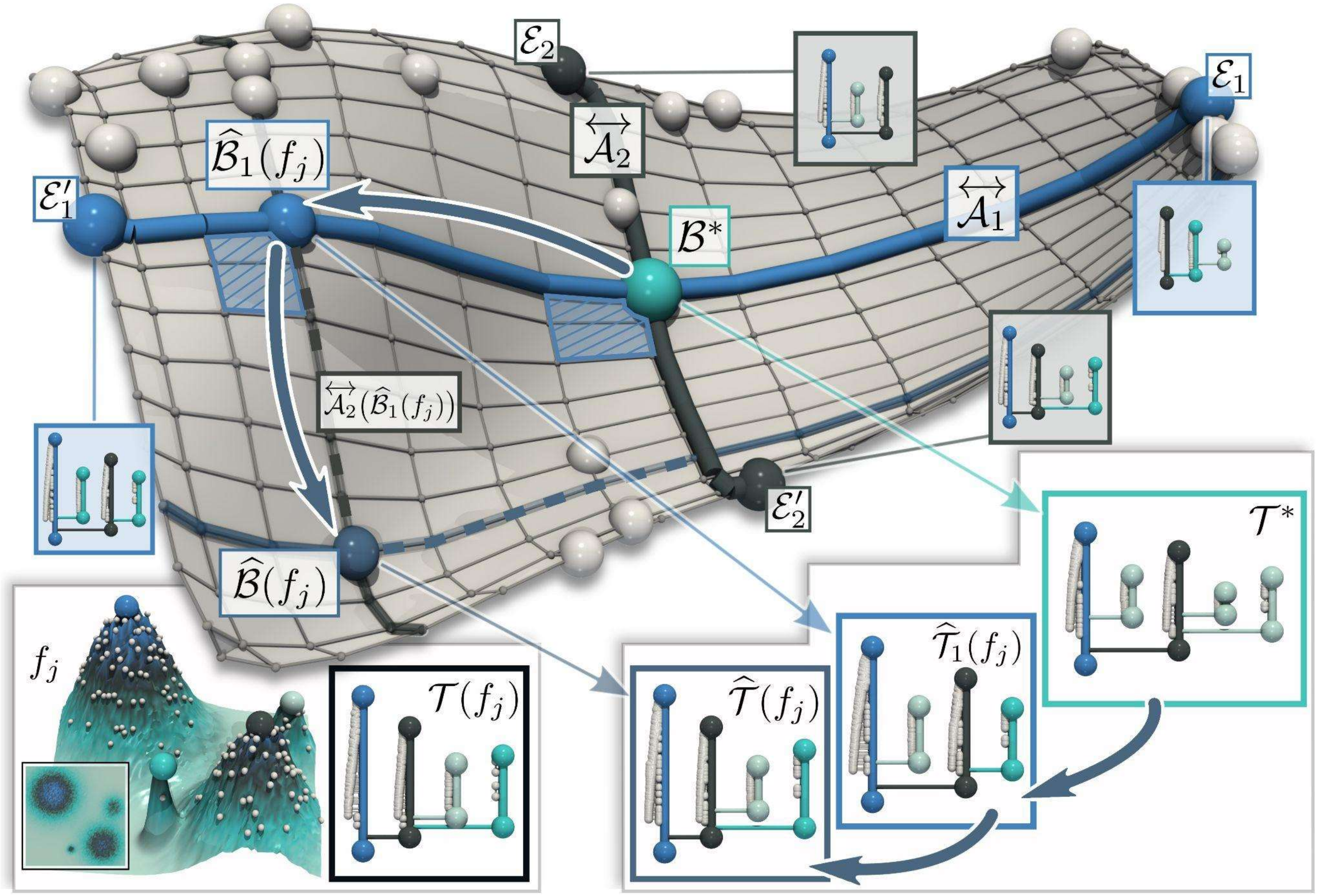
- **BDT Basis**
  - Barycenter
  - Geodesic axis
  - Orthogonal axes

- **Estimation**
  - Axis projection
  - Translated axis



Overview

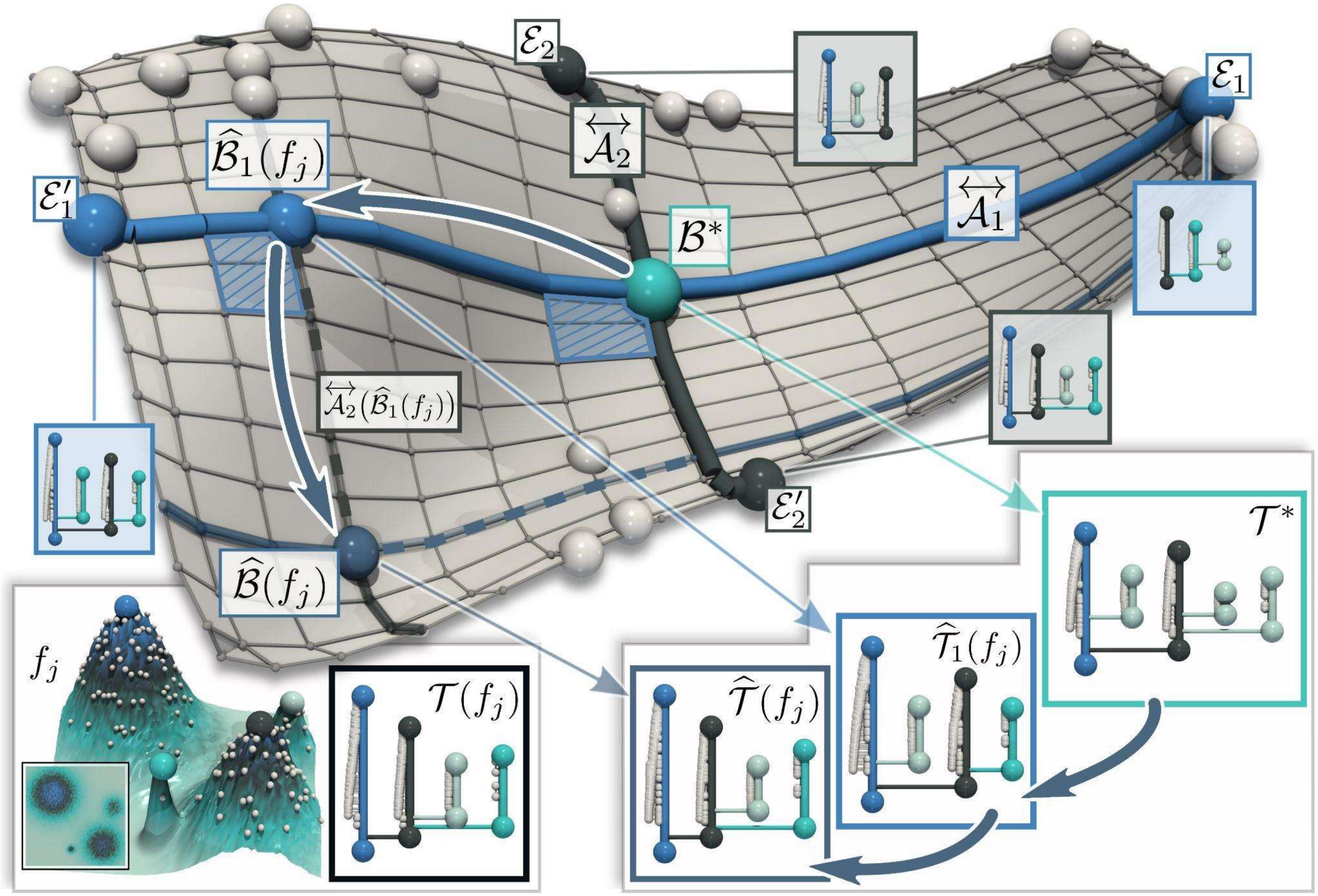
- **BDT Basis**
  - Barycenter
  - Geodesic axis
  - Orthogonal axes
- **Estimation**
  - Axis projection
  - Translated axis



$$E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}}) = \sum_{j=1}^{N} W_2^{\mathcal{T}} \left(\mathcal{B}(f_j), \mathcal{B}^* + \sum_{i=1}^{d'} \overrightarrow{\mathcal{A}_i}(\widehat{\mathcal{B}}_i(f_j)) \right)^2$$

Overview

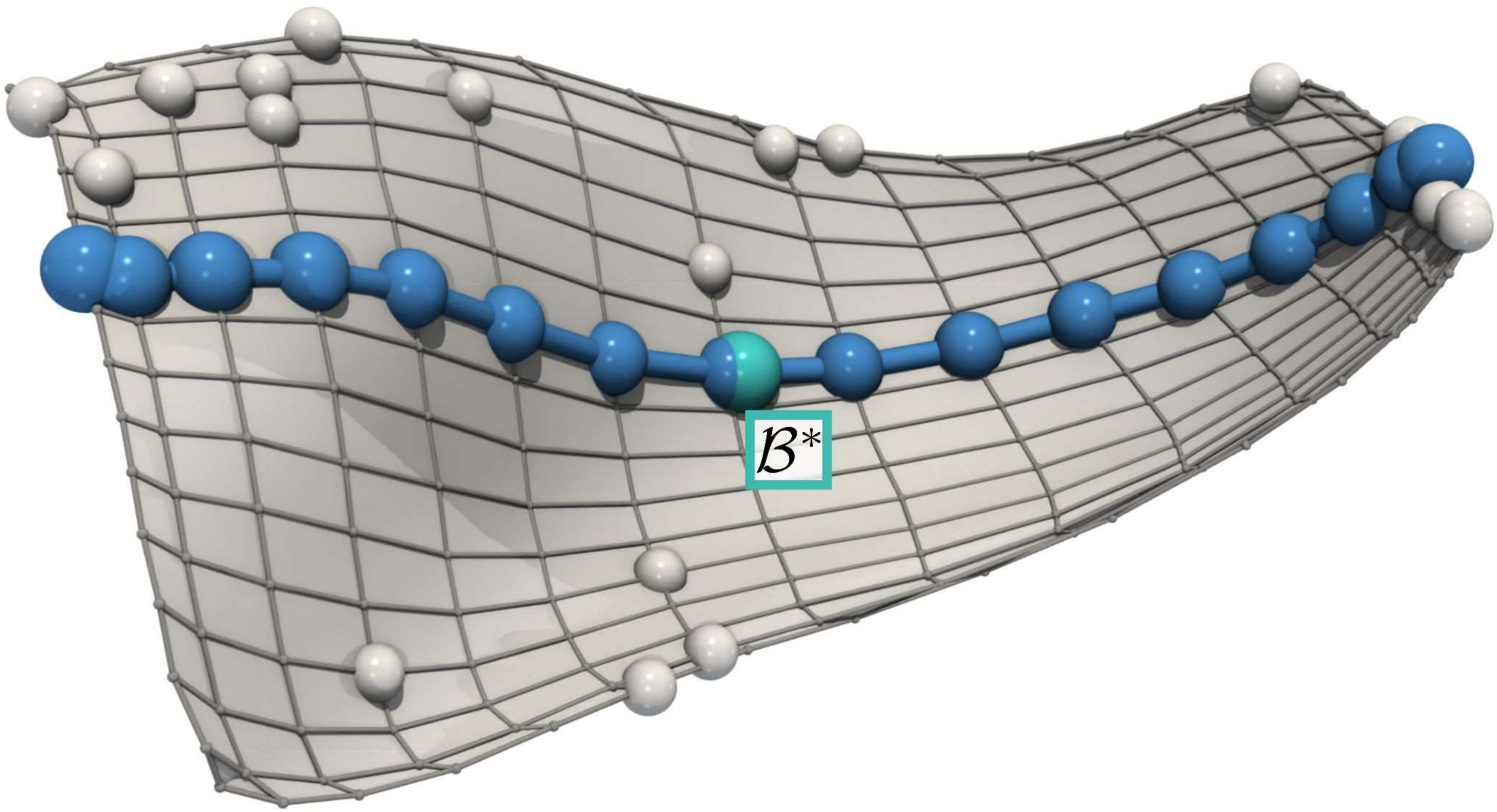
- **BDT Basis**
  - Barycenter
  - Geodesic axis
  - Orthogonal axes
- **Estimation**
  - Axis projection
  - Translated axis



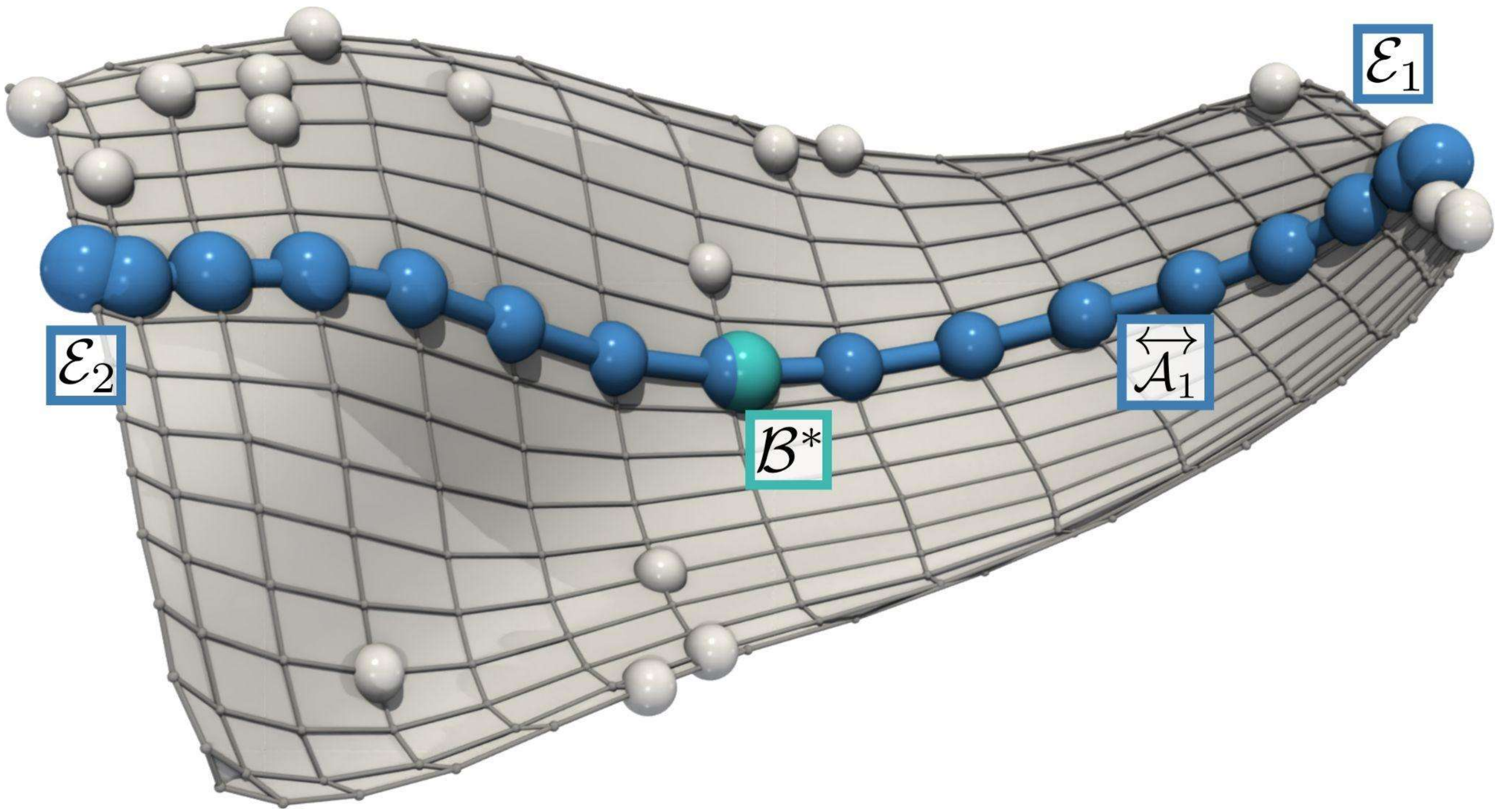
$$E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}}) = \sum_{j=1}^{N} W_2^{\mathcal{T}}\Big(\mathcal{B}(f_j), \mathcal{B}^* + \sum_{i=1}^{d'} \overrightarrow{\mathcal{A}_i}\big(\widehat{\mathcal{B}}_i(f_j)\big)\Big)^2$$

Algorithm

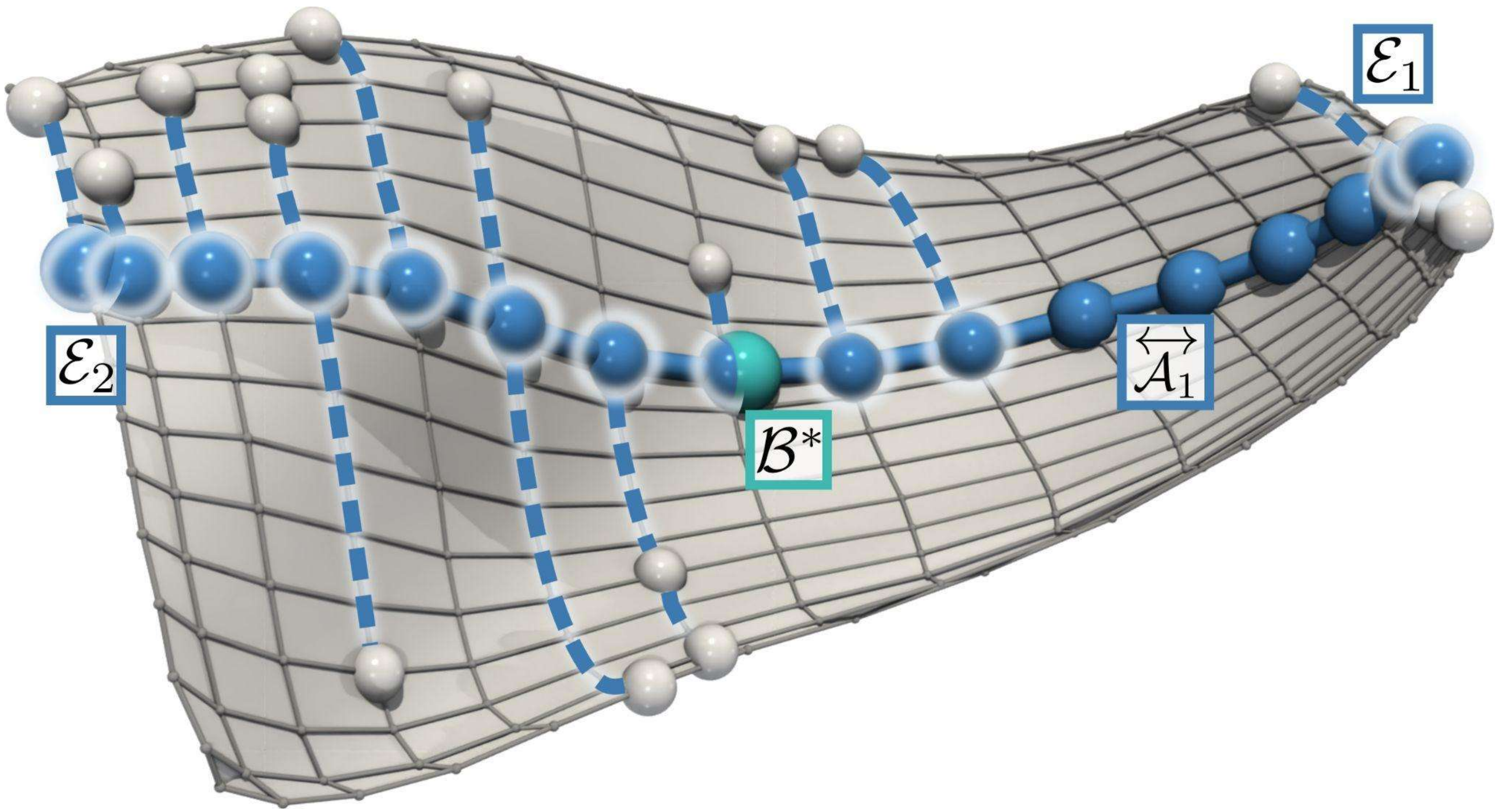
Axis initialization



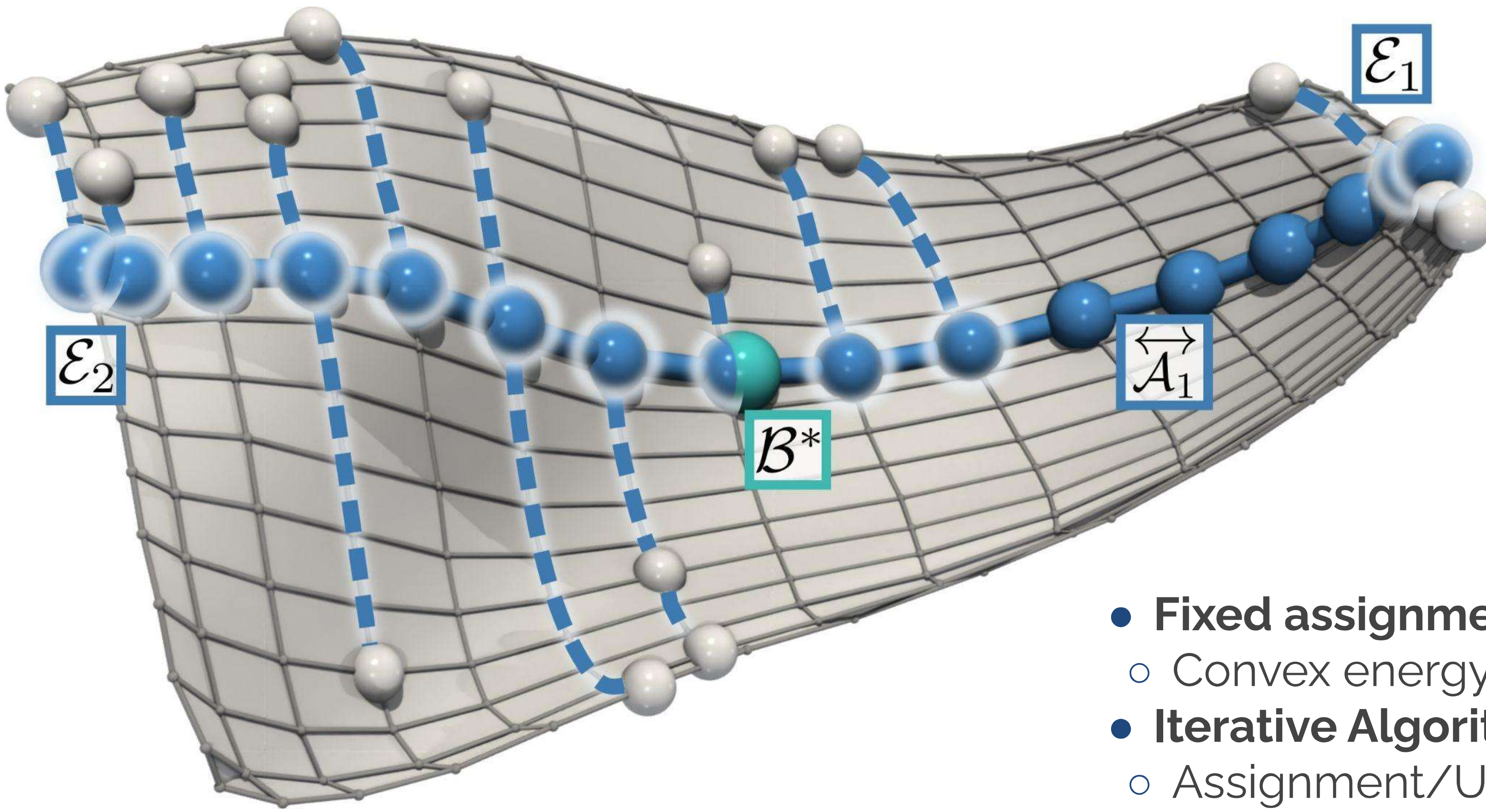
Axis optimization



Axis optimization



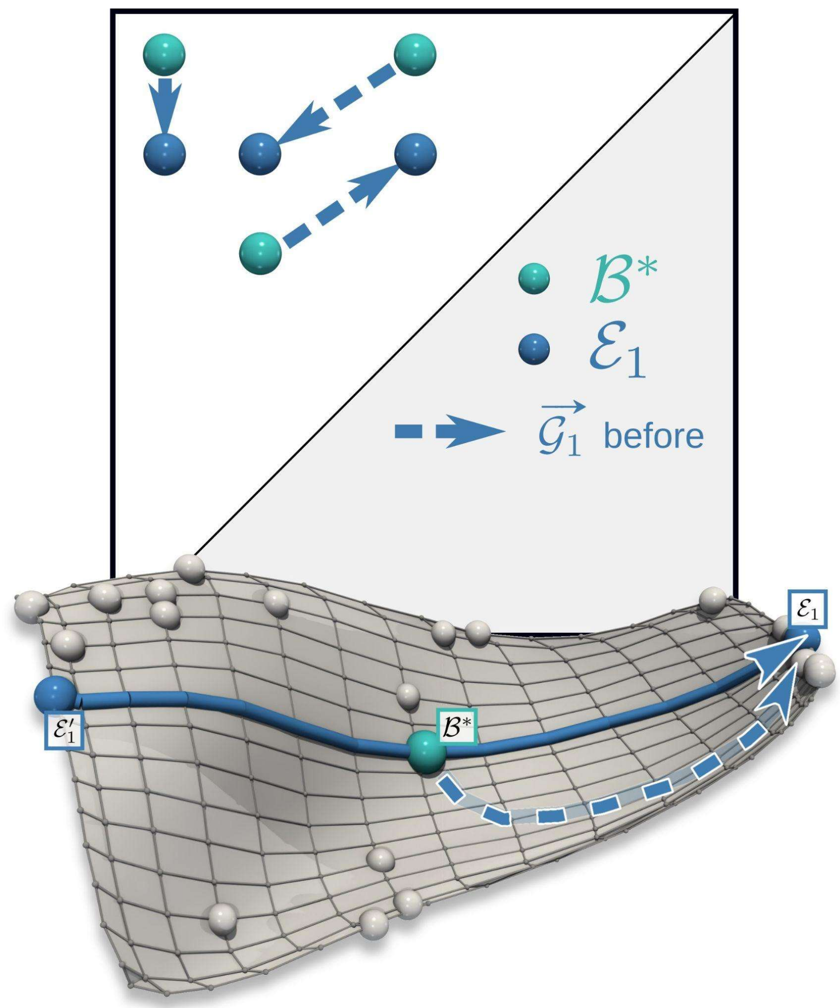
Axis optimization



- **Fixed assignments**
  - Convex energy
- **Iterative Algorithm**
  - Assignment/Update

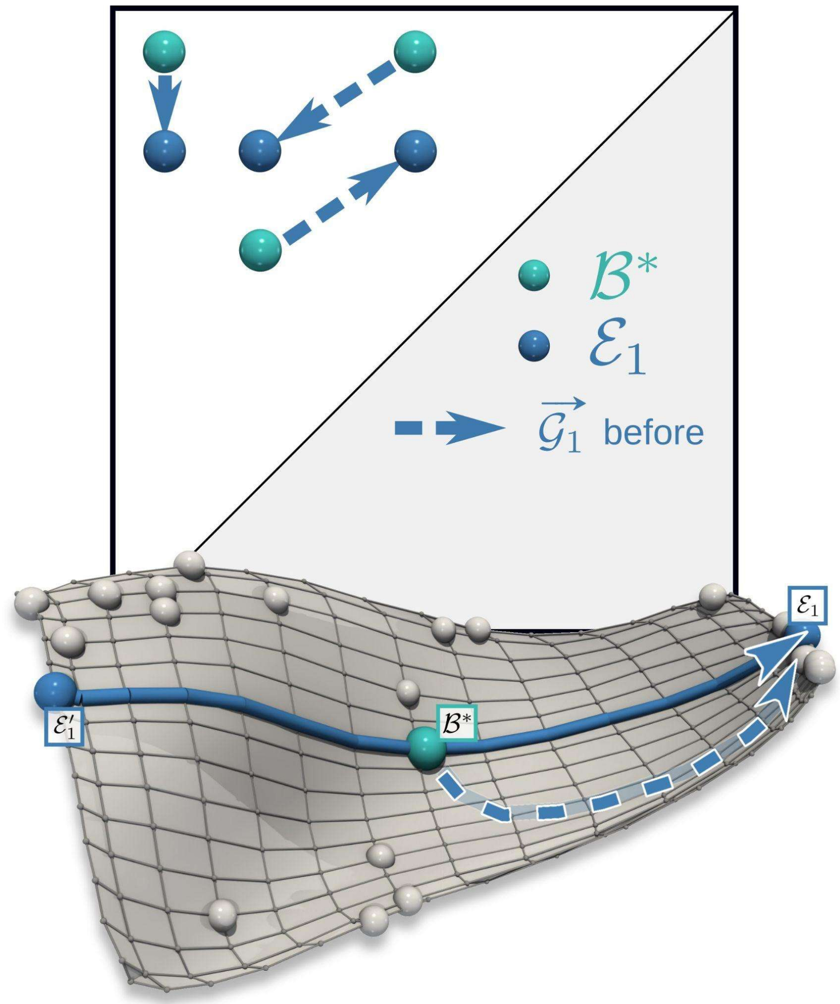
Geodesic enforcement

- **Axis optimization**
  - Axis vectors
    - Optimized freely



Geodesic enforcement

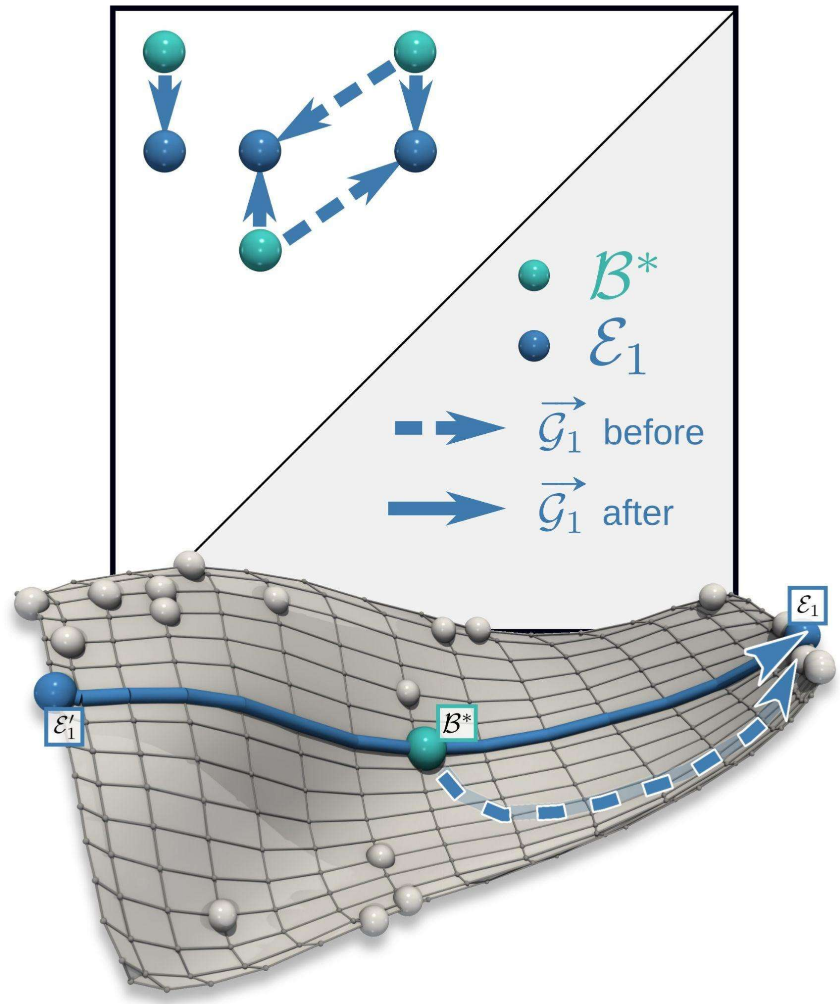
- **Axis optimization**
  - Axis vectors
    - Optimized freely
    - May not be geodesic



Geodesic enforcement

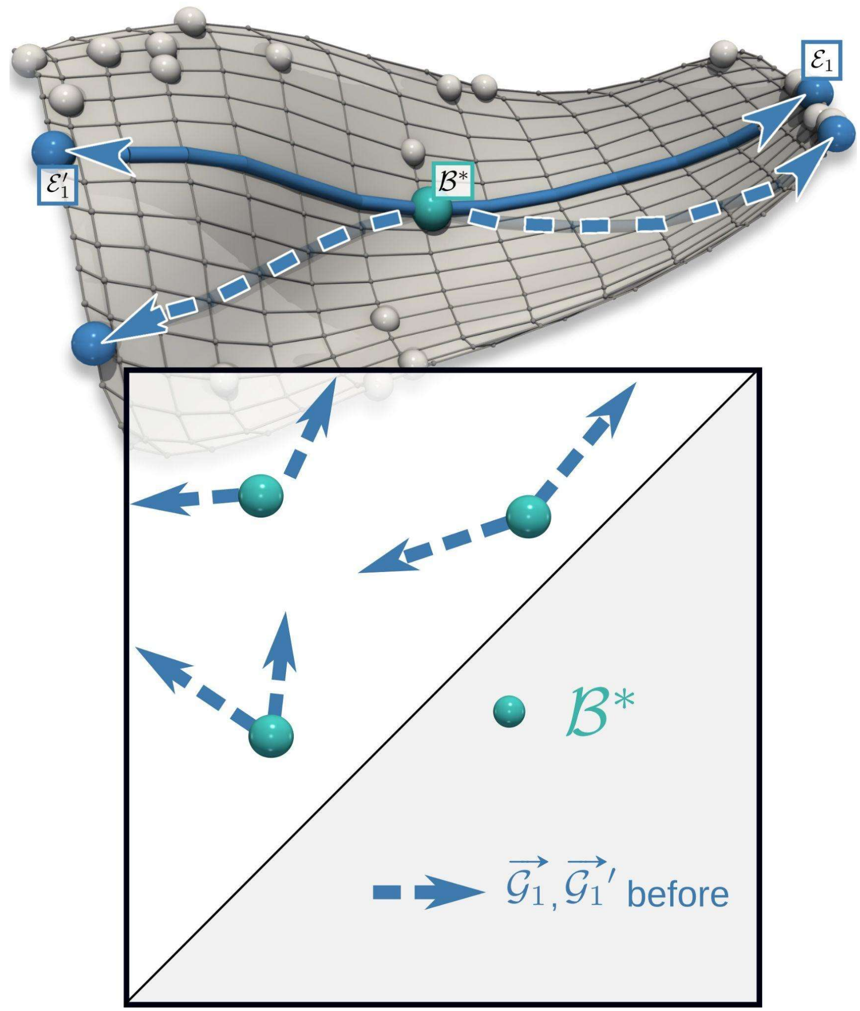
- **Axis optimization**
  - Axis vectors
    - Optimized freely
    - May not be geodesic

- **Geodesic enforcement**
  - Re-compute the assignments



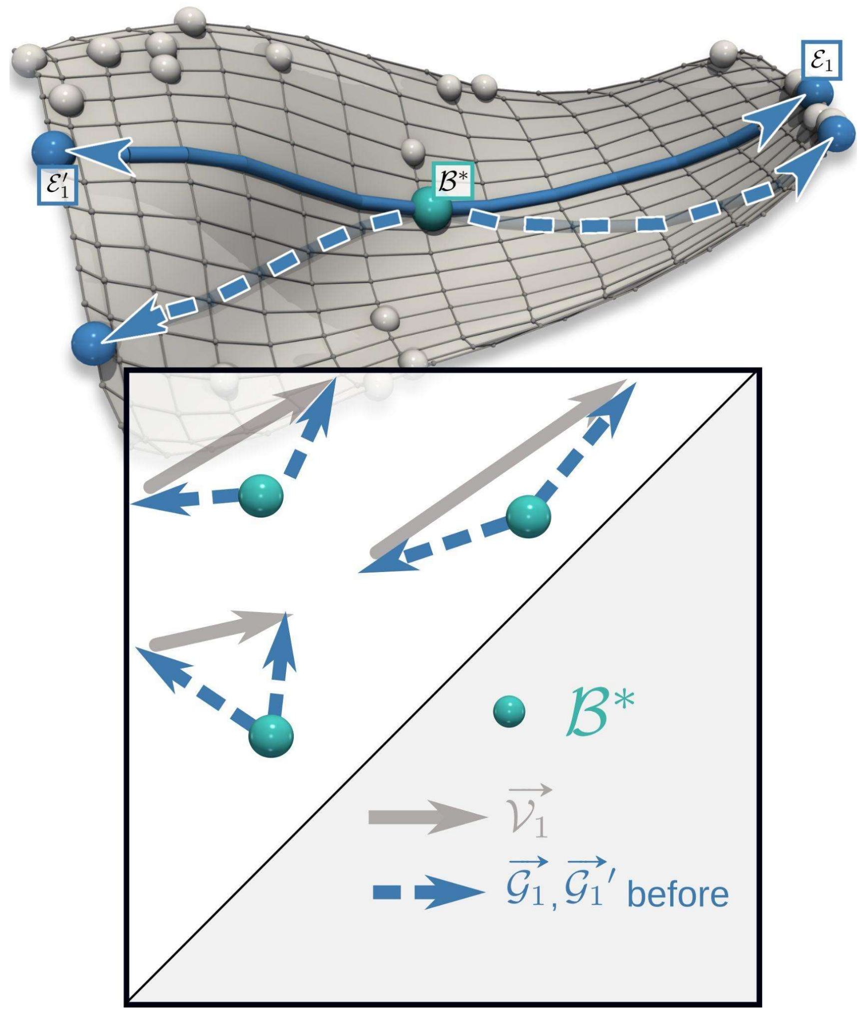
Negative collinearity enforcement

- **Axis optimization**
  - Axis vectors
    - Optimized independently
    - May not be collinear



Negative collinearity enforcement

- **Axis optimization**
  - Axis vectors
    - Optimized independently
    - May not be collinear
- **Negative collinearity enforcement**

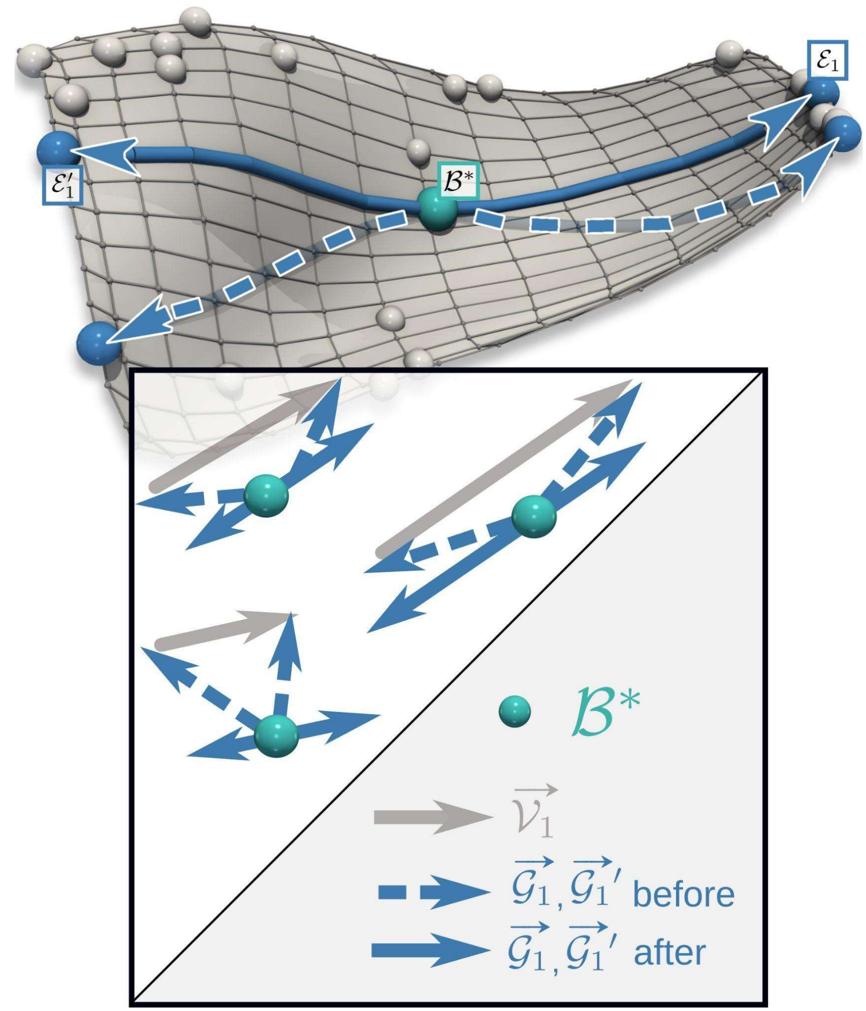


Negative collinearity enforcement

- **Axis optimization**
  - Axis vectors
    - Optimized independently
    - May not be collinear
- **Negative collinearity enforcement**
  - Updated geodesics

$$\overrightarrow{\mathcal{G}_{d'}} \leftarrow \beta' \times \overrightarrow{\mathcal{V}_{d'}}$$

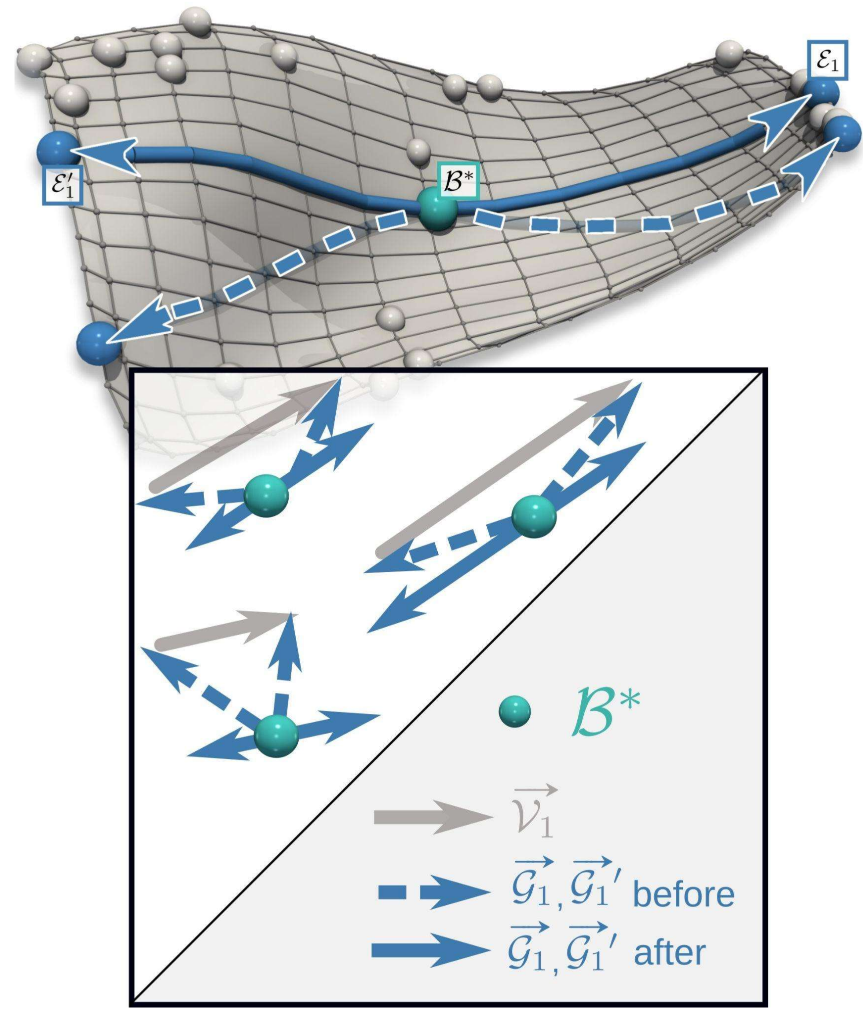
$$\overrightarrow{\mathcal{G}'_{d'}} \leftarrow -(1-\beta') \times \overrightarrow{\mathcal{V}_{d'}}$$



Negative collinearity enforcement

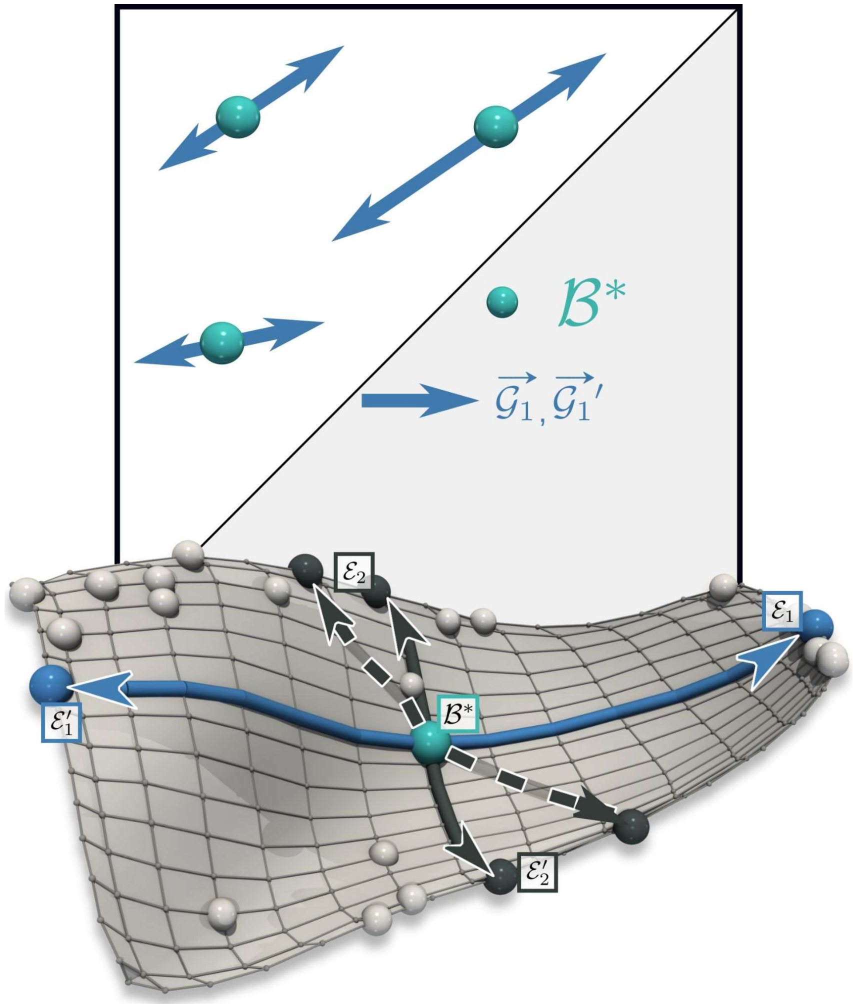
- **Axis optimization**
  - Axis vectors
    - Optimized independently
    - May not be collinear
- **Negative collinearity enforcement**
  - Updated geodesics

$$\overrightarrow{\mathcal{G}_{d'}} \leftarrow \beta' \times \overrightarrow{\mathcal{V}_{d'}}$$
$$\overrightarrow{\mathcal{G}'_{d'}} \leftarrow -(1-\beta') \times \overrightarrow{\mathcal{V}_{d'}}$$
$$\beta' = ||\overrightarrow{\mathcal{G}_{d'}}||/(||\overrightarrow{\mathcal{G}_{d'}}|| + ||\overrightarrow{\mathcal{G}'_{d'}}||)$$



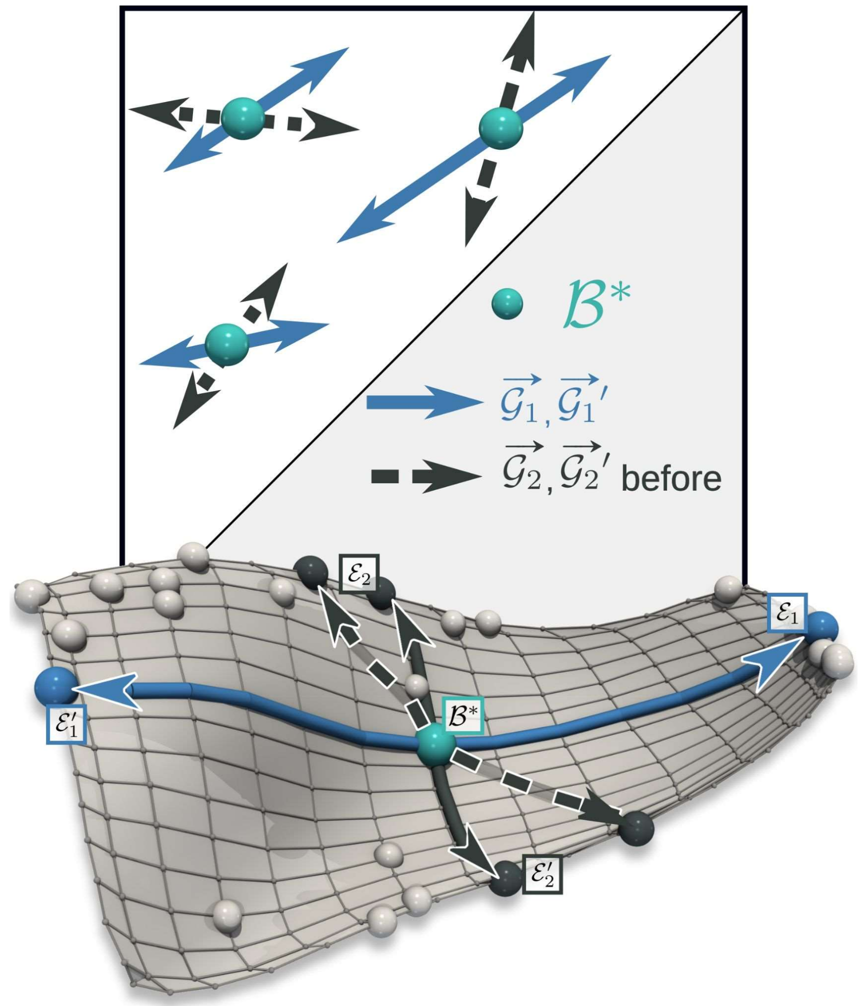
Orthogonality enforcement

- **Axis optimization**
  - Axis vectors
    - Optimized freely
    - May not be orthogonal



Orthogonality enforcement

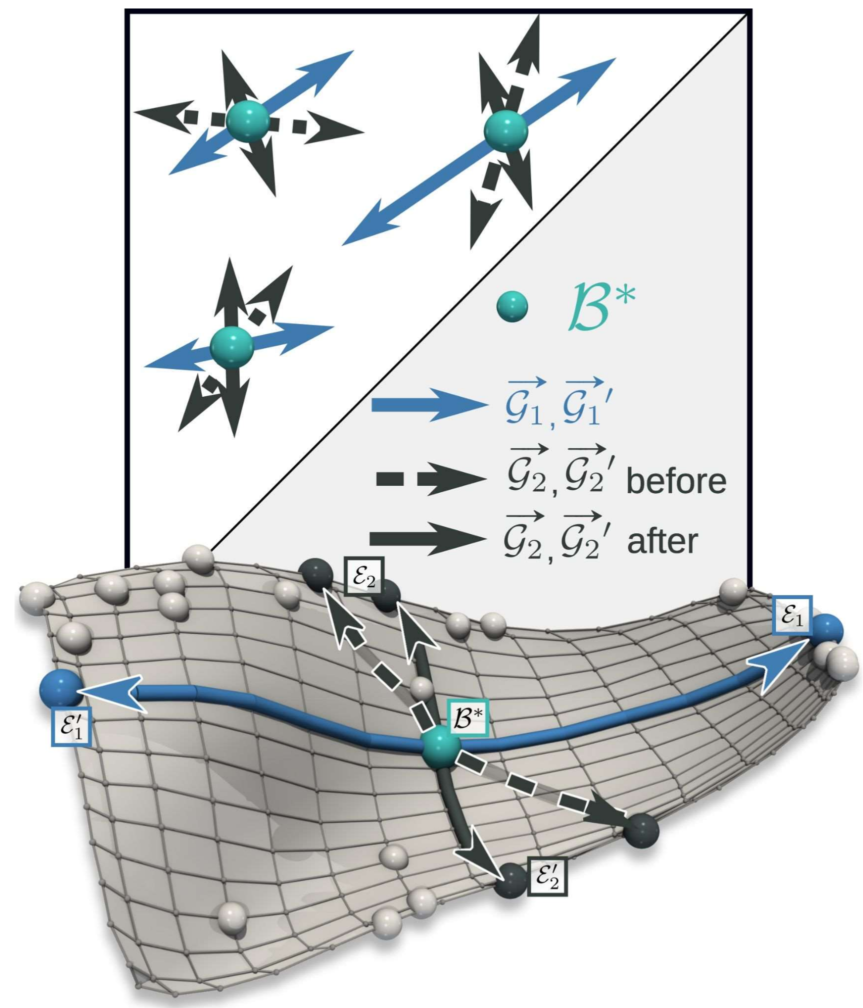
- **Axis optimization**
  - Axis vectors
    - Optimized freely
    - May not be orthogonal



Orthogonality enforcement

- **Axis optimization**
  - Axis vectors
    - Optimized freely
    - May not be orthogonal

- **Orthogonality enforcement**
  - Gram-Schmidt orthogonalization

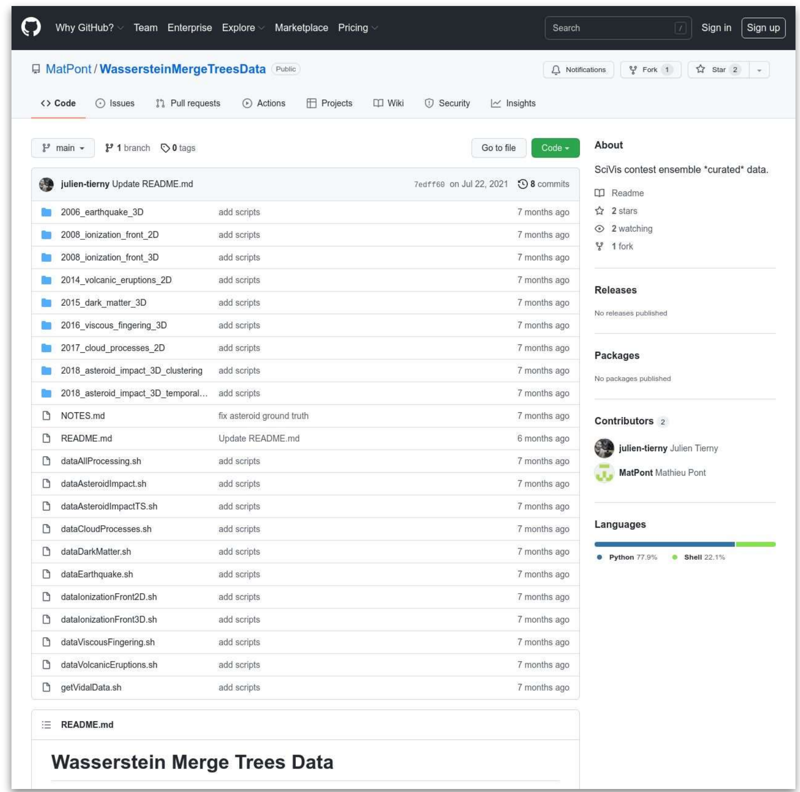


Experiments

Benchmark data

- **https://github.com/MatPont/WassersteinMergeTreesData**
  - Publicly available ensembles
    - **http://sciviscontest.ieeevis.org/**
    - 12 ensembles
    - 7 < N < 48
    - Branches: Dozens to thousands
  - Curation scripts & data & ground-truth

- **Application domains**
  - Astrophysics
  - Meteorology
  - Material physics
  - Volcanic measurements
  - Fluid mechanics, etc.



Time performance

Dataset	N	$\|\mathcal{B}\|$	PD-PGA			MT-PGA		
			1 c.	20 c.	Speedup	1 c.	20 c.	Speedup
Asteroid Impact (3D)	7	1,295	1,392.17	147.40	9.44	1,180.72	117.97	10.01
Cloud processes (2D)	12	1,209	817.64	61.88	13.21	517.94	38.49	13.46
Viscous fingering (3D)	15	118	86.69	9.17	9.45	42.89	4.71	9.11
Dark matter (3D)	40	2,592	18,388.86	1,366.45	13.46	24,480.42	1,758.04	13.92
Volcanic eruptions (2D)	12	811	460.17	37.99	12.11	1,004.37	81.75	12.29
Ionization front (2D)	16	135	104.74	12.00	8.73	55.73	6.26	8.90
Ionization front (3D)	16	763	3,750.00	300.96	12.46	4,029.71	294.29	13.69
Earthquake (3D)	12	1,203	3,896.52	338.64	11.51	1,973.49	158.12	12.48
Isabel (3D)	12	1,338	1,969.79	164.49	11.98	1,472.54	115.66	12.73
Starting Vortex (2D)	12	124	17.71	2.72	6.51	11.51	1.65	6.98
Sea Surface Height (2D)	48	1,787	12,420.98	670.00	18.54	27,791.00	1,669.52	16.65
Vortex Street (2D)	45	23	18.75	2.69	6.97	35.79	3.93	9.11

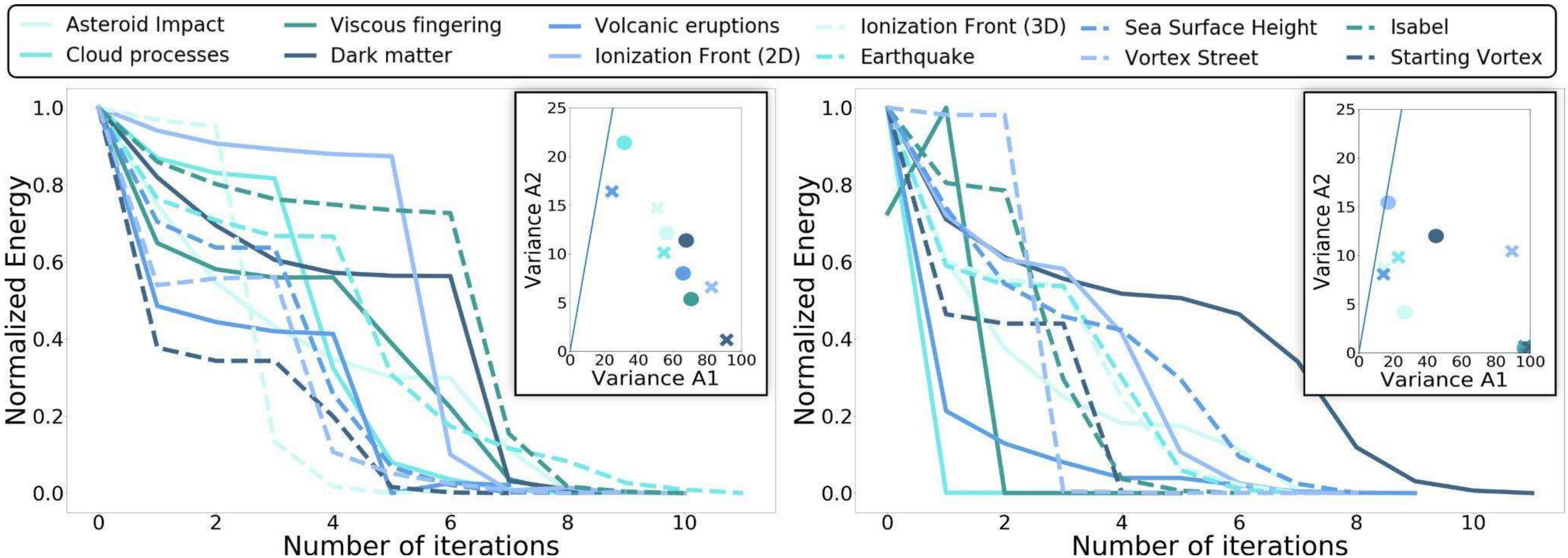
Computation time (s.)

Time performance

Dataset	N	$\|\mathcal{B}\|$	PD-PGA			MT-PGA		
			1 c.	20 c.	Speedup	1 c.	20 c.	Speedup
Asteroid Impact (3D)	7	1,295	1,392.17	147.40	9.44	1,180.72	117.97	10.01
Cloud processes (2D)	12	1,209	817.64	61.88	13.21	517.94	38.49	13.46
Viscous fingering (3D)	15	118	86.69	9.17	9.45	42.89	4.71	9.11
Dark matter (3D)	40	2,592	18,388.86	1,366.45	13.46	24,480.42	1,758.04	13.92
Volcanic eruptions (2D)	12	811	460.17	37.99	12.11	1,004.37	81.75	12.29
Ionization front (2D)	16	135	104.74	12.00	8.73	55.73	6.26	8.90
Ionization front (3D)	16	763	3,750.00	300.96	12.46	4,029.71	294.29	13.69
Earthquake (3D)	12	1,203	3,896.52	338.64	11.51	1,973.49	158.12	12.48
Isabel (3D)	12	1,338	1,969.79	164.49	11.98	1,472.54	115.66	12.73
Starting Vortex (2D)	12	124	17.71	2.72	6.51	11.51	1.65	6.98
Sea Surface Height (2D)	48	1,787	12,420.98	670.00	18.54	27,791.00	1,669.52	16.65
Vortex Street (2D)	45	23	18.75	2.69	6.97	35.79	3.93	9.11

Computation time (s.)

Energy evolution



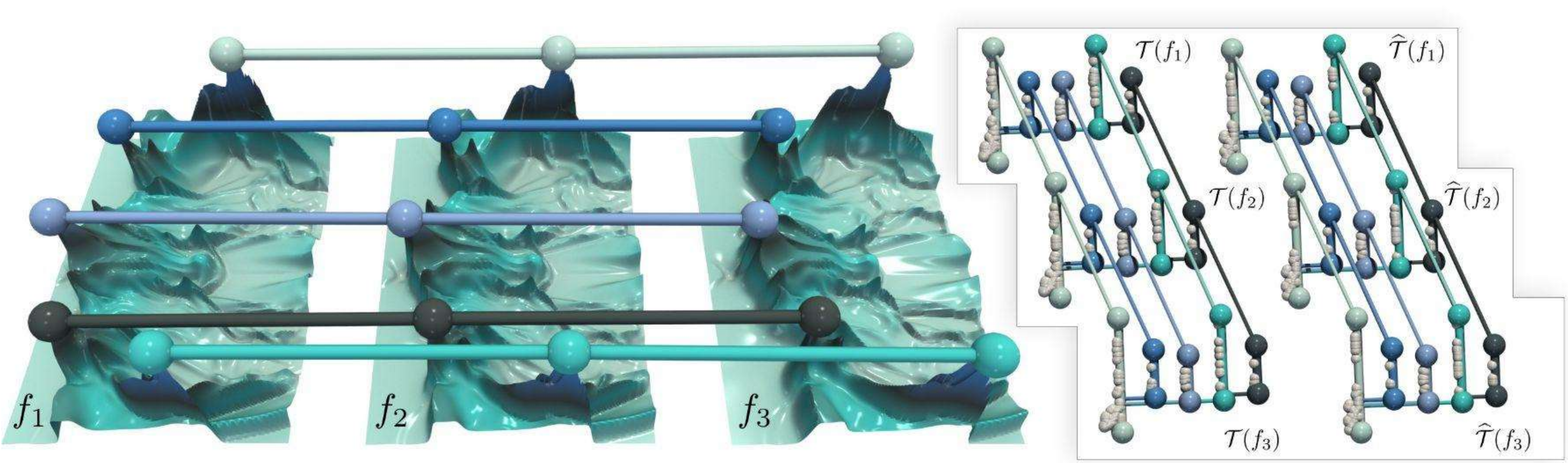
PD-PGA

MT-PGA

Applications

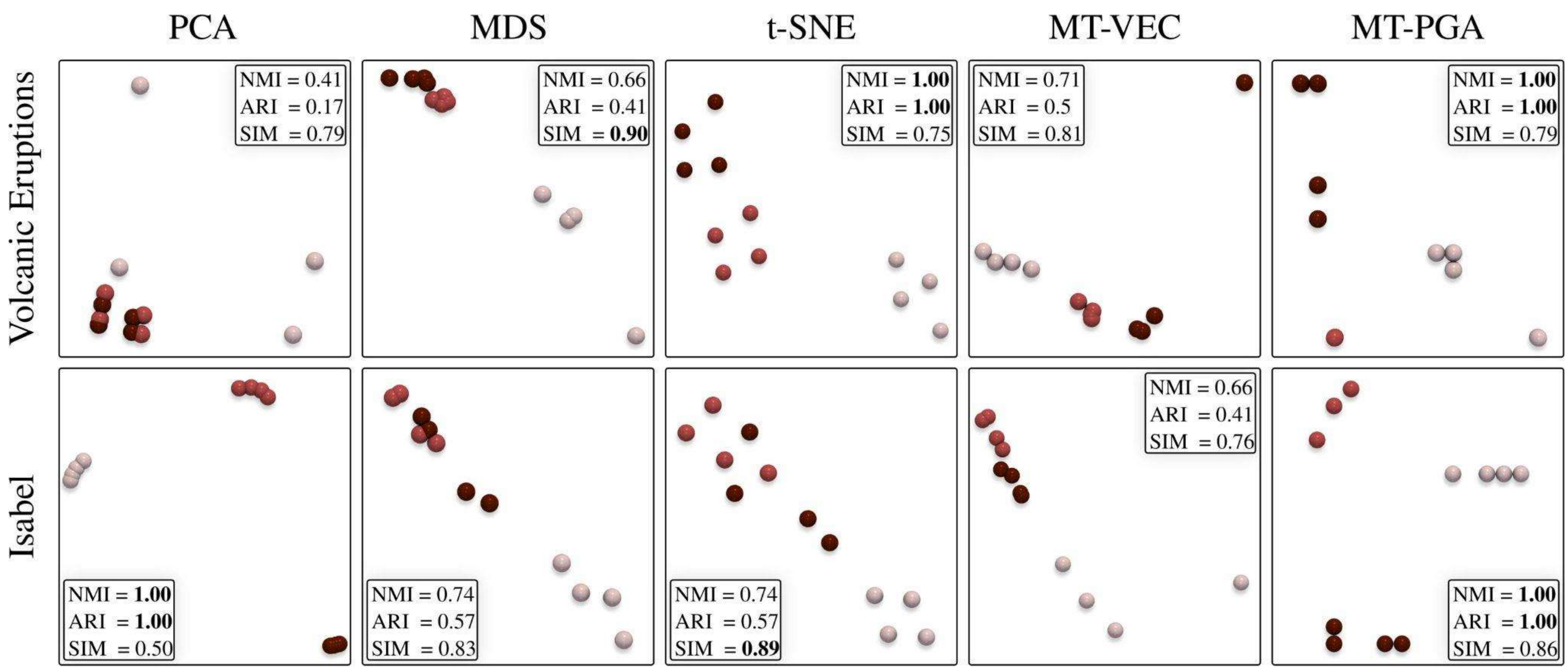
Data reduction

- **Only store**
  - The basis
  - The coordinates

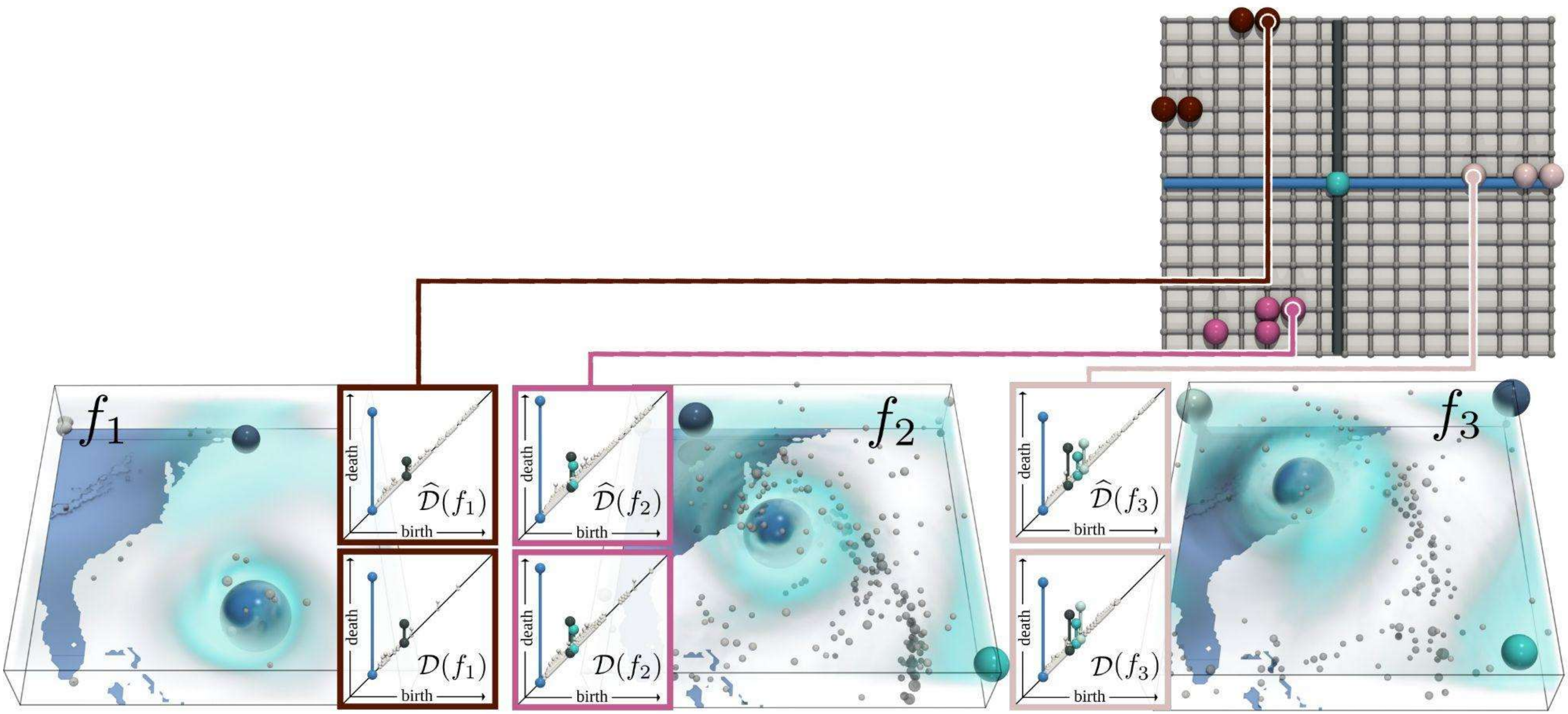


Compression factor: 5.12

Dimensionality reduction

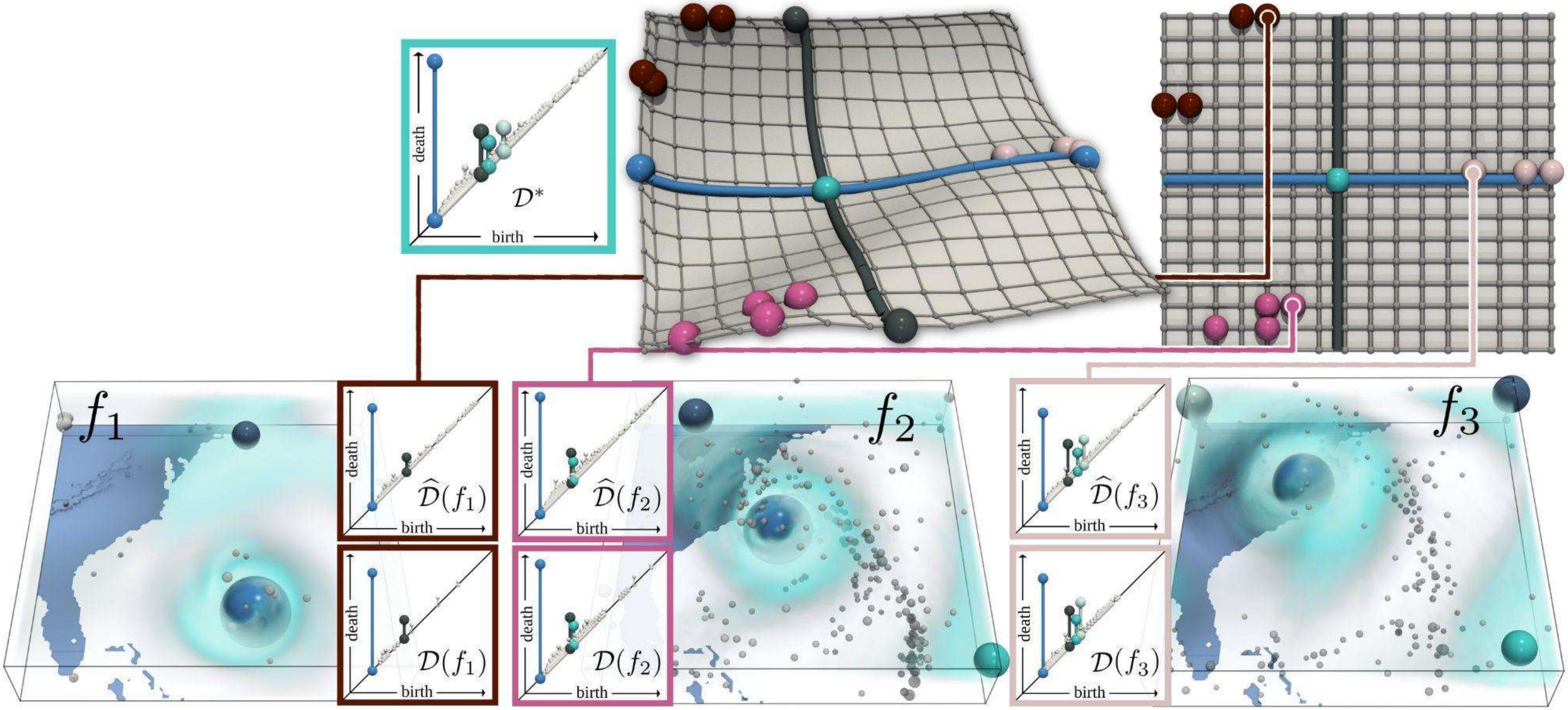


PD-PGA



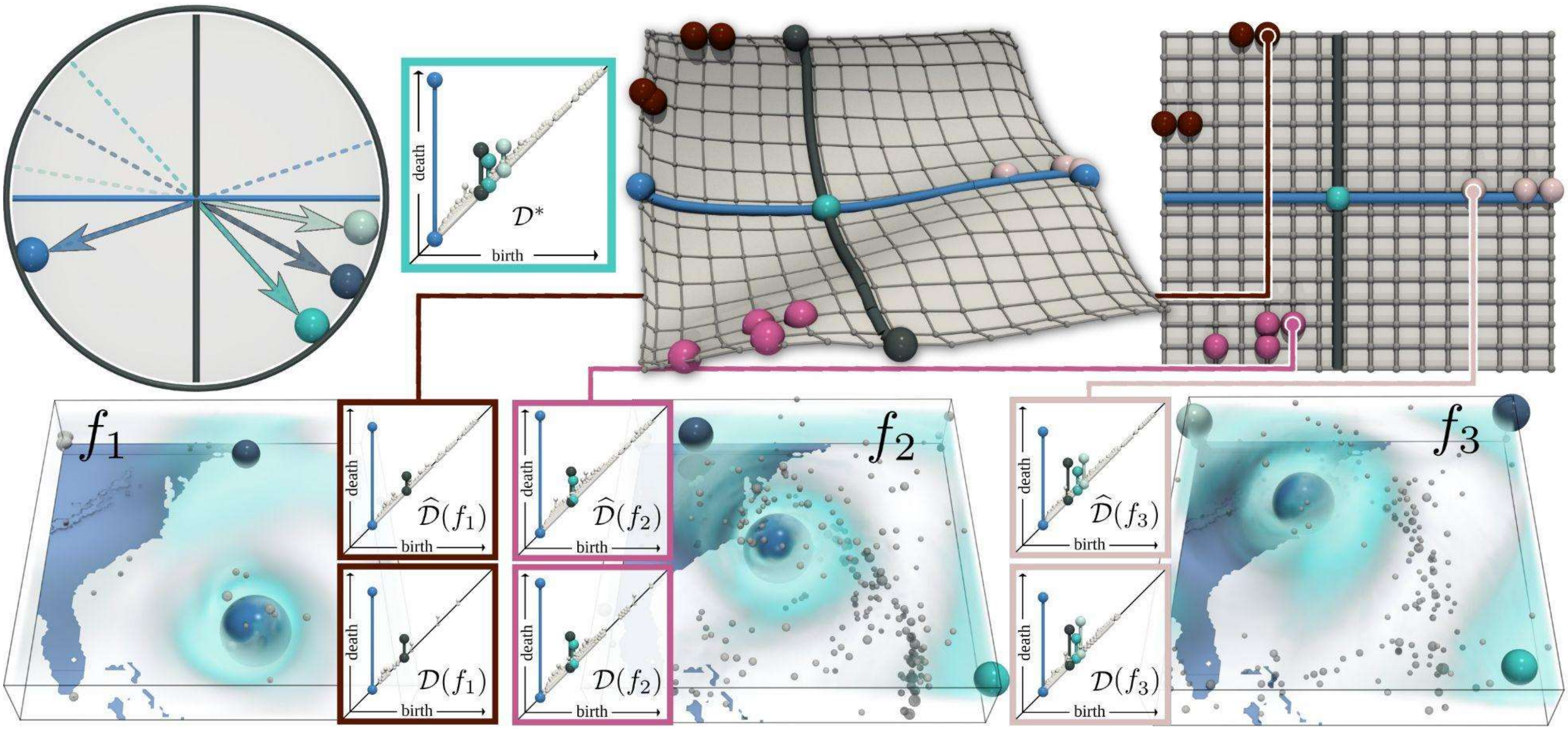
f_1
f_2
f_3
death
birth
$\widehat{\mathcal{D}}(f_1)$
$\mathcal{D}(f_1)$
$\widehat{\mathcal{D}}(f_2)$
$\mathcal{D}(f_2)$
$\widehat{\mathcal{D}}(f_3)$
$\mathcal{D}(f_3)$

PD-PGA



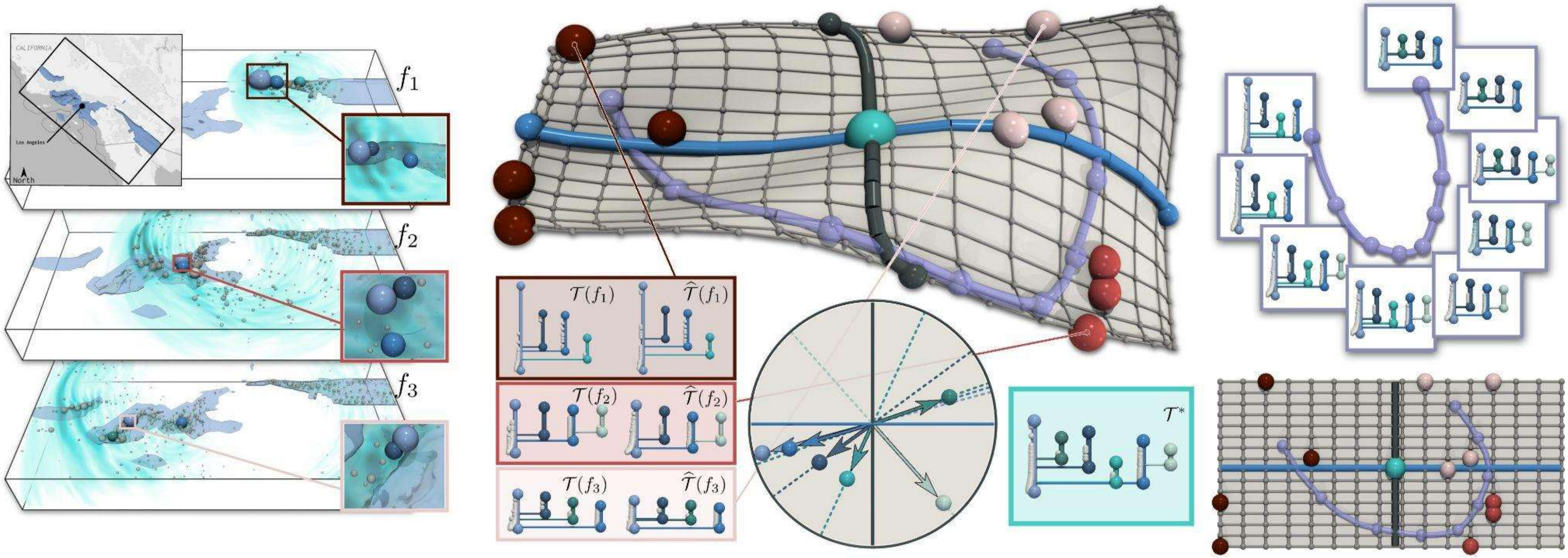
$\mathcal{D}^*$
death
birth
f_1
f_2
f_3
$\widehat{\mathcal{D}}(f_1)$
$\mathcal{D}(f_1)$
$\widehat{\mathcal{D}}(f_2)$
$\mathcal{D}(f_2)$
$\widehat{\mathcal{D}}(f_3)$
$\mathcal{D}(f_3)$

PD-PGA



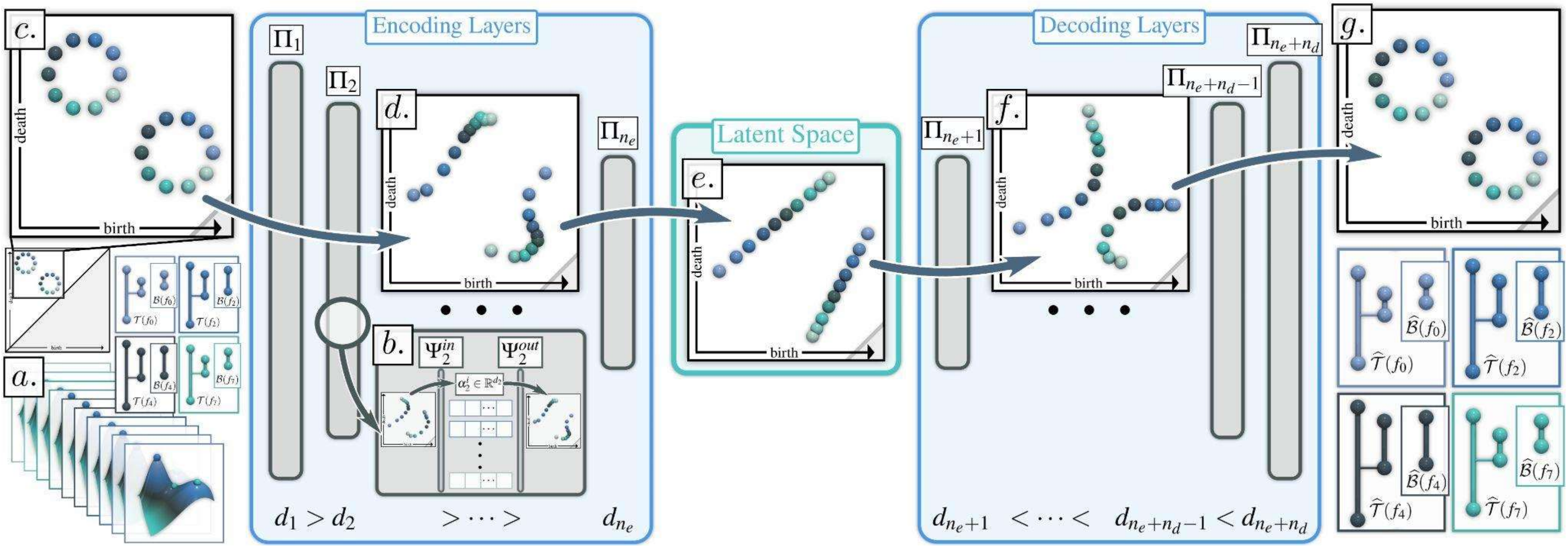
death
birth
$\mathcal{D}^*$
f_1
f_2
f_3
$\widehat{\mathcal{D}}(f_1)$
$\mathcal{D}(f_1)$
$\widehat{\mathcal{D}}(f_2)$
$\mathcal{D}(f_2)$
$\widehat{\mathcal{D}}(f_3)$
$\mathcal{D}(f_3)$

MT-PGA



f_1
f_2
f_3
CALIFORNIA
Los Angeles
North
$\mathcal{T}(f_1)$
$\widehat{\mathcal{T}}(f_1)$
$\mathcal{T}(f_2)$
$\widehat{\mathcal{T}}(f_2)$
$\mathcal{T}(f_3)$
$\widehat{\mathcal{T}}(f_3)$
$\mathcal{T}^*$

Beyond linear combinations

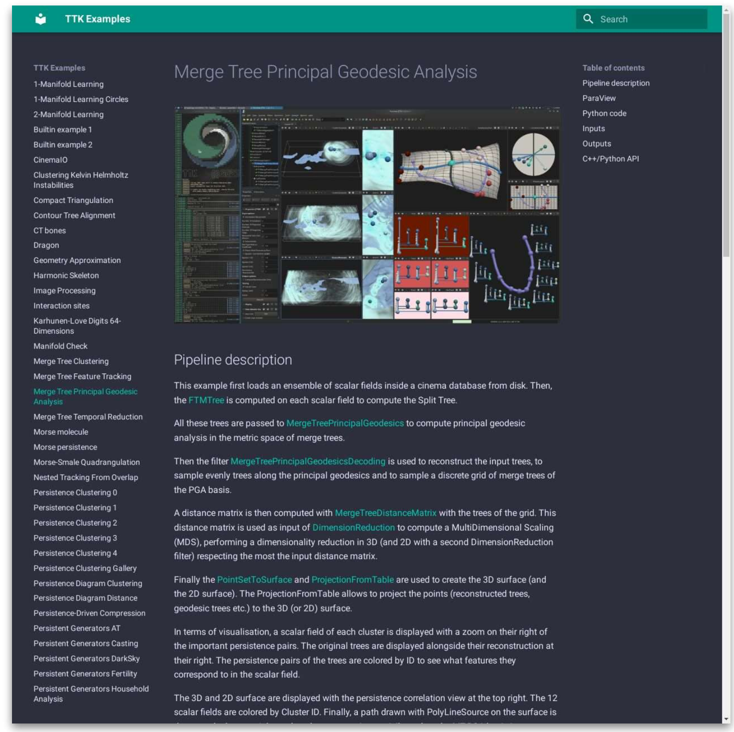


- **Wasserstein Auto-Encoders of Merge-Trees (and Persistence Diagrams)**
  - IEEE TVCG 2024 (presented at IEEE VIS 2024)

Try it with TTK!

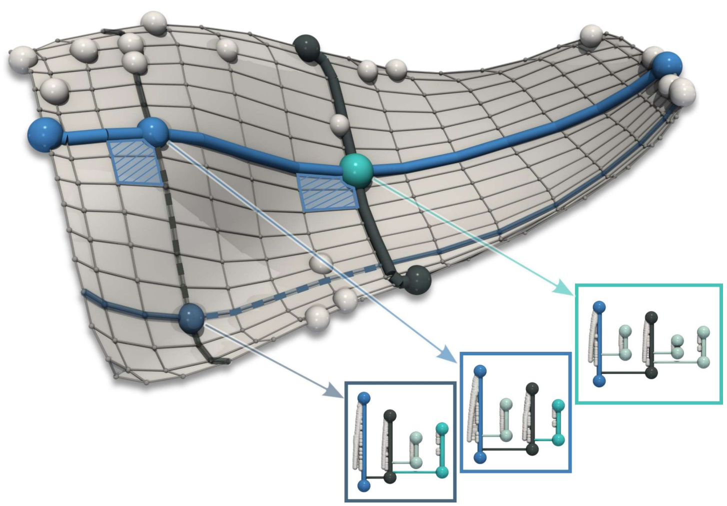
- **https://topology-tool-kit.github.io/installation.html**
  - ParaView >= 5.10, Ubuntu, Anaconda, others

- **https://topology-tool-kit.github.io/examples/**
  - Screenshots
  - Pipeline description
  - ParaView statefile
  - Python script
  - C++/Python API pointers



Conclusion

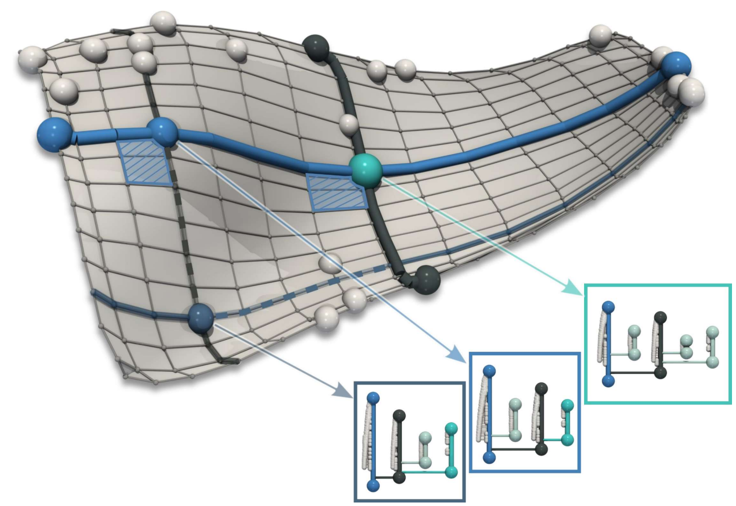
- **Computational framework for Principal Geodesic Analysis**
  - Merge trees
  - Persistence diagrams



Conclusion

- **Computational framework for Principal Geodesic Analysis**
  - Merge trees
  - Persistence diagrams

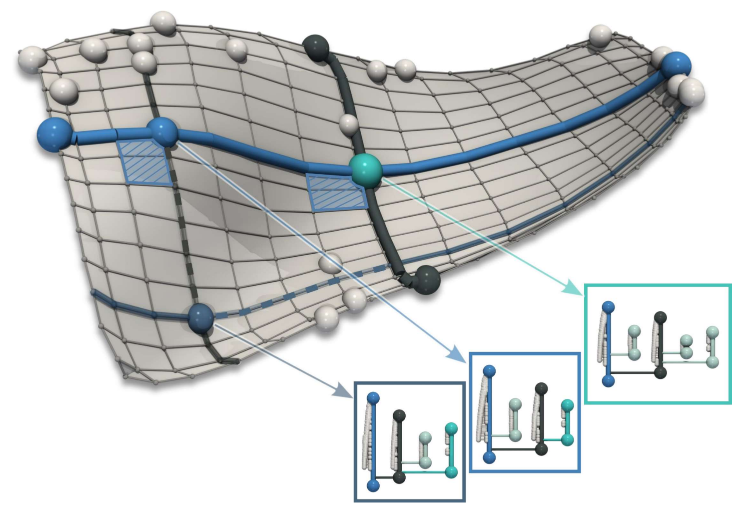
- **Applications to ensemble analysis**



Conclusion

- **Computational framework for Principal Geodesic Analysis**
  - Merge trees
  - Persistence diagrams

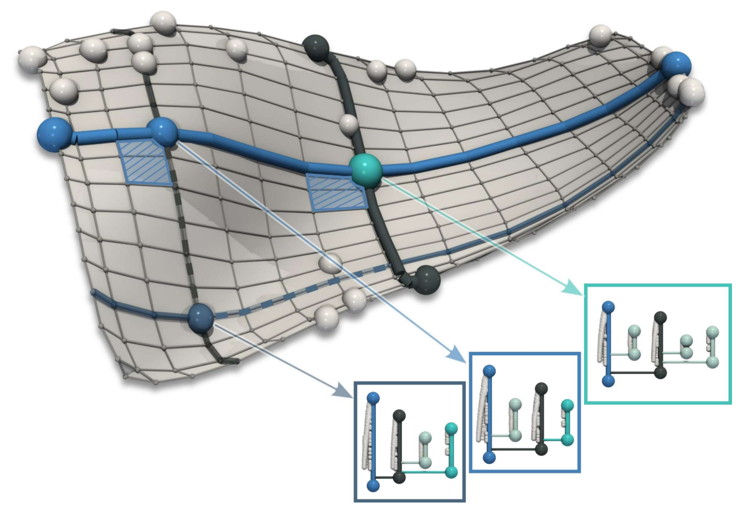
- **Applications to ensemble analysis**



- **Replicable paper**
  - **http://www.replicabilitystamp.org/**
  - Exact code & data

Conclusion

- **Computational framework for Principal Geodesic Analysis**
  - Merge trees
  - Persistence diagrams
- **Applications to ensemble analysis**
- **Replicable paper**
  - **http://www.replicabilitystamp.org/**
  - Exact code & data
- **Perspectives**
  - Other topological descriptors
  - Towards more advanced ensemble analysis



Sylvain GERBAUD

A la recherche d'un poste de MCF – CNU 27 – pour la rentrée de septembre 2025

2020 - 2023 Thèse au laboratoire XLIM à Poitiers, équipe Informatique Graphique

2023 - 2024 ATER – Université de Poitiers

2024 - 2025 Post-Doctorant ERC TORI, LIP6 – Université de la Sorbonne
CV HAL – https://cv.hal.science/sylvain-gerbaud

Coordonnées

Tél. 06 44 88 47 82
Mail. sylvain.gerbaud@lip6.fr

Thématiques de recherche

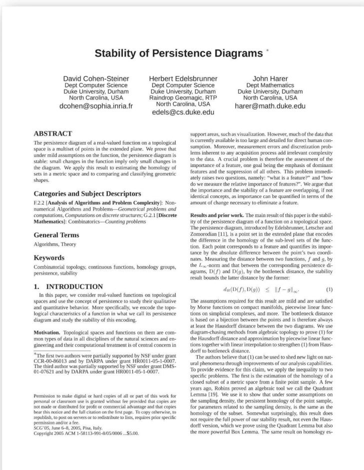
Topologie – Modélisation volumique – HPC – Imagerie Médicale

Thanks!

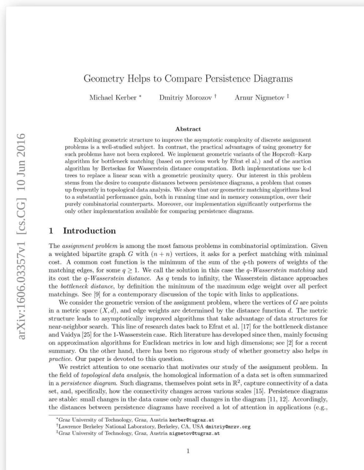
- Curated data
  - https://github.com/MatPont/WassersteinMergeTreesData
- Paper implementation
  - https://github.com/MatPont/MT-PGA
- TTK
  - https://topology-tool-kit.github.io/
  - https://topology-tool-kit.github.io/examples/
  - https://github.com/topology-tool-kit/ttk/
- ERC Project TORI
  - https://erc-tori.github.io/
  - We're hiring!!!

Metric spaces for Persistence diagrams

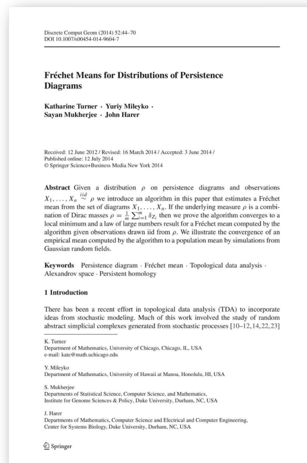
- Vast literature!


Stability of Persistence Diagrams
David Cohen-Steiner, Herbert Edelsbrunner, John Harer

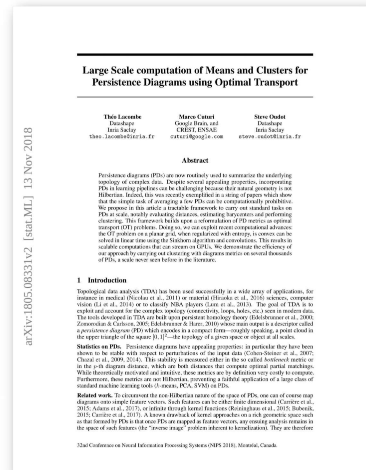
[Cohen-Steiner '05]


Geometry Helps to Compare Persistence Diagrams
Michael Kerber, Dmitriy Morozov, Arnur Nigmetov

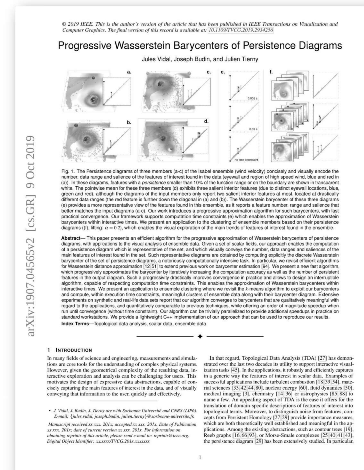
[Kerber '16]


Fréchet Means for Distributions of Persistence Diagrams
Katharine Turner · Yuriy Mileyko · Sayan Mukherjee · John Harer

[Turner '14]

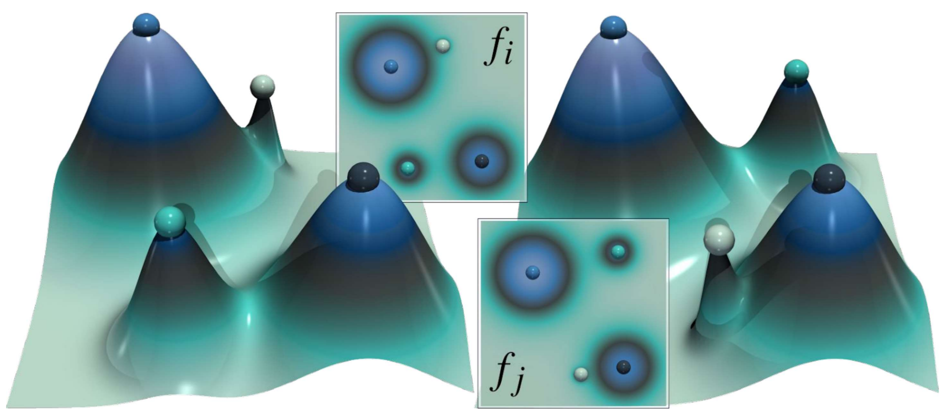

Large Scale computation of Means and Clusters for Persistence Diagrams using Optimal Transport
Théo Lacombe, Marco Cuturi, Steve Oudot

[Lacombe '18]

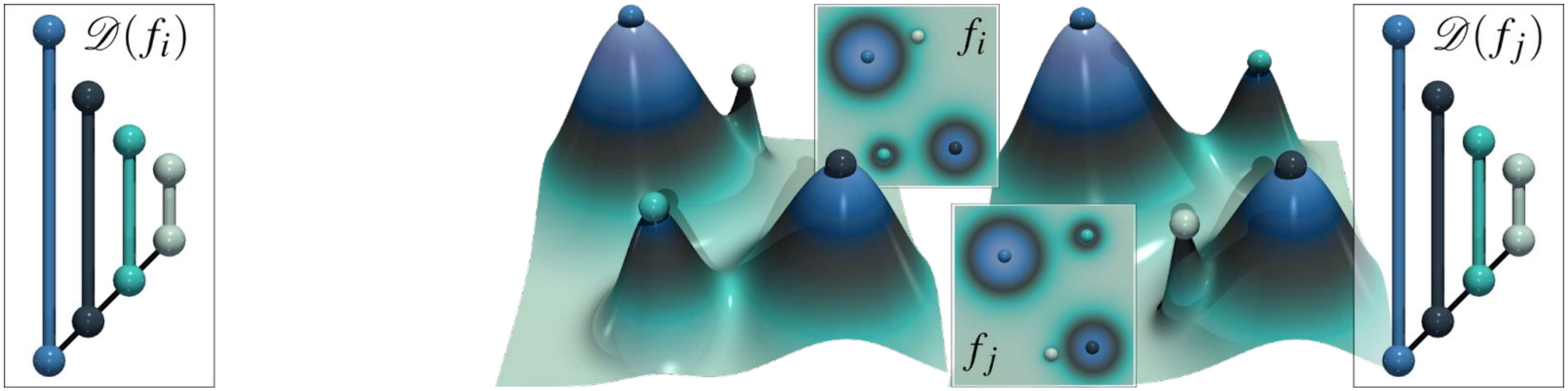

Progressive Wasserstein Barycenters of Persistence Diagrams
Jules Vidal, Joseph Budin, and Julien Tierny

[Vidal '19]

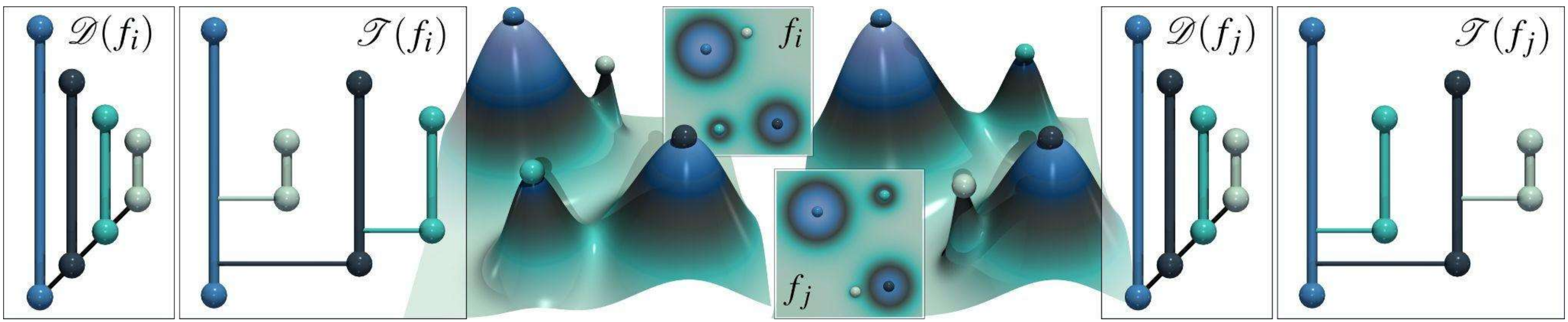
Limitations of Persistence diagrams



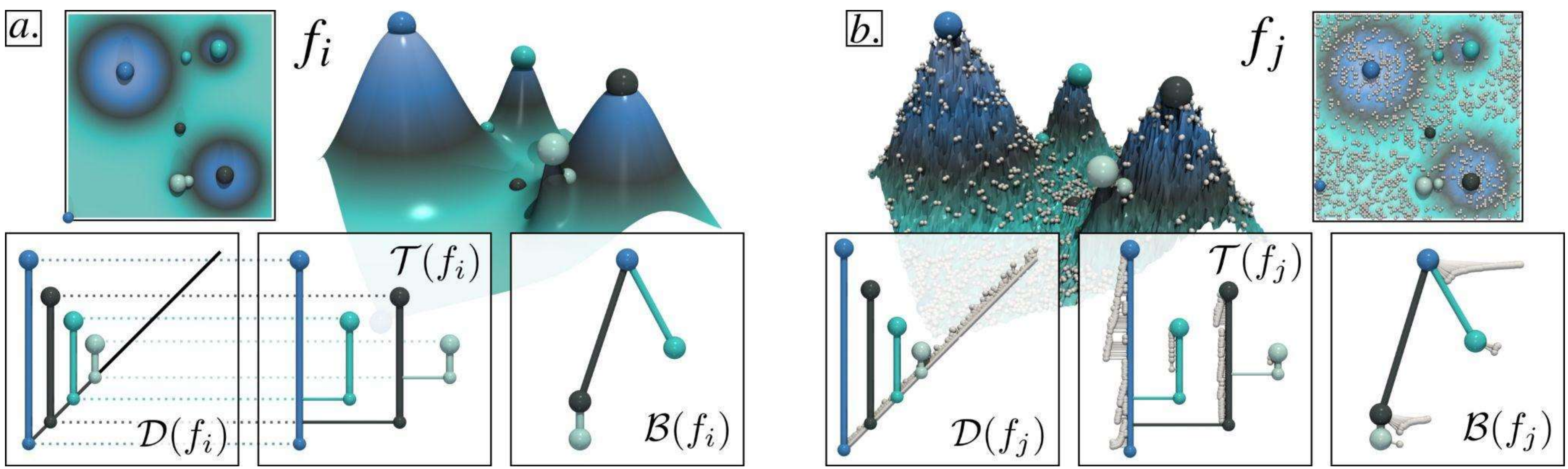
Limitations of Persistence diagrams



Limitations of Persistence diagrams

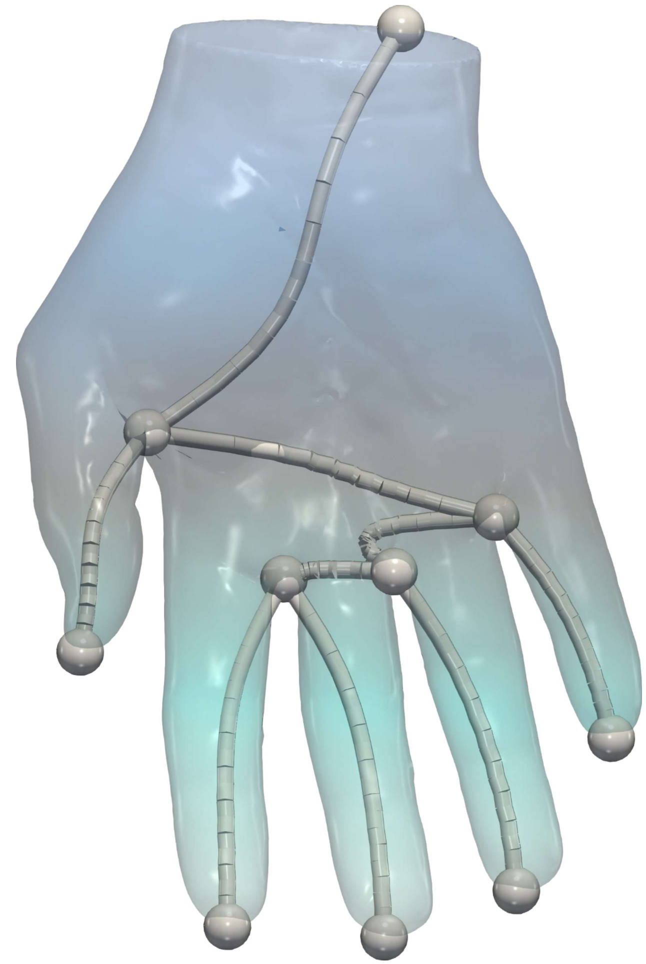


PDs, MTs, BDTs

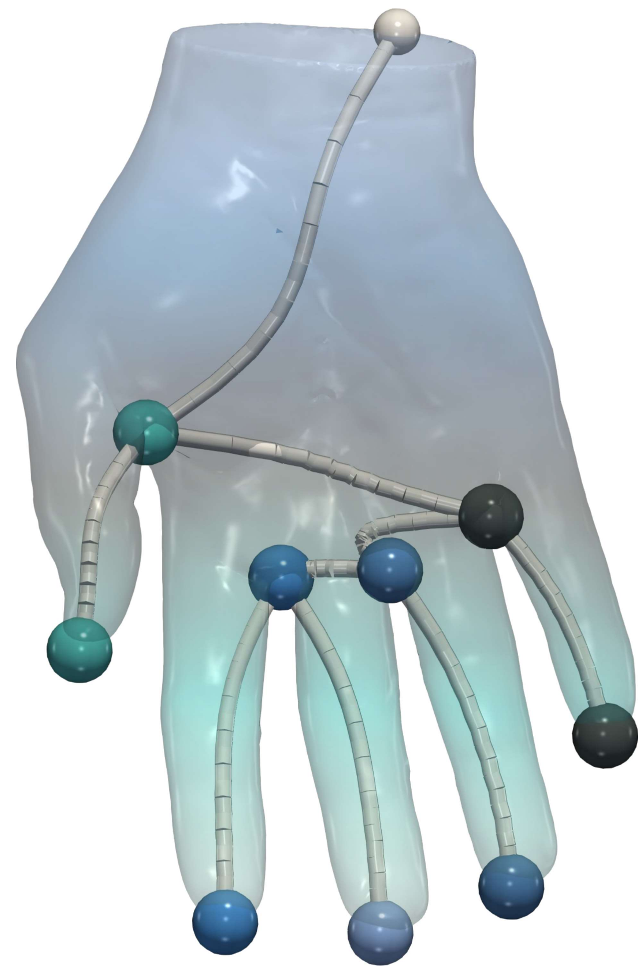


a.
f_i
$\mathcal{T}(f_i)$
$\mathcal{D}(f_i)$
$\mathcal{B}(f_i)$
b.
f_j
$\mathcal{T}(f_j)$
$\mathcal{D}(f_j)$
$\mathcal{B}(f_j)$

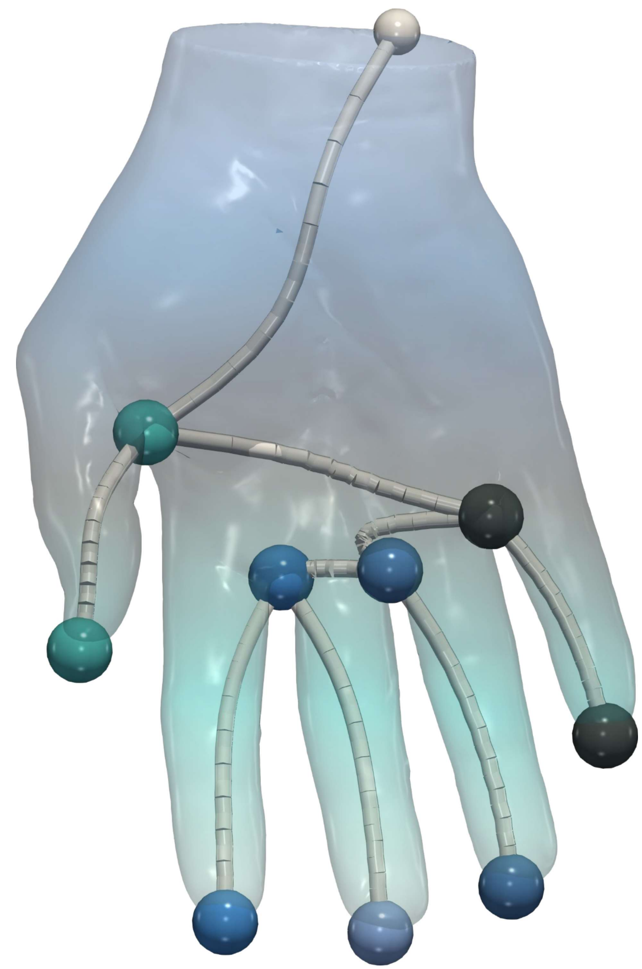
Branch decomposition tree (BDT)



Branch decomposition tree (BDT)

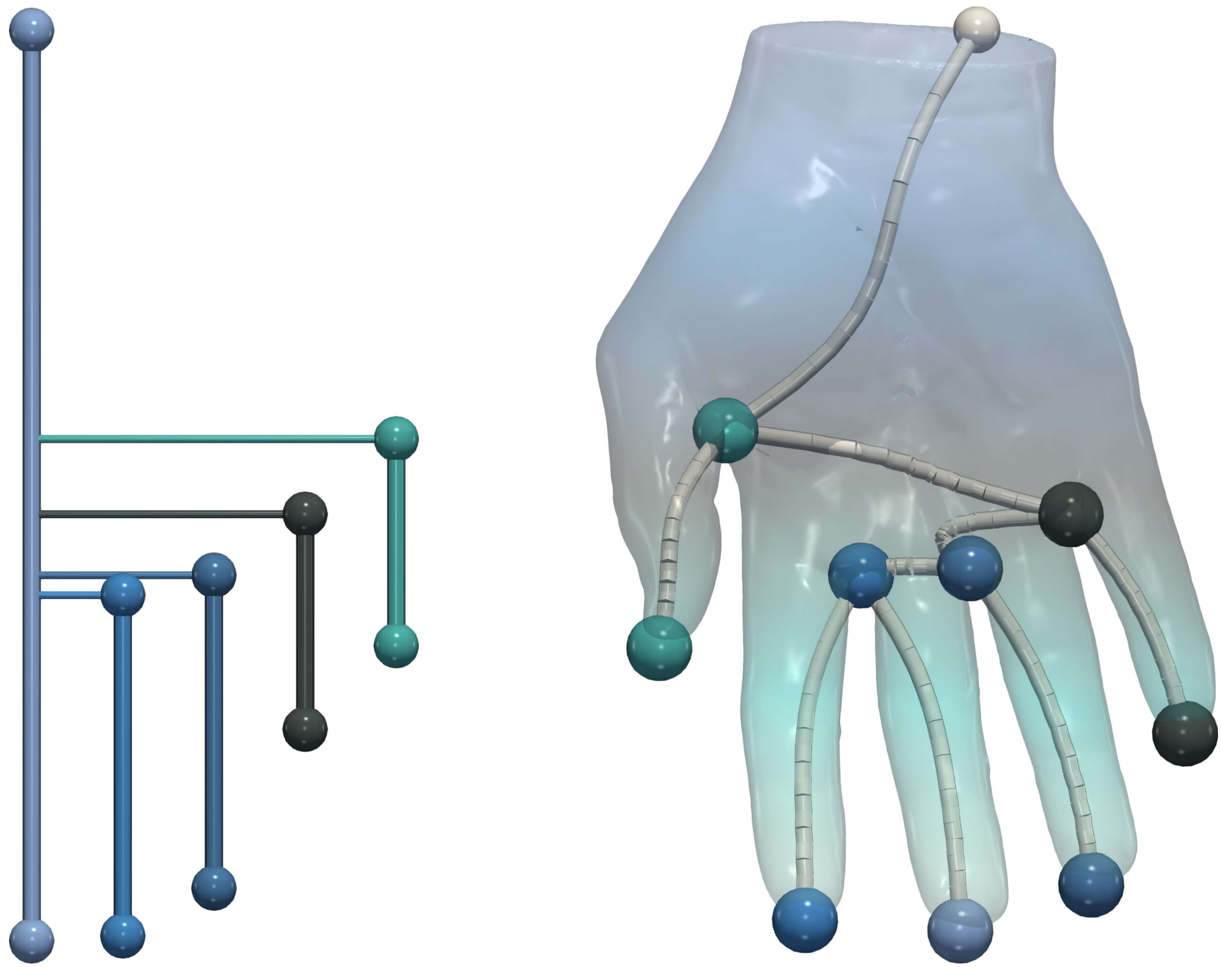


Branch decomposition tree (BDT)

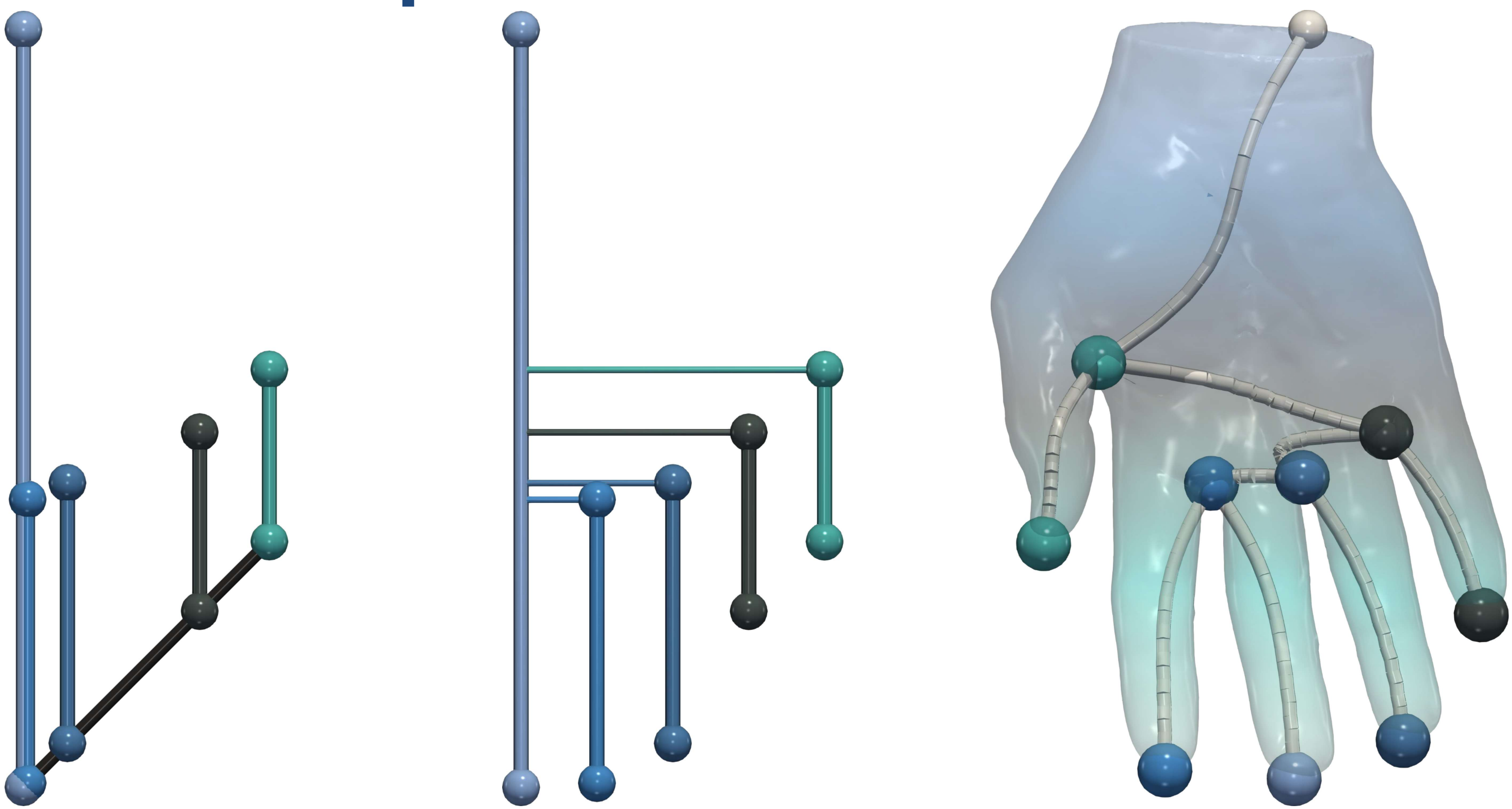


- **Persistent branch**
  - Monotonic path
  - From a minimum
  - Up to its persistence-paired saddle

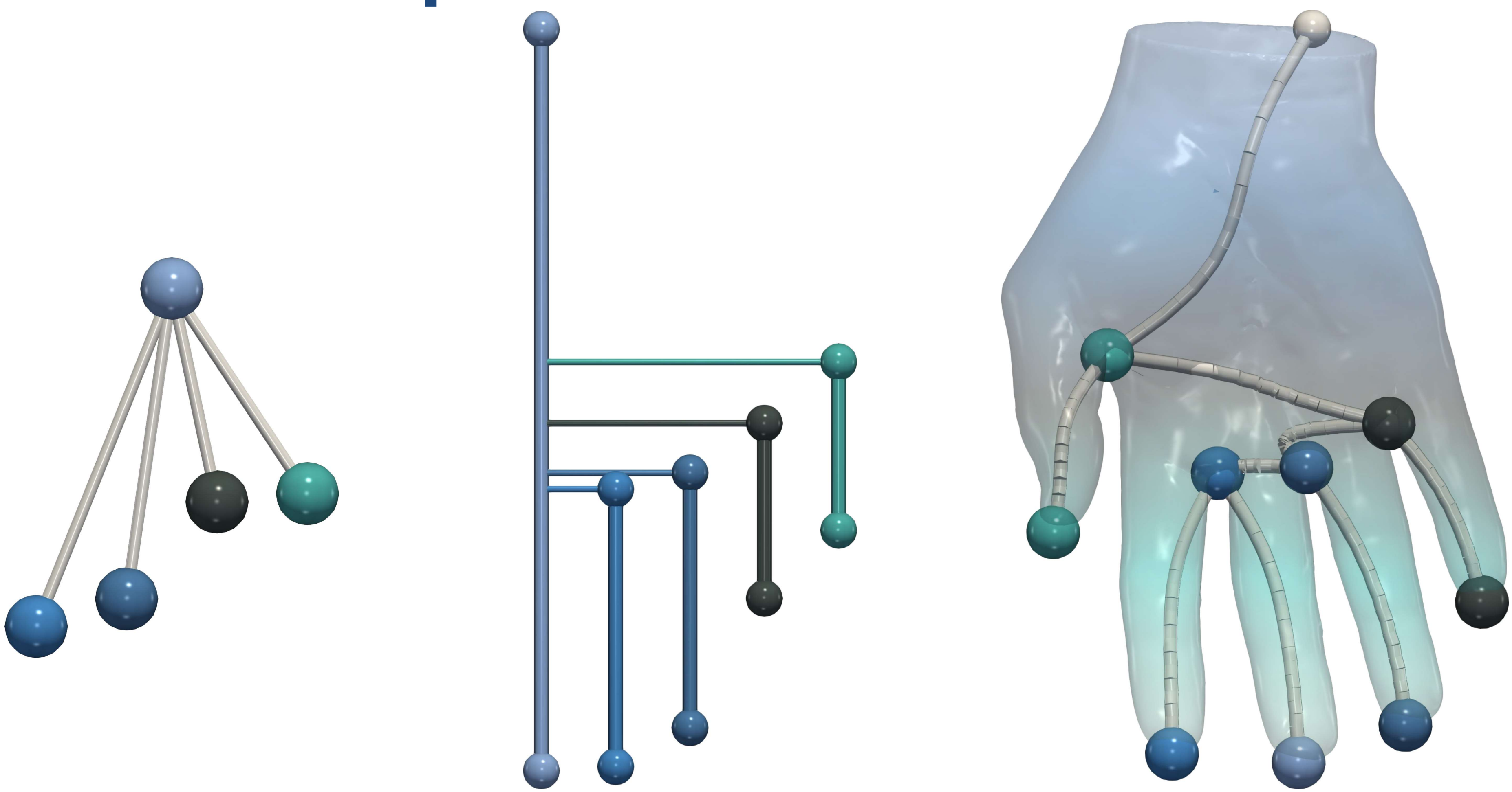
Branch decomposition tree (BDT)



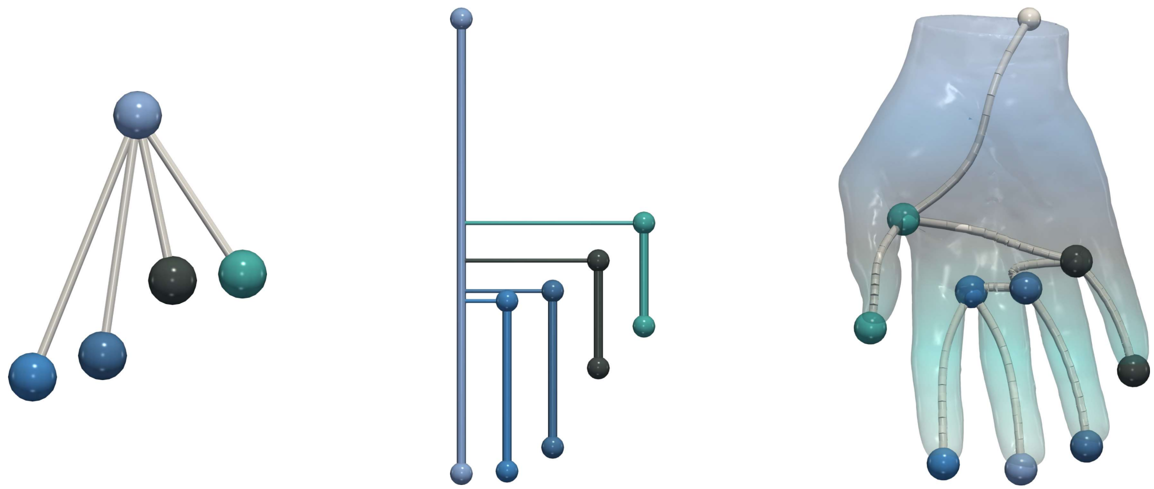
Branch decomposition tree (BDT)



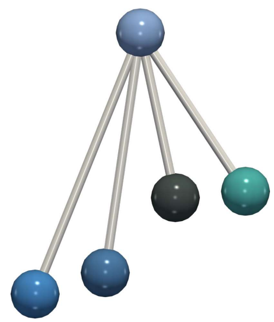
Branch decomposition tree (BDT)

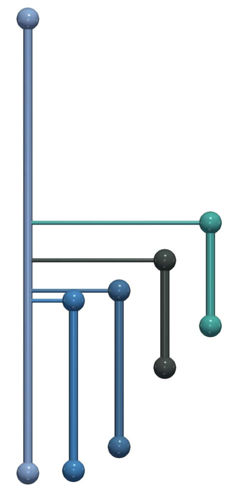


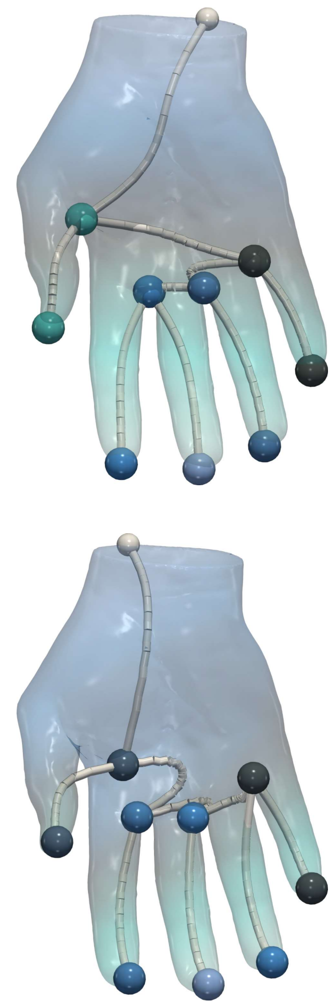
BDTs in practice



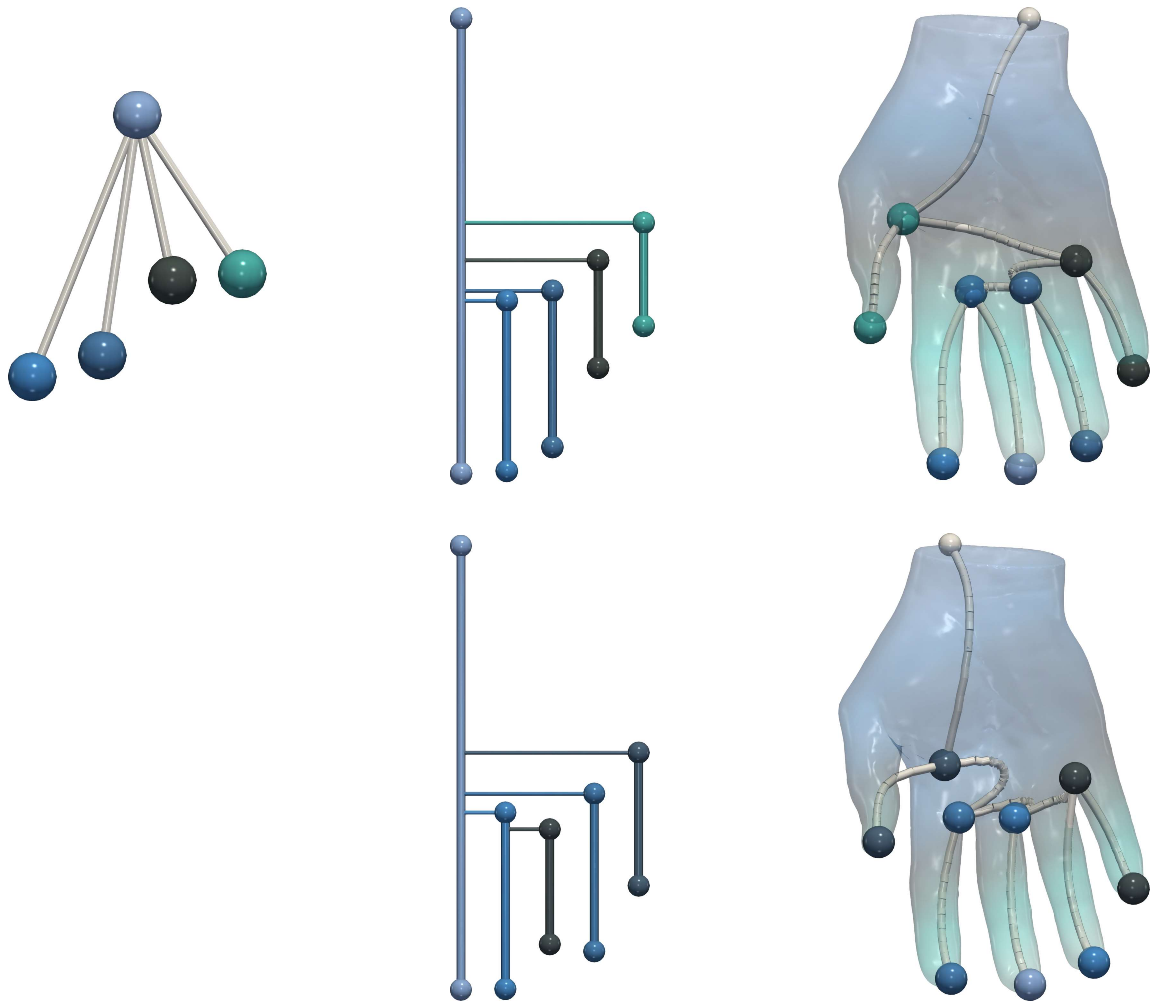
BDTs in practice



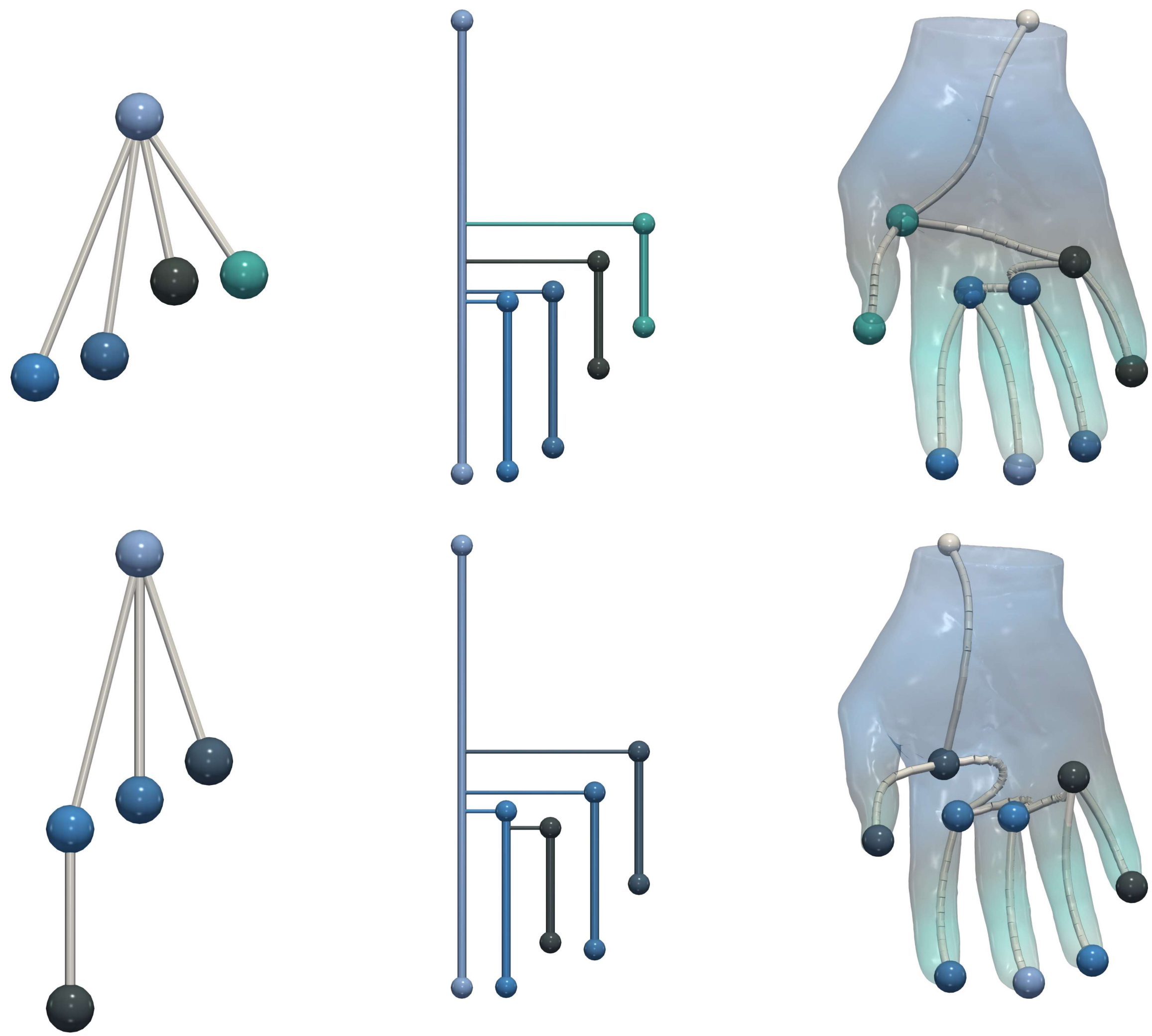




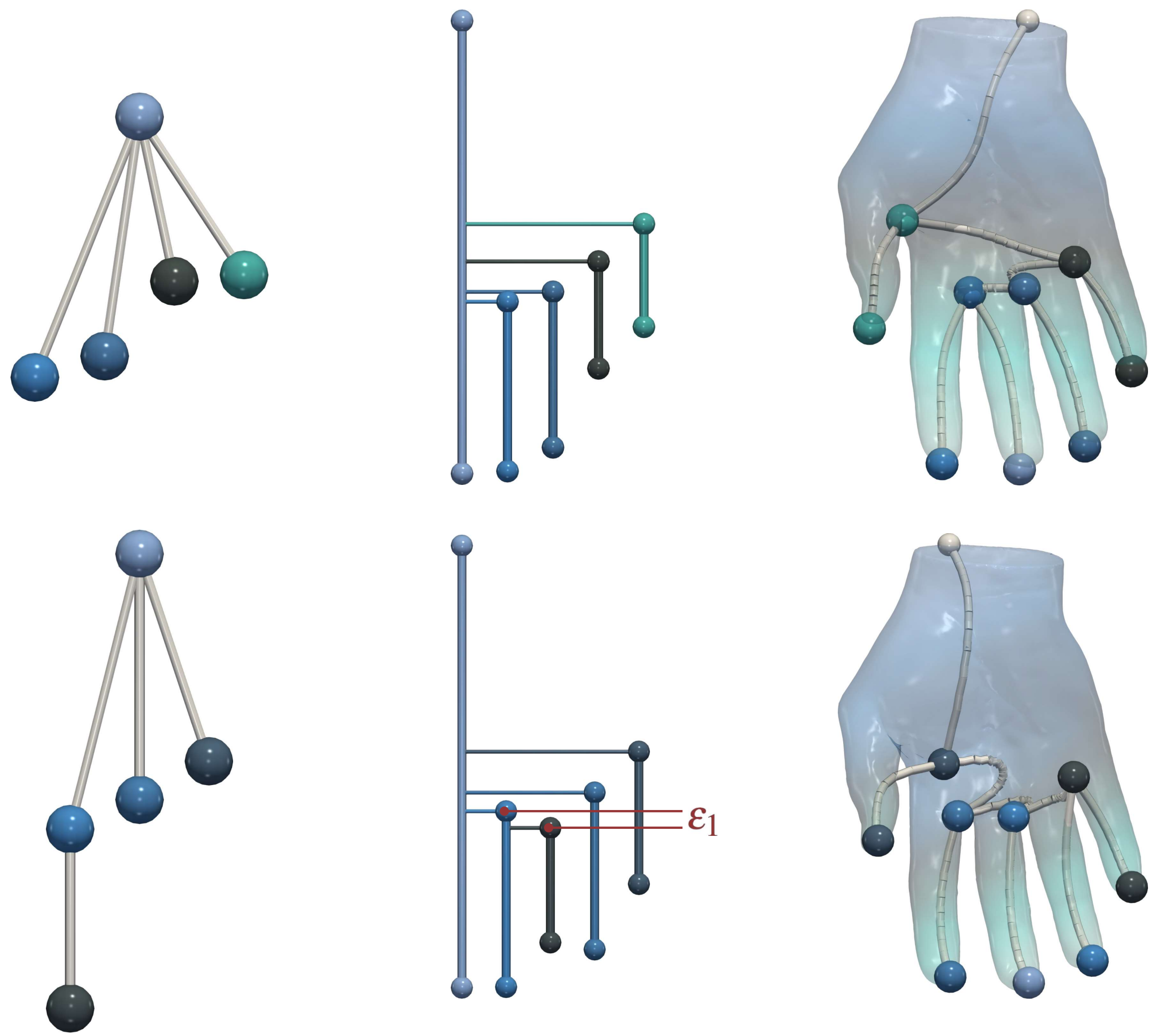
BDTs in practice



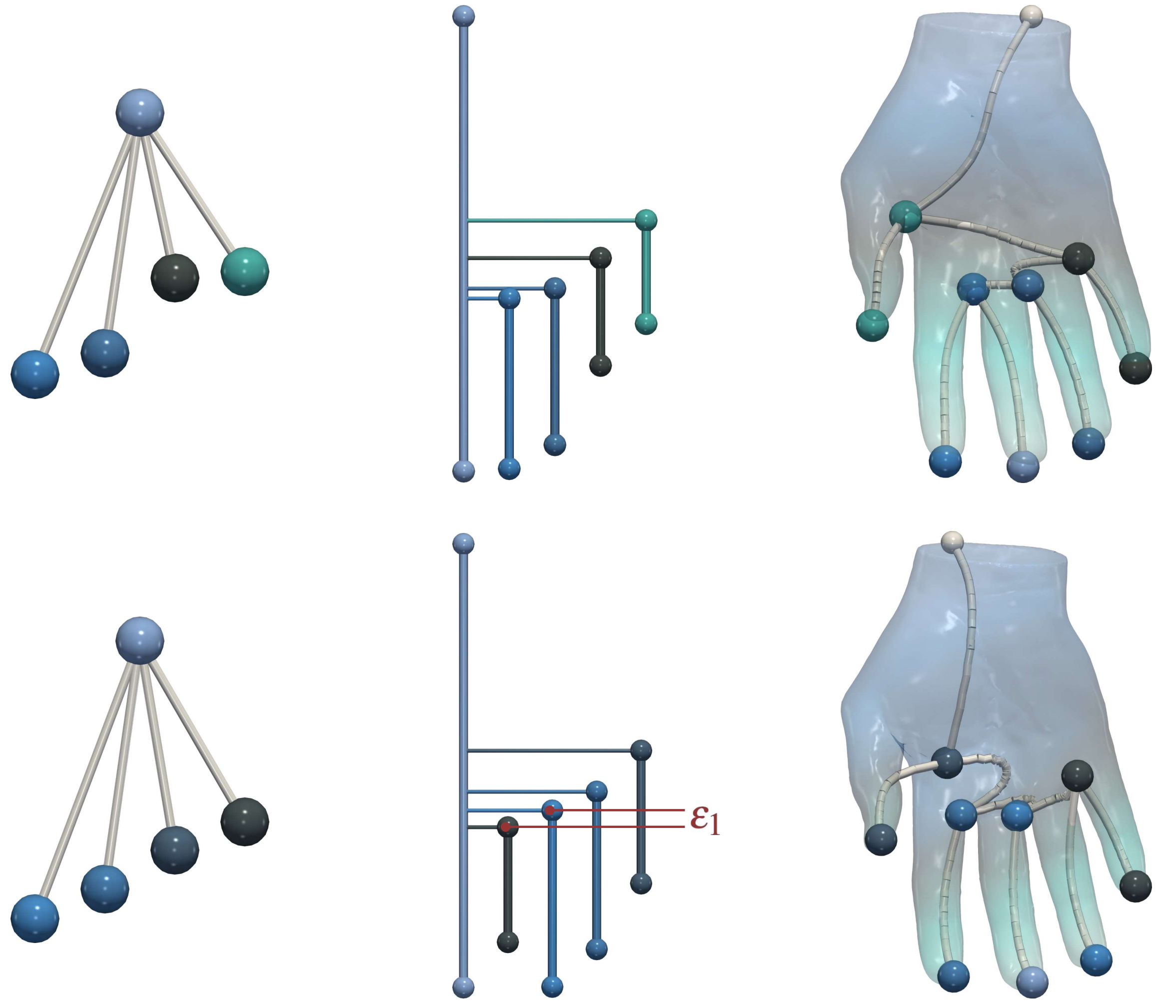
BDTs in practice



BDTs in practice



BDTs in practice



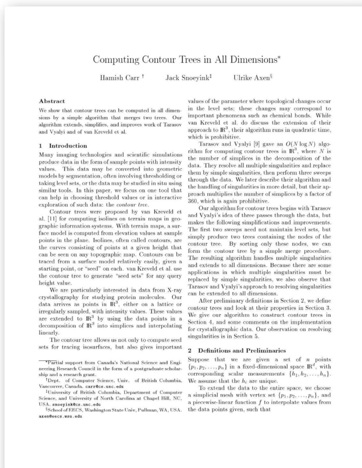
Merge trees in Visualization

- Vast literature!

Construction of Contour Trees in 3D in $O(n \log n)$ steps

Sergey P. Tarasov, Michael N. Vyalyi

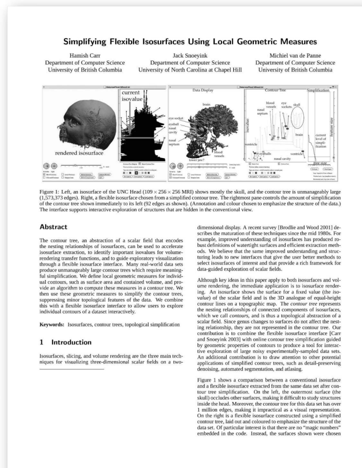
[Tarasov '98]



Computing Contour Trees in All Dimensions

Hamish Carr, Jack Snoeyink, Ulrike Axen

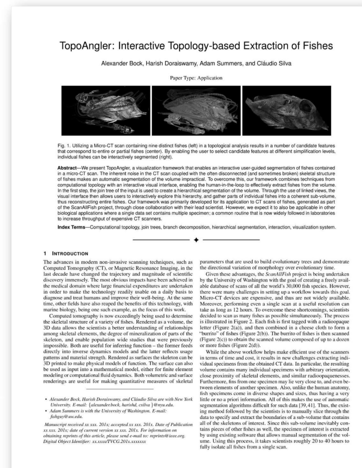
[Carr '00]



Simplifying Flexible Isosurfaces Using Local Geometric Measures

Hamish Carr, Jack Snoeyink, Michiel van de Panne

[Carr '04]



TopoAngler: Interactive Topology-based Extraction of Fishes

Alexander Bock, Harish Doraiswamy, Adam Summers, and Cláudio Silva

[Bock '17]

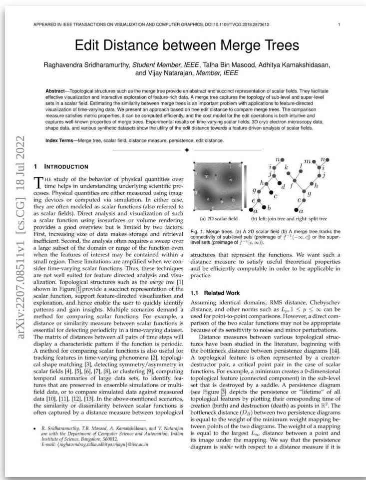
Task-based Augmented Contour Trees with Fibonacci Heaps

C. Gueunet, P. Fortin, J. Jomier and J. Tierny

[Gueunet '19]

Merge trees for ensemble analysis

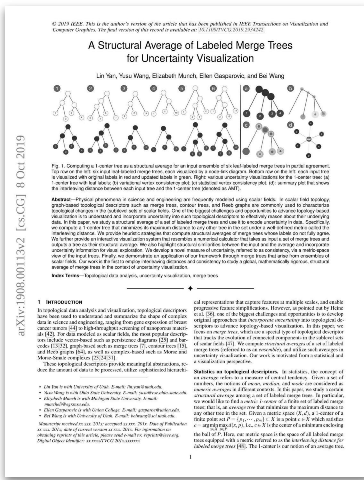
- Recent works



Edit Distance between Merge Trees

Raghavendra Sridharamurthy, *Student Member, IEEE*, Talha Bin Masood, Adhitya Kamakshidasan, and Vijay Natarajan, *Member, IEEE*

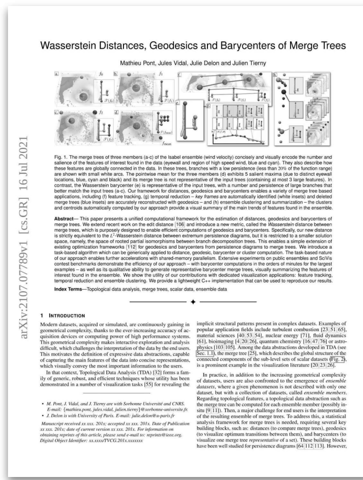
[Sridharamurthy '18]



A Structural Average of Labeled Merge Trees for Uncertainty Visualization

Lin Yan, Yusu Wang, Elizabeth Munch, Ellen Gasparovic, and Bei Wang

[Yan '19]



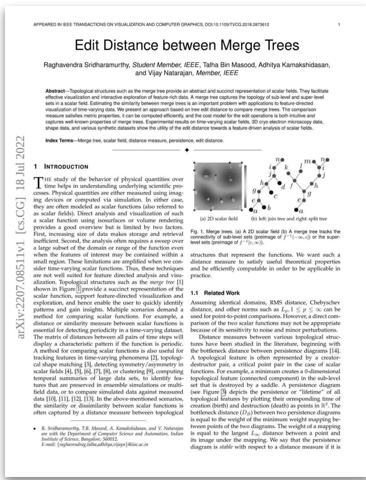
Wasserstein Distances, Geodesics and Barycenters of Merge Trees

Mathieu Pont, Jules Vidal, Julie Delon and Julien Tierny

[Pont '21]

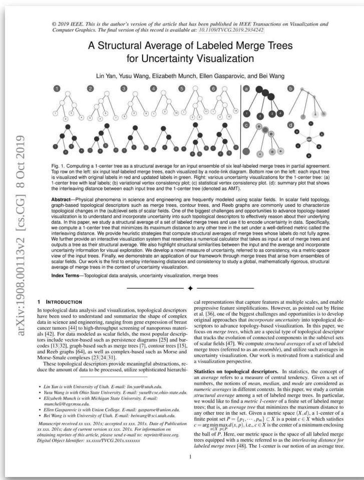
Merge trees for ensemble analysis

- Recent works


Edit Distance between Merge Trees

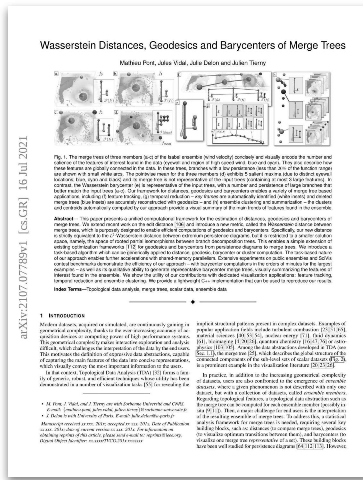
Raghavendra Sridharamurthy, Student Member, IEEE, Talha Bin Masood, Adhitya Kamakshidasan, and Vijay Natarajan, Member, IEEE

[Sridharamurthy '18]


A Structural Average of Labeled Merge Trees for Uncertainty Visualization

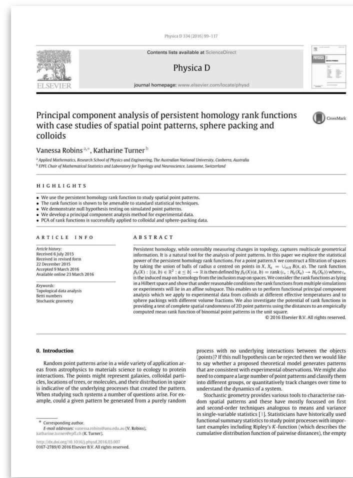
Lin Yan, Yusu Wang, Elizabeth Munch, Ellen Gasparovic, and Bei Wang

[Yan '19]


Wasserstein Distances, Geodesics and Barycenters of Merge Trees

Mathieu Pont, Jules Vidal, Julie Delon and Julien Tierny

[Pont '21]


Physica D

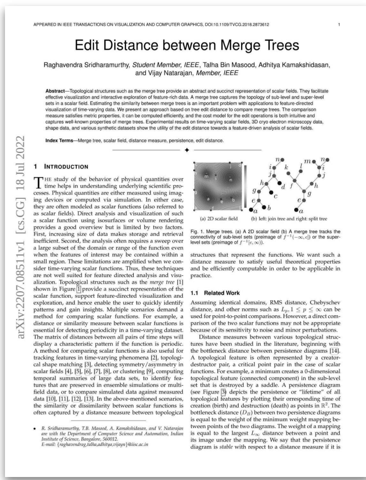
Principal component analysis of persistent homology rank functions with case studies of spatial point patterns, sphere packing and colloids

Vanessa Robins, Katharine Turner

[Robins '16]

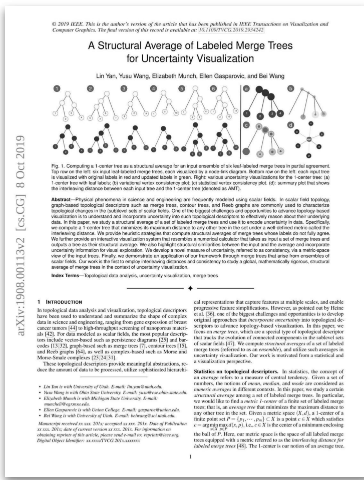
Merge trees for ensemble analysis

- Recent works


Edit Distance between Merge Trees

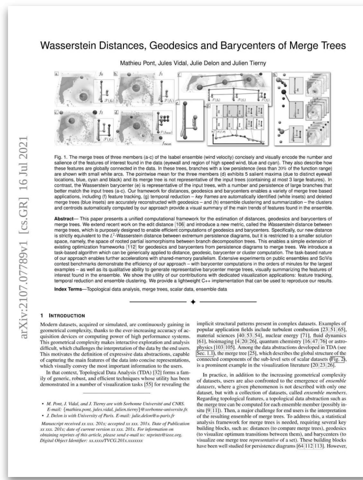
Raghavendra Sridharamurthy, Student Member, IEEE, Talha Bin Masood, Adhitya Kamakshidasan, and Vijay Natarajan, Member, IEEE

[Sridharamurthy '18]


A Structural Average of Labeled Merge Trees for Uncertainty Visualization

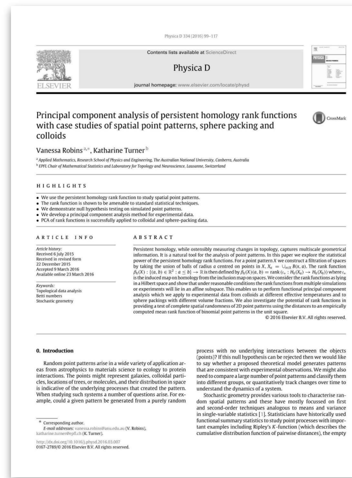
Lin Yan, Yusu Wang, Elizabeth Munch, Ellen Gasparovic, and Bei Wang

[Yan '19]


Wasserstein Distances, Geodesics and Barycenters of Merge Trees

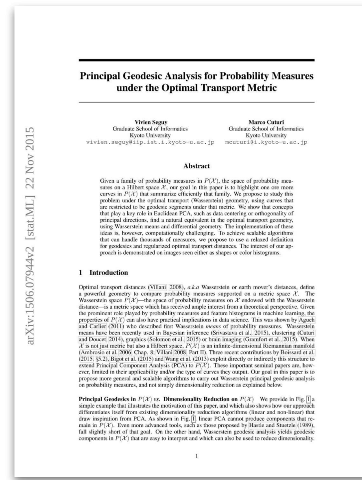
Mathieu Pont, Jules Vidal, Julie Delon and Julien Tierny

[Pont '21]


Principal component analysis of persistent homology rank functions with case studies of spatial point patterns, sphere packing and colloids

Vanessa Robins, Katharine Turner

[Robins '16]

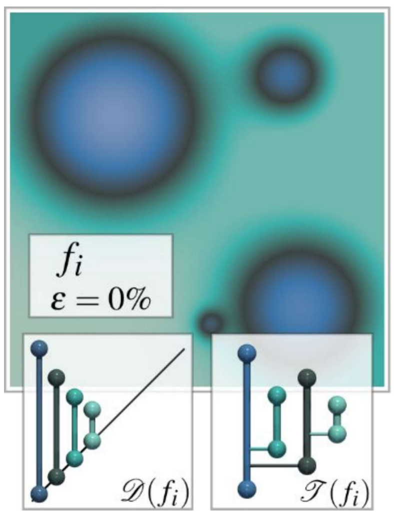

Principal Geodesic Analysis for Probability Measures under the Optimal Transport Metric

Vivien Seguy, Marco Cuturi

[Seguy '15]

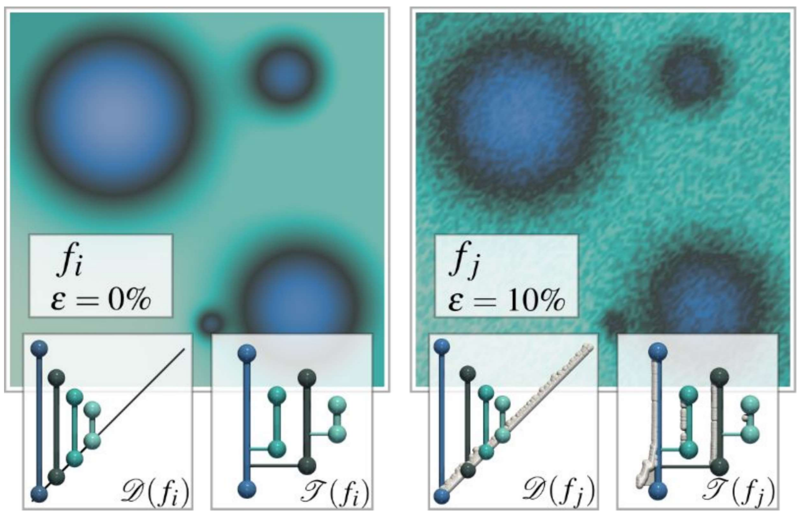
Evaluation

- **Stability**



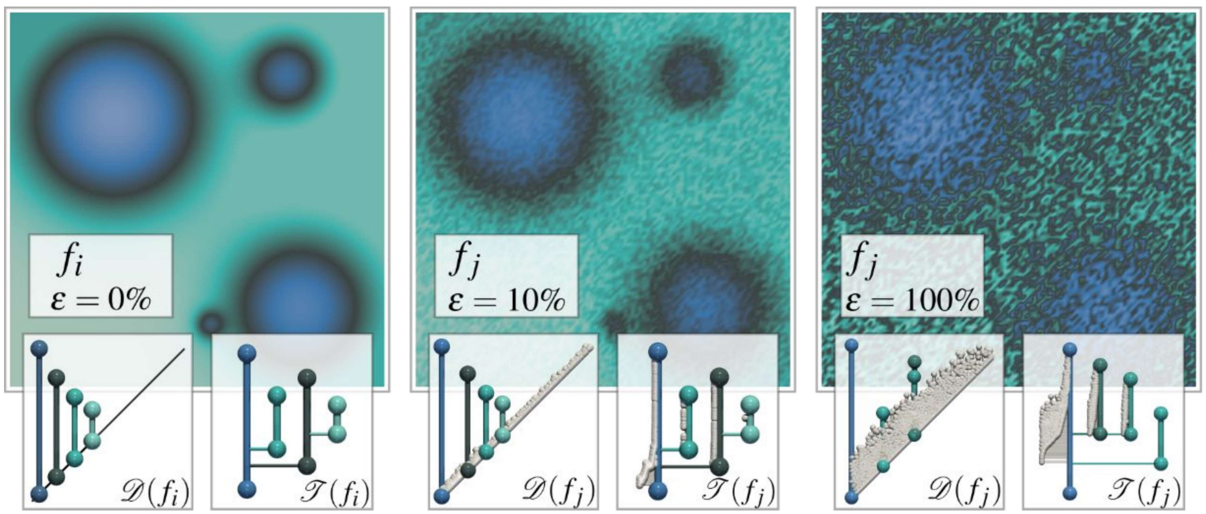
Evaluation

- **Stability**



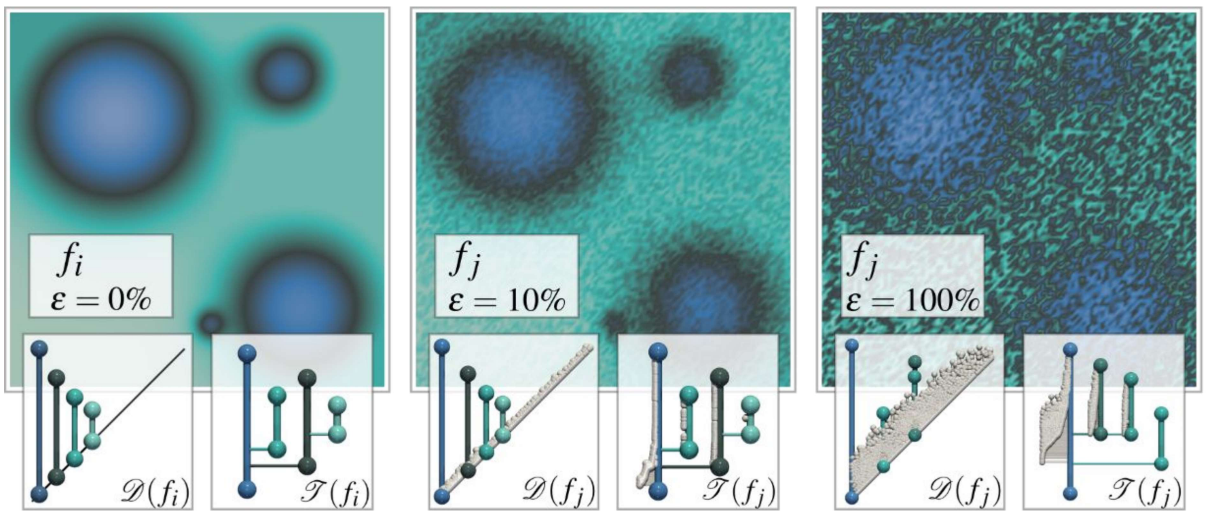
Evaluation

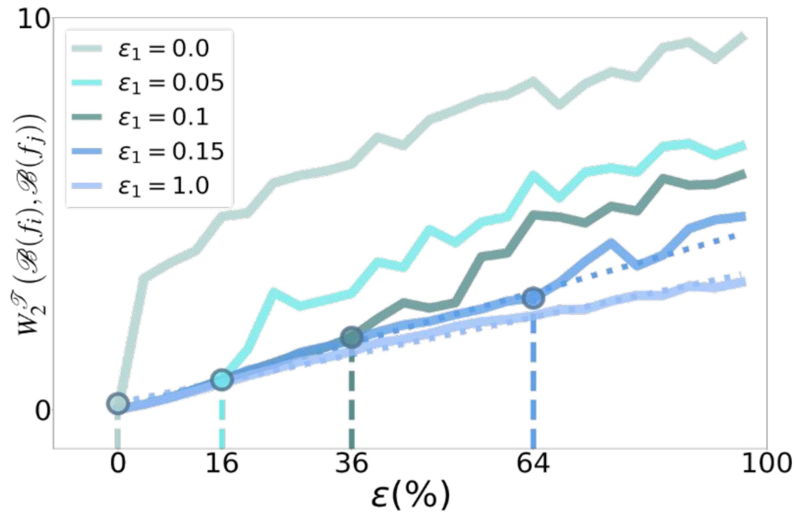
- **Stability**



Evaluation

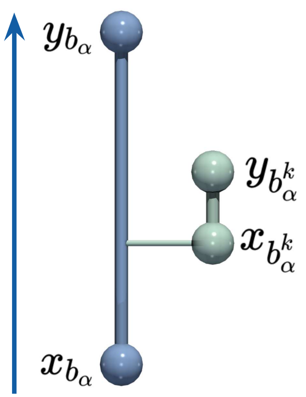
- **Stability**





Local normalization

- **Nesting condition**
  - $[x^k_{b_\alpha}, y^k_{b_\alpha}] \subseteq [x_{b_\alpha}, y_{b_\alpha}]$



Local normalization

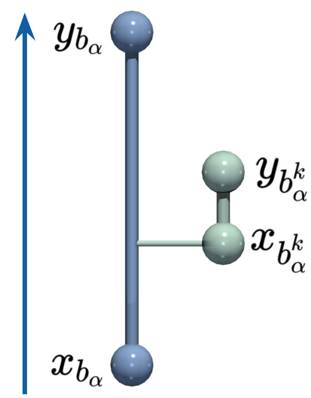
- **Nesting condition**
  - $[x^k_{b_\alpha}, y^k_{b_\alpha}] \subseteq [x_{b_\alpha}, y_{b_\alpha}]$

- **Strategy**
  - Local normalization
  - Distance/Interpolation
  - Normalization reversal

$$\mathcal{N}(b_i{}^k) = \left(\mathcal{N}_x(b_i{}^k), \mathcal{N}_y(b_i{}^k)\right)$$

$$\mathcal{N}_x(b_i{}^k) = (x_{b_i{}^k} - x_{b_i})/(y_{b_i} - x_{b_i})$$

$$\mathcal{N}_y(b_i{}^k) = (y_{b_i{}^k} - x_{b_i})/(y_{b_i} - x_{b_i})$$



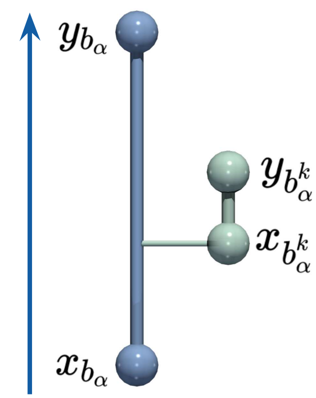
Local normalization

- **Nesting condition**
  - $[x^k_{b_\alpha}, y^k_{b_\alpha}] \subseteq [x_{b_\alpha}, y_{b_\alpha}]$
- **Strategy**
  - Local normalization
  - Distance/Interpolation
  - Normalization reversal

$$\mathcal{N}({b_i}^k) = \left(\mathcal{N}_x({b_i}^k), \mathcal{N}_y({b_i}^k)\right)$$

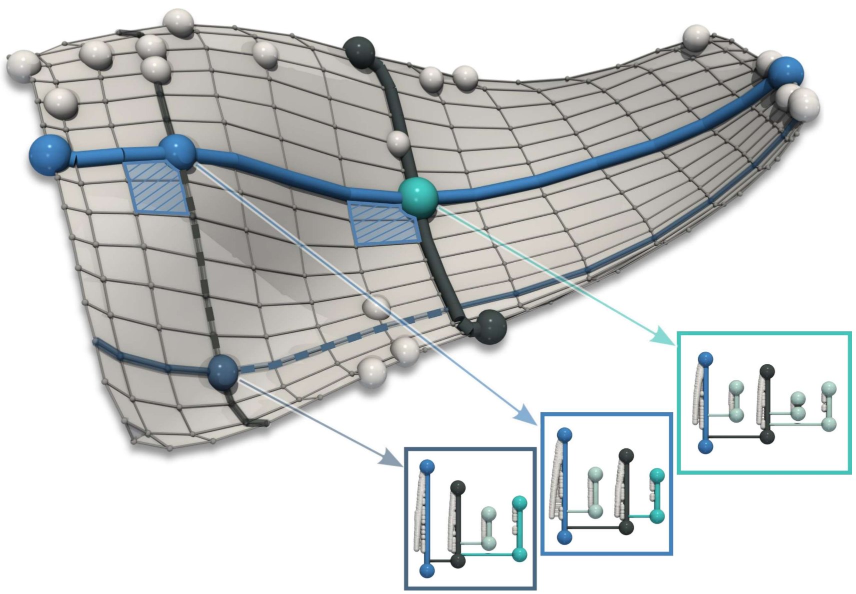
$$\mathcal{N}_x({b_i}^k) = (x_{{b_i}^k} - x_{b_i})/(y_{b_i} - x_{b_i})$$

$$\mathcal{N}_y({b_i}^k) = (y_{{b_i}^k} - x_{b_i})/(y_{b_i} - x_{b_i})$$



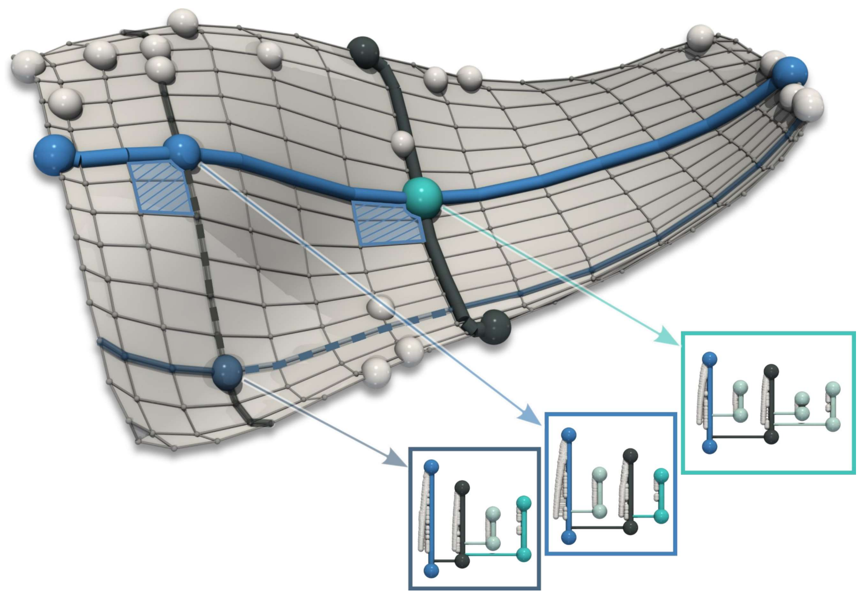
- **BDT pre-process**
  - Relative persistence < ε_3
  - Until ε_2 of their parents
  - Default: $\varepsilon_1 = 0.05$, $\varepsilon_2 = 0.95$, $\varepsilon_3 = 0.9$

From PCA to MT-PGA



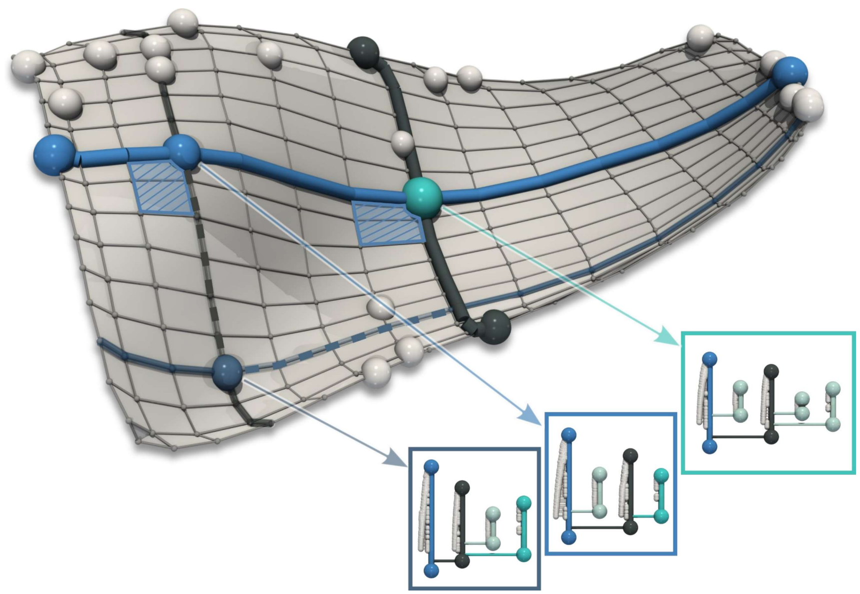
From PCA to MT-PGA

- Required low-level tools

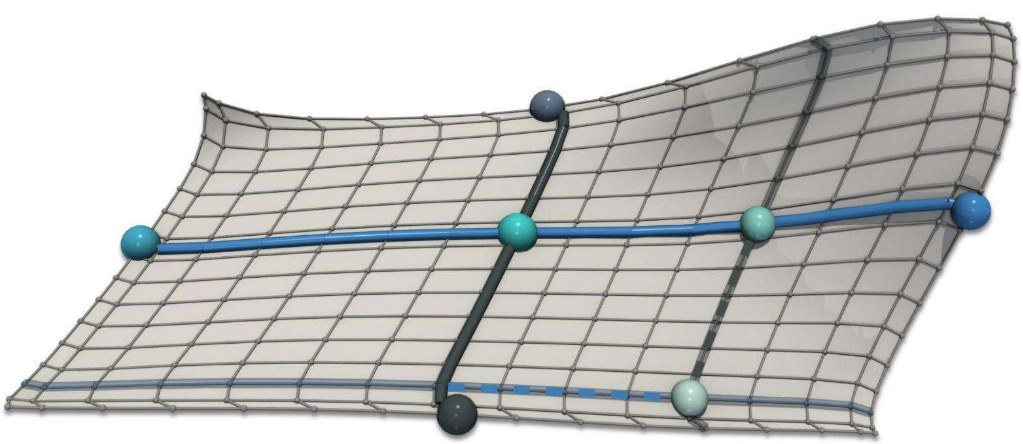


From PCA to MT-PGA

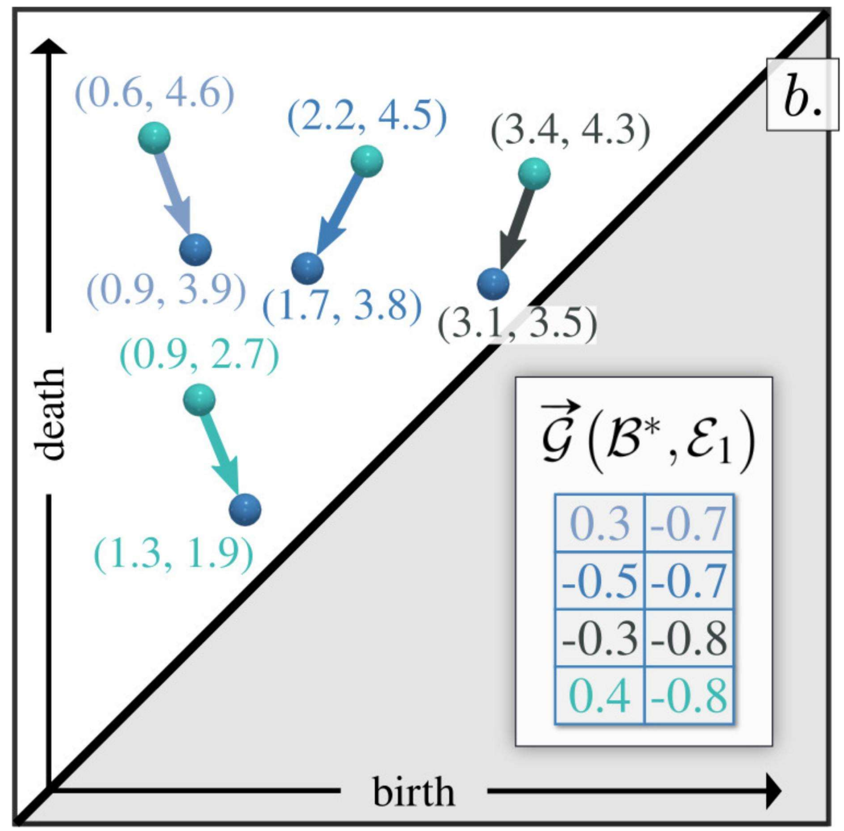
- **Required low-level tools**
  - Geodesic
  - Orthogonal geodesics
  - Collinear geodesics
  - Geodesic axis
  - Axis projection
  - Orthogonal axes
  - Axis translation
  - Orthogonal basis



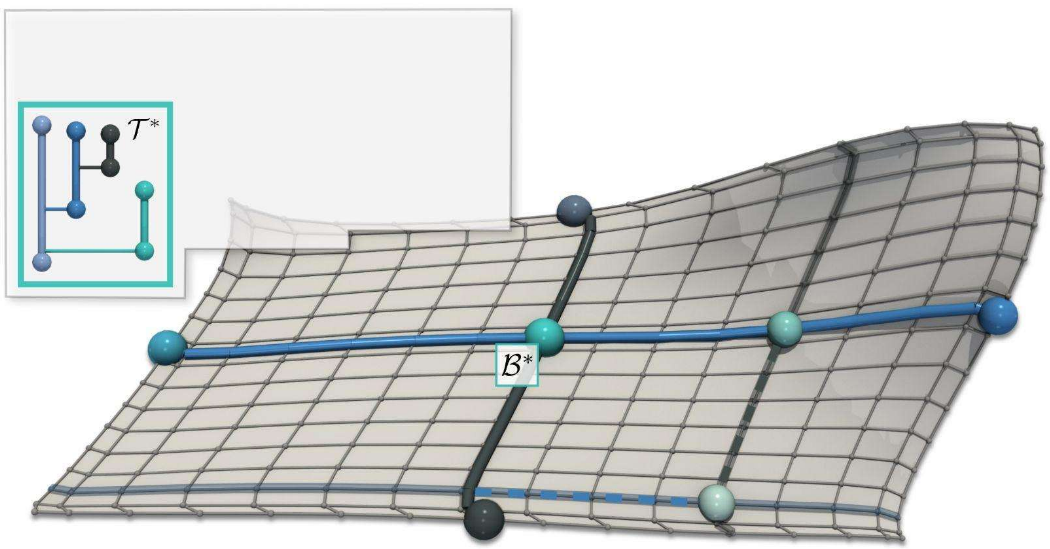
Geodesics



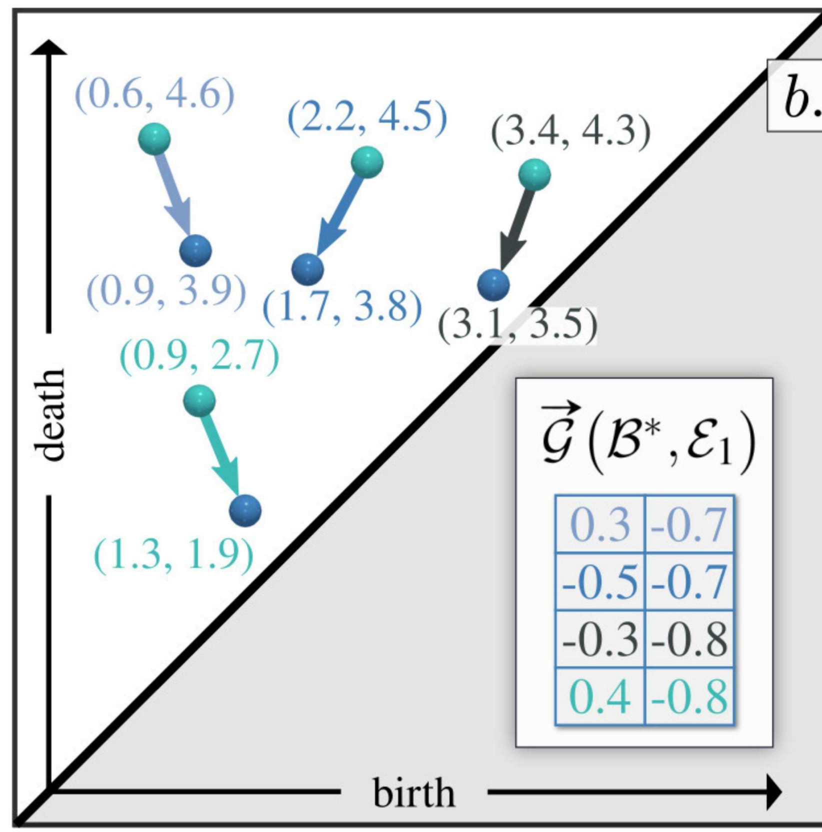
$$\vec{\mathcal{G}}(\mathcal{E},\mathcal{E}') \in \mathbb{R}^{2\times|\mathcal{E}|}$$



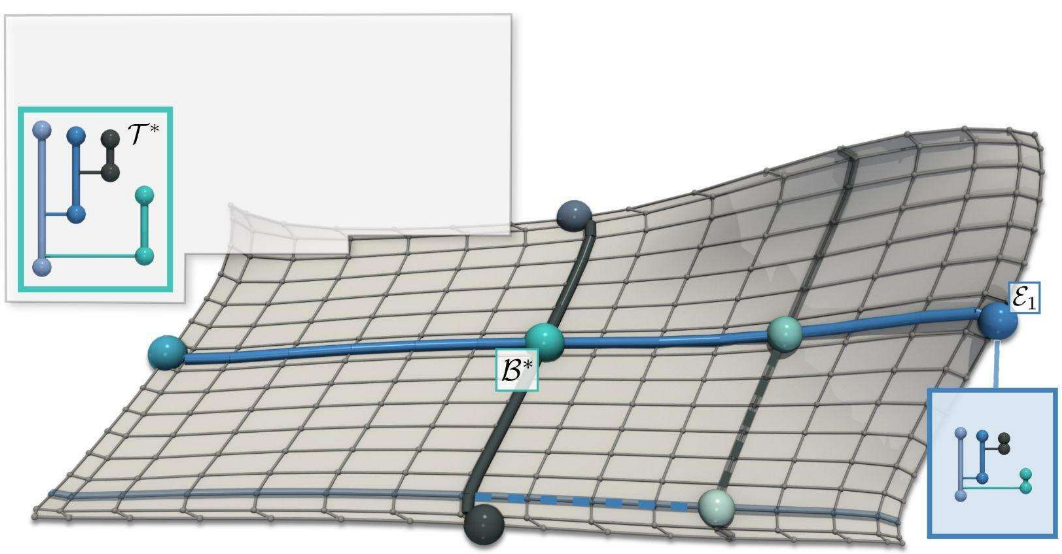
Geodesics



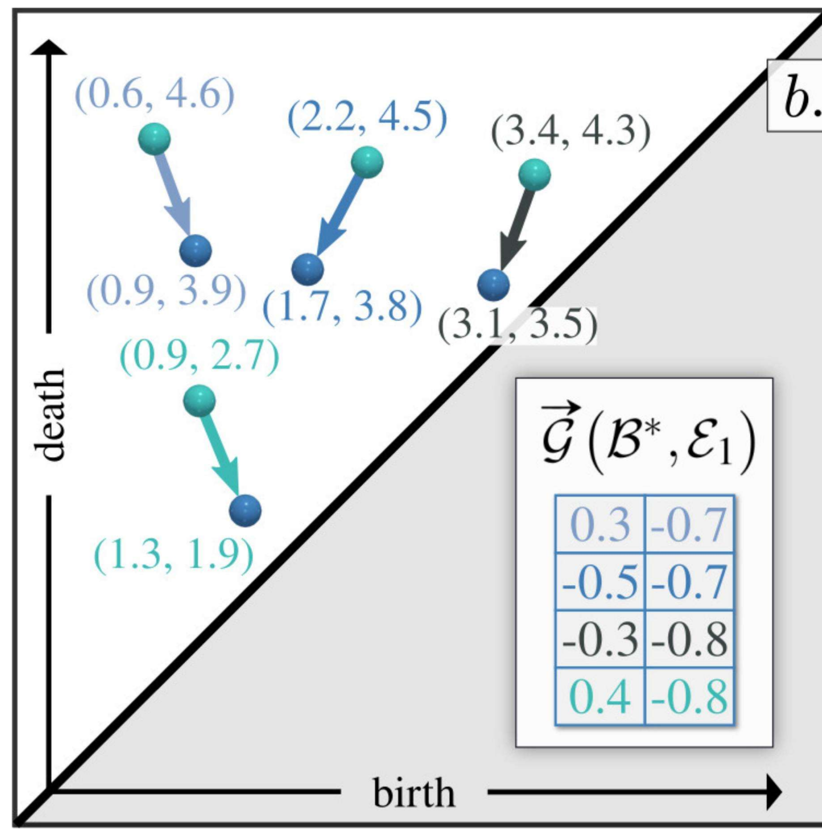
$$\vec{\mathcal{G}}\left(\mathcal{E},\mathcal{E}'\right) \in \mathbb{R}^{2\times|\mathcal{E}|}$$



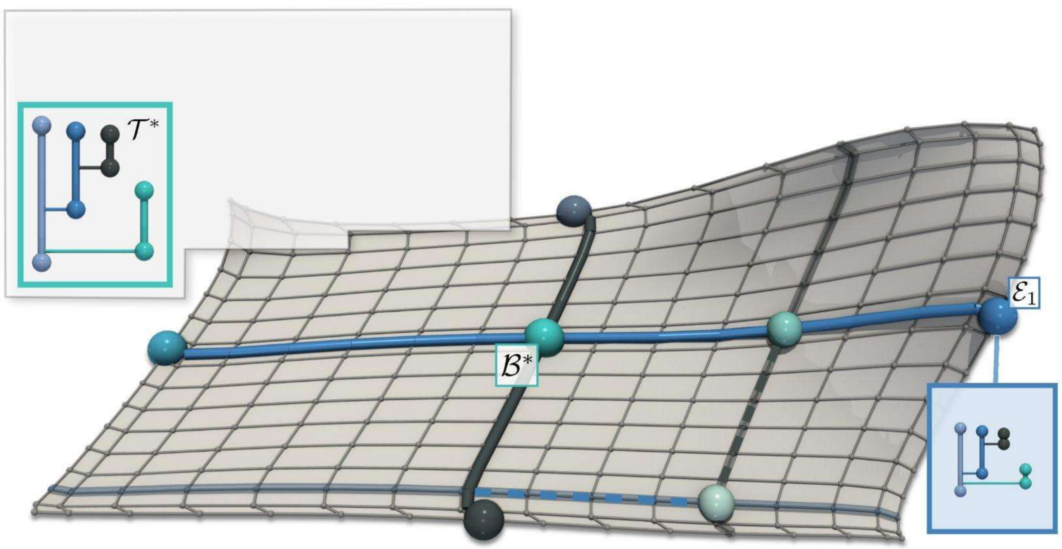
Geodesics



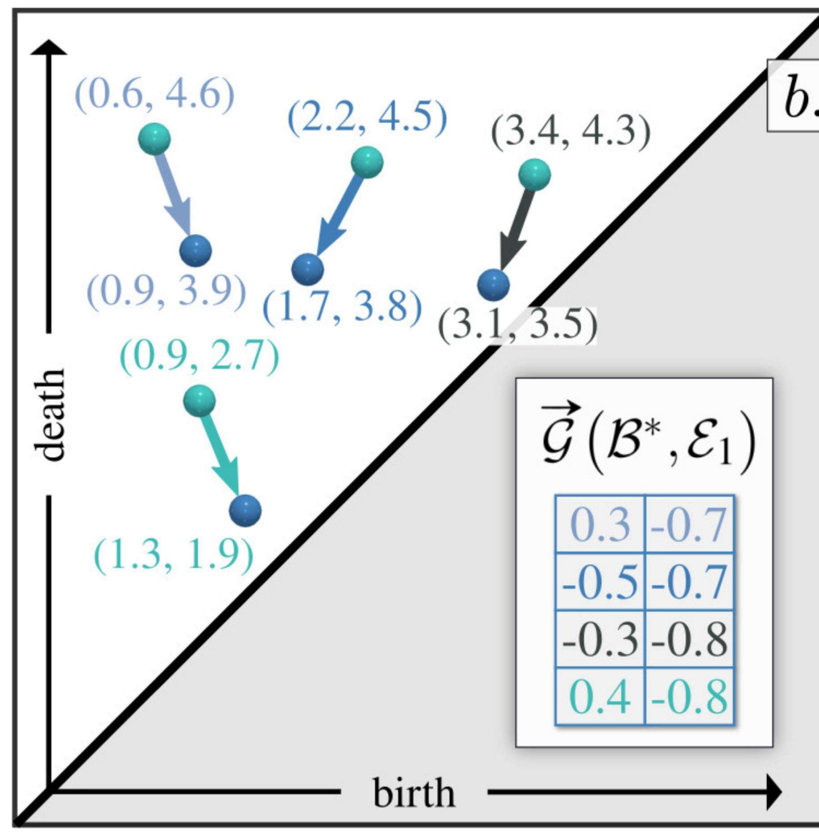
$$\vec{\mathcal{G}}\left(\mathcal{E},\mathcal{E}'\right) \in \mathbb{R}^{2\times|\mathcal{E}|}$$



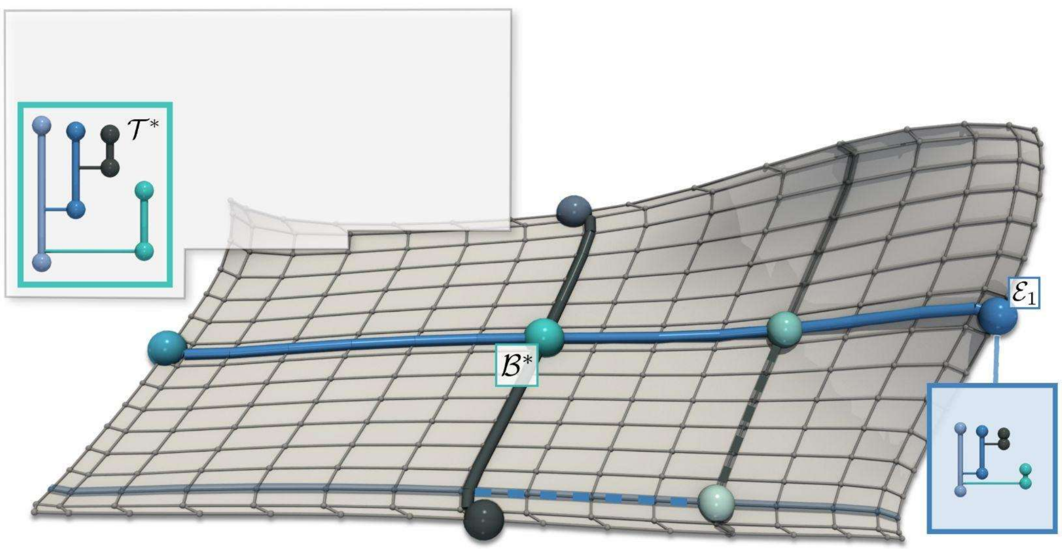
Orthogonal geodesics



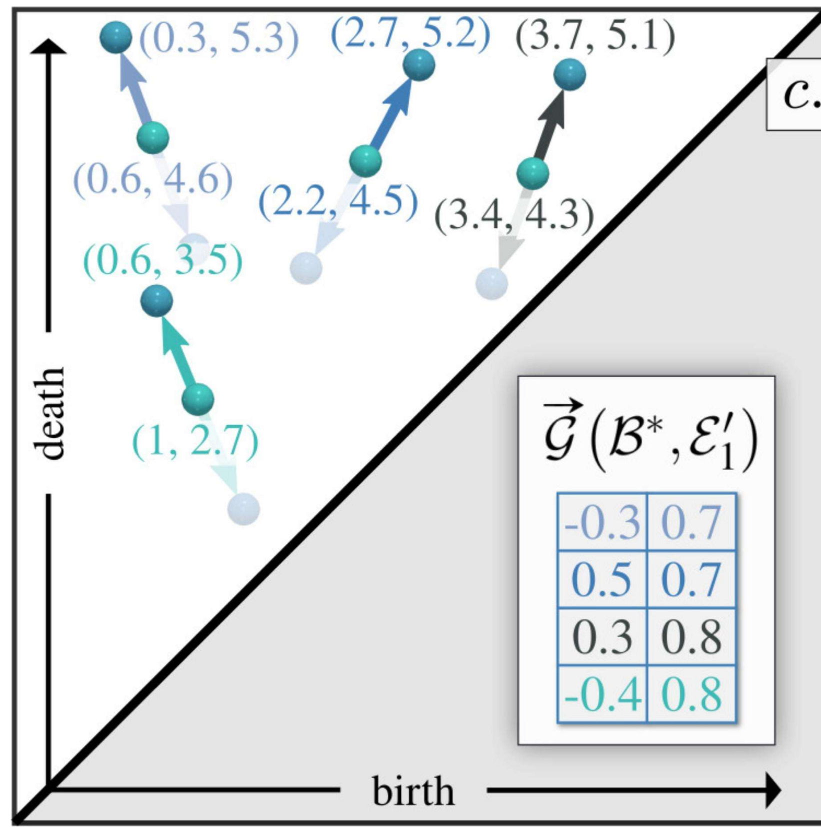
$$\vec{\mathcal{G}}\left(\mathcal{E},\mathcal{E}'\right)\cdot\vec{\mathcal{G}}\left(\mathcal{E},\mathcal{E}''\right)=0$$



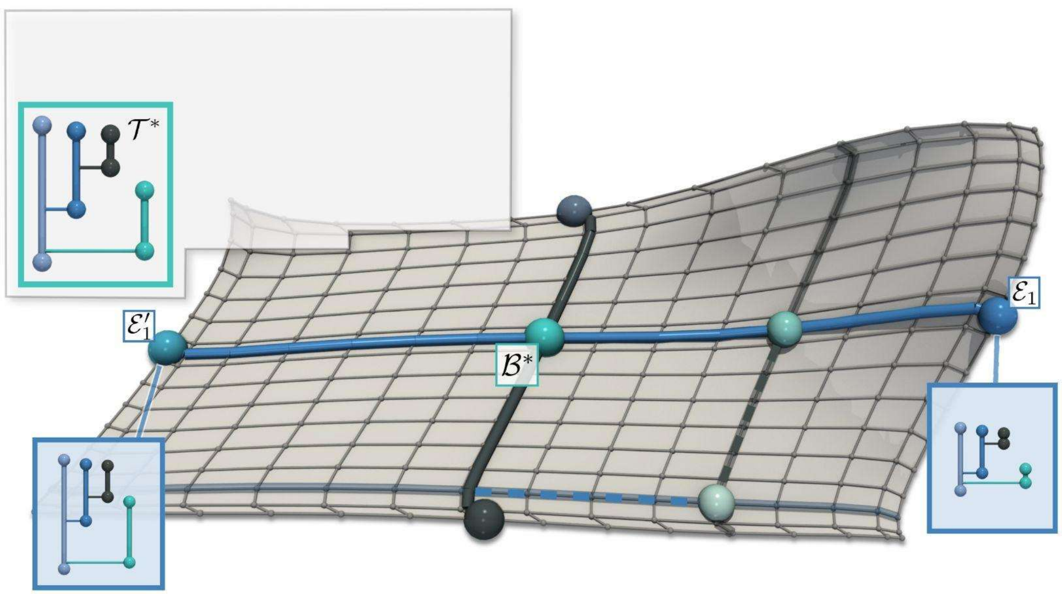
Collinear geodesics



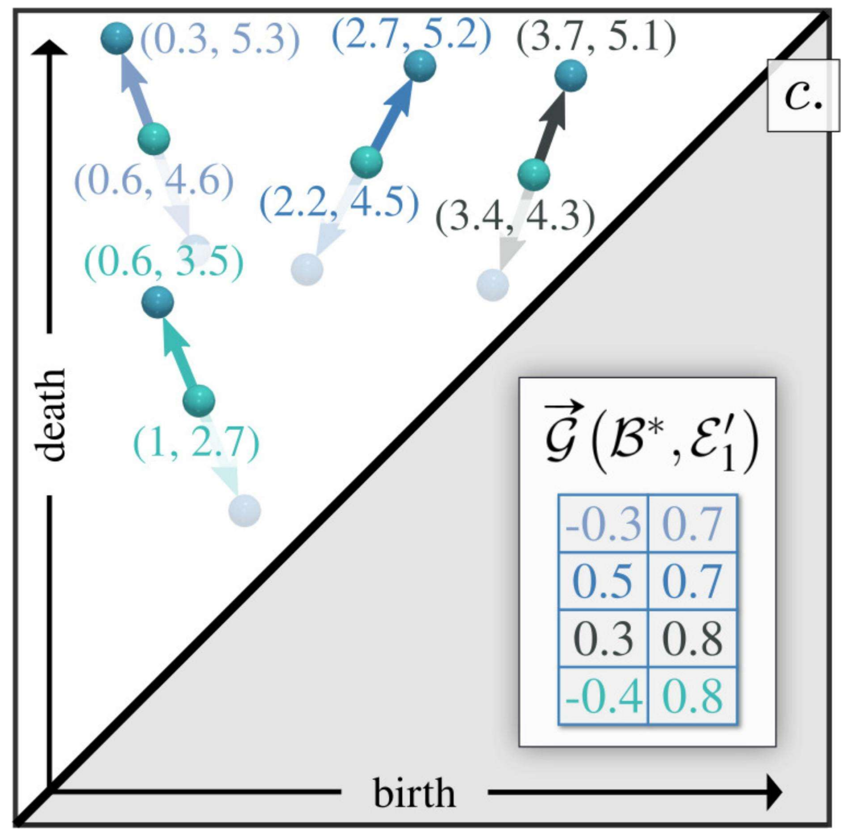
$$\vec{\mathcal{G}}(\mathcal{E},\mathcal{E}') = \lambda \vec{\mathcal{G}}(\mathcal{E},\mathcal{E}'')$$



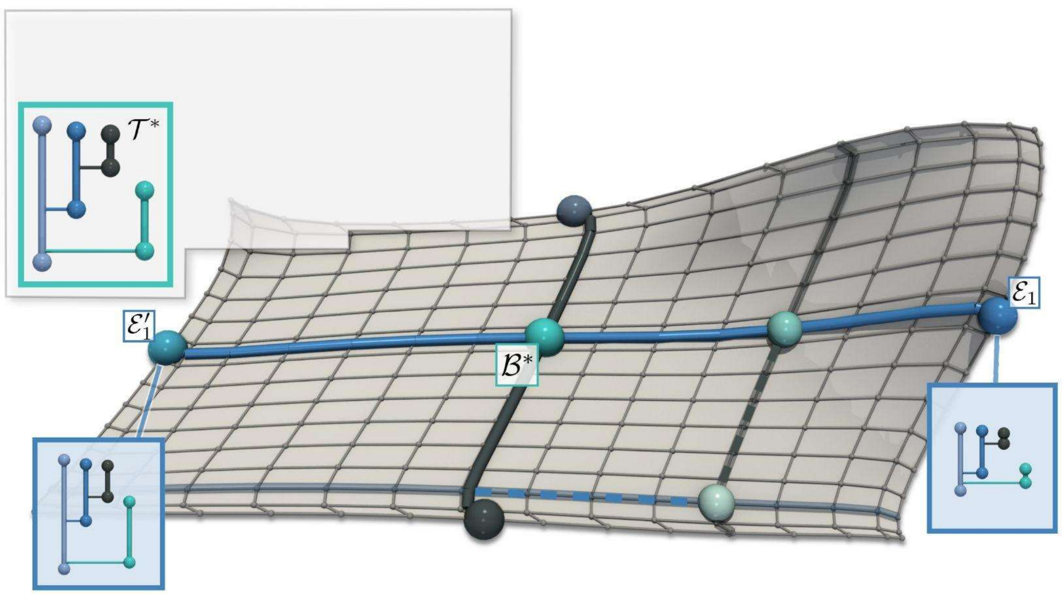
Collinear geodesics

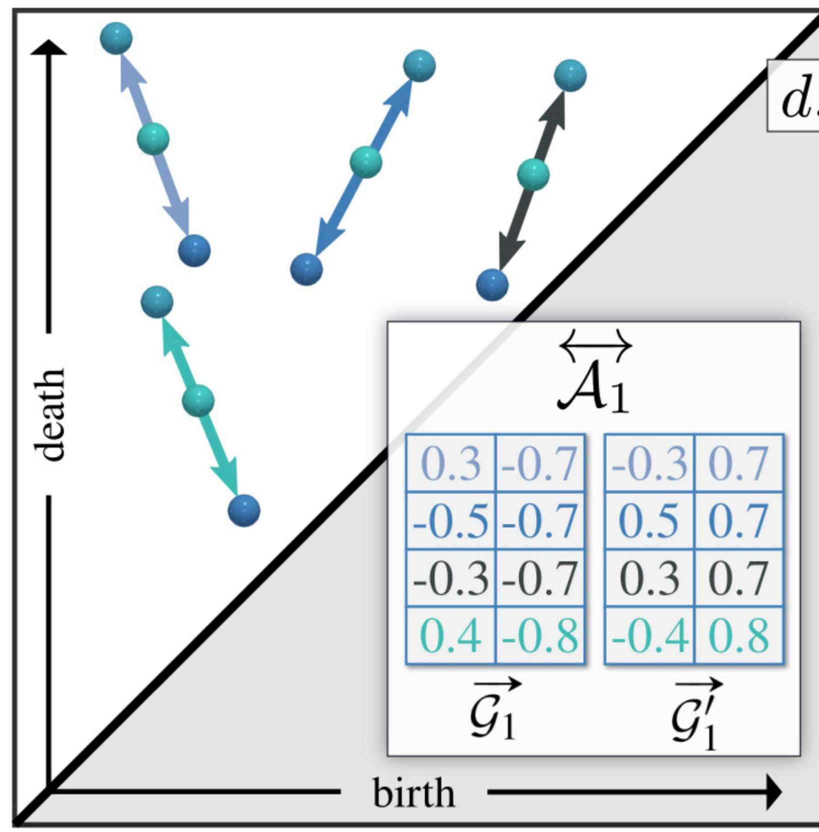


$$\vec{\mathcal{G}}(\mathcal{E},\mathcal{E}') = \lambda\vec{\mathcal{G}}(\mathcal{E},\mathcal{E}'')$$

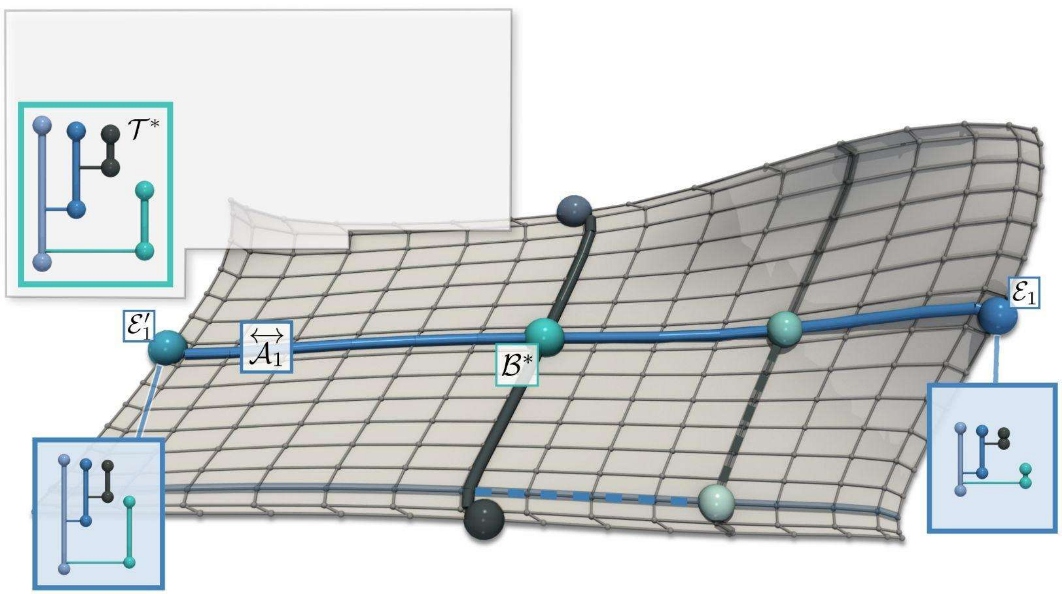


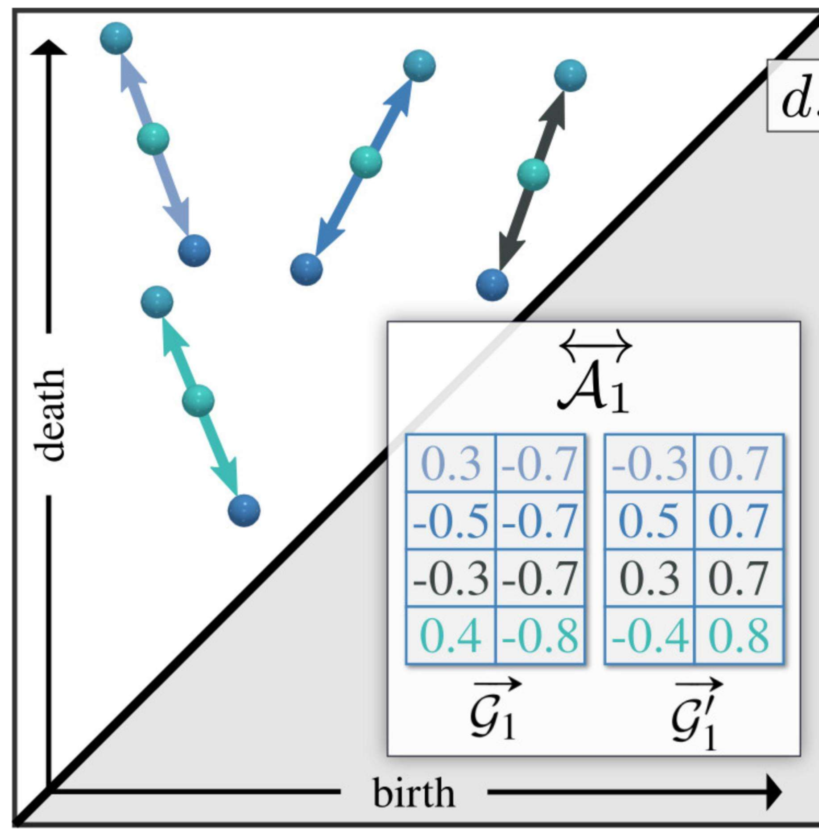
Geodesic axis



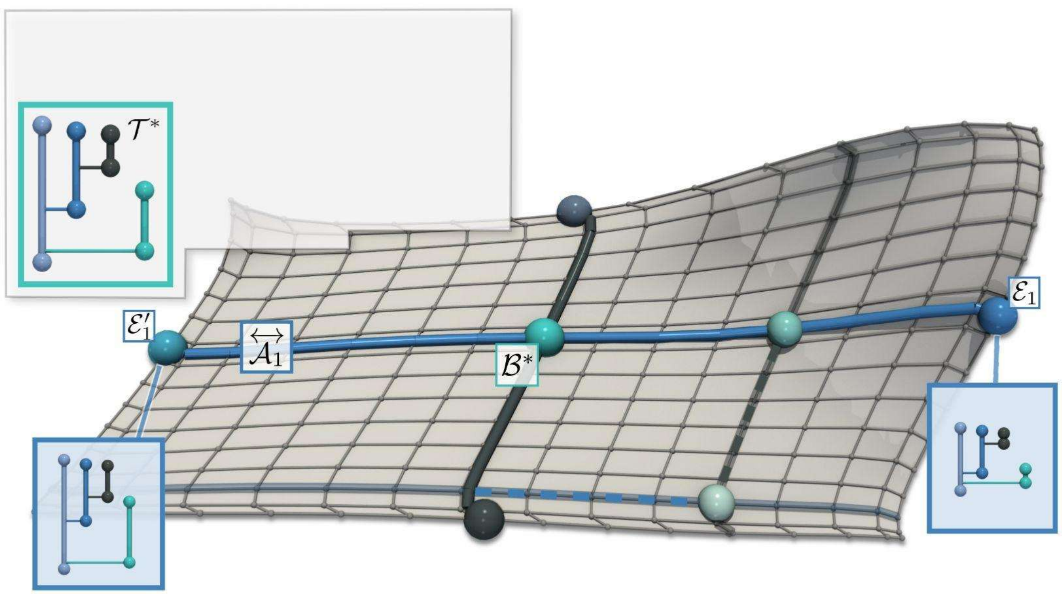


Geodesic axis

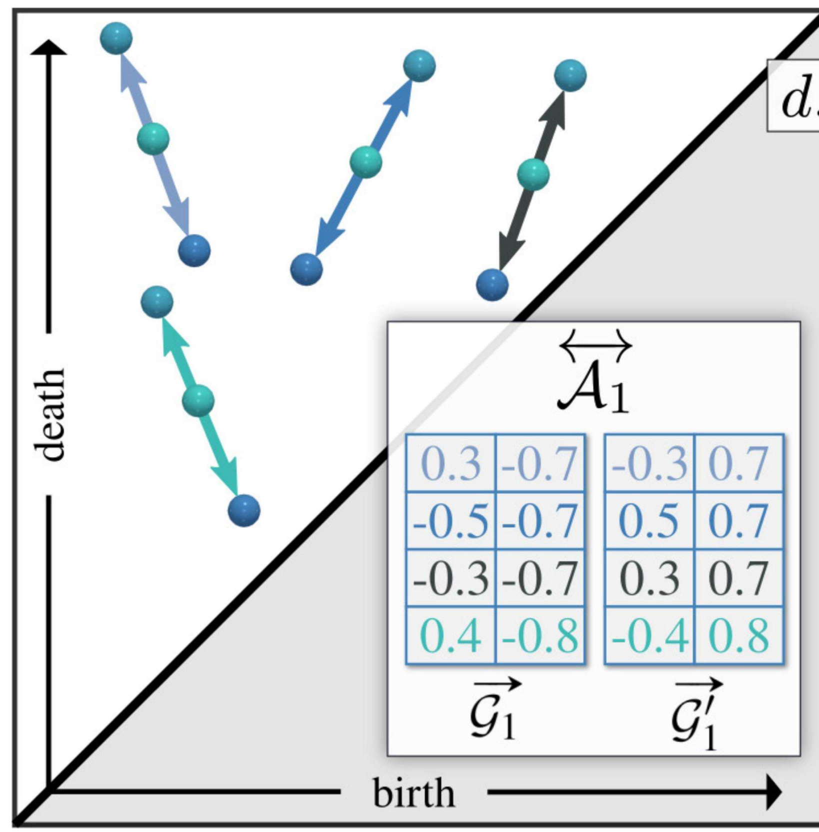




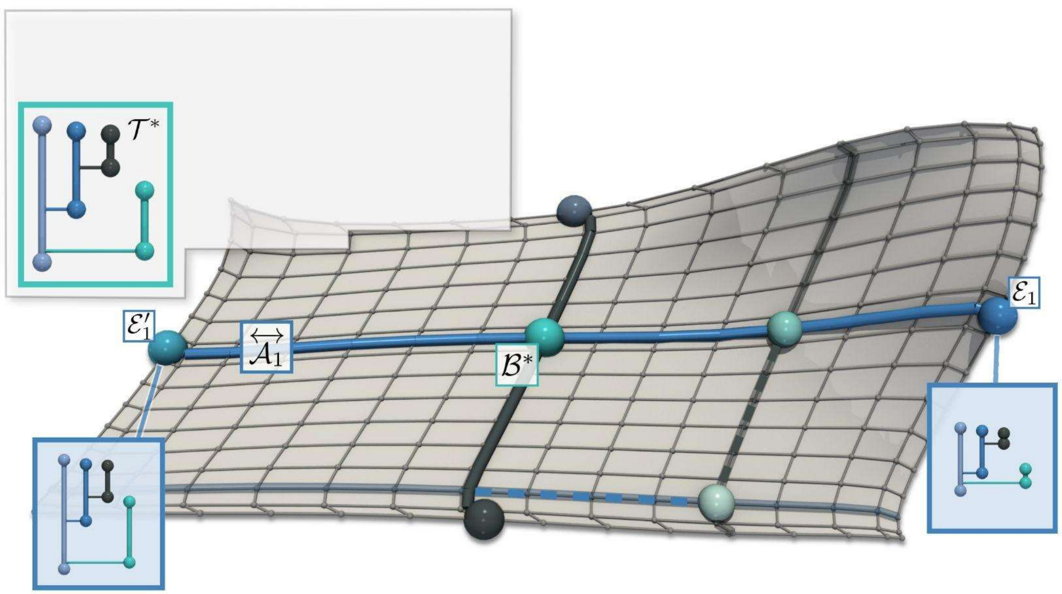
Geodesic axis



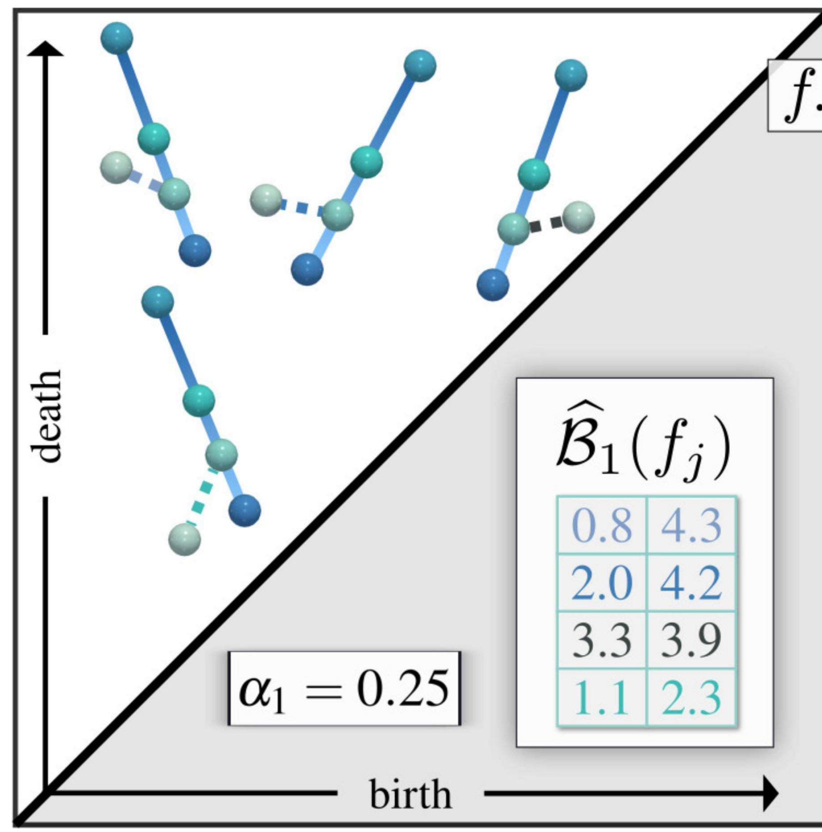
$$\overrightarrow{\mathcal{V}_i} = \overrightarrow{\mathcal{G}_i} - \overrightarrow{\mathcal{G}_i'} = \overrightarrow{\mathcal{G}}\left(\mathcal{E}_i', \mathcal{E}_i\right)$$



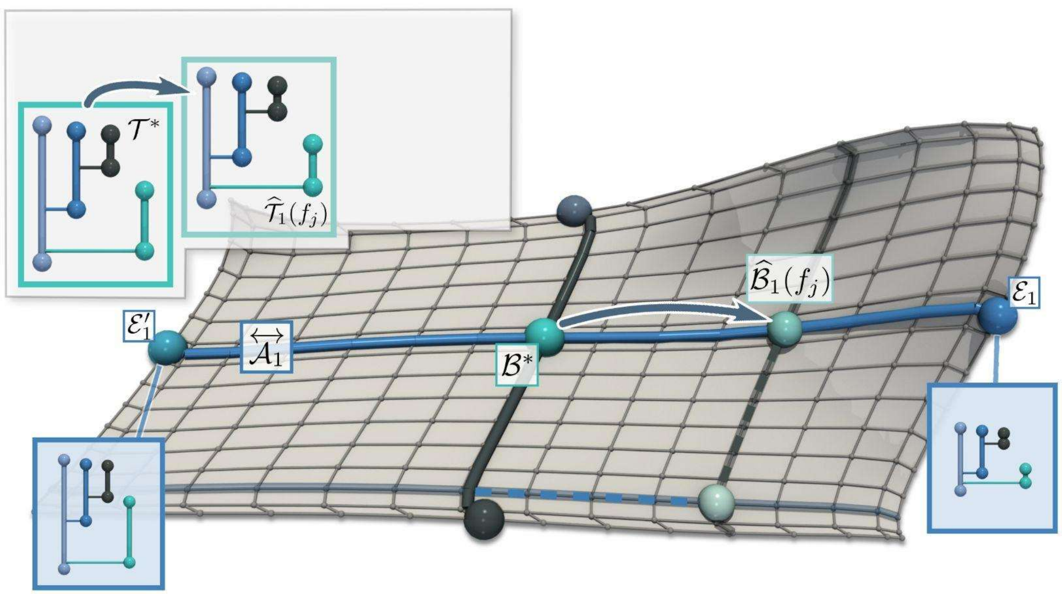
Axis projection



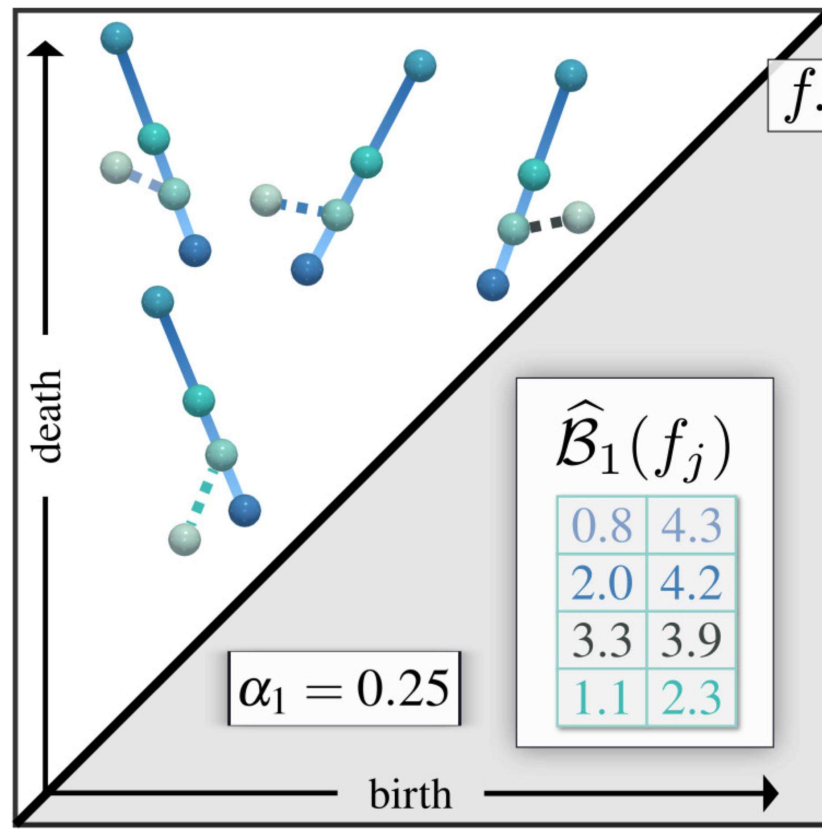
$$\mathcal{B}_{\overleftrightarrow{\mathcal{A}_i}} = \underset{\mathcal{B}' \in \overleftrightarrow{\mathcal{A}_i}}{\arg\min} \left(W_2^{\mathcal{T}}\left(\mathcal{B}, \mathcal{B}'\right) \right)$$



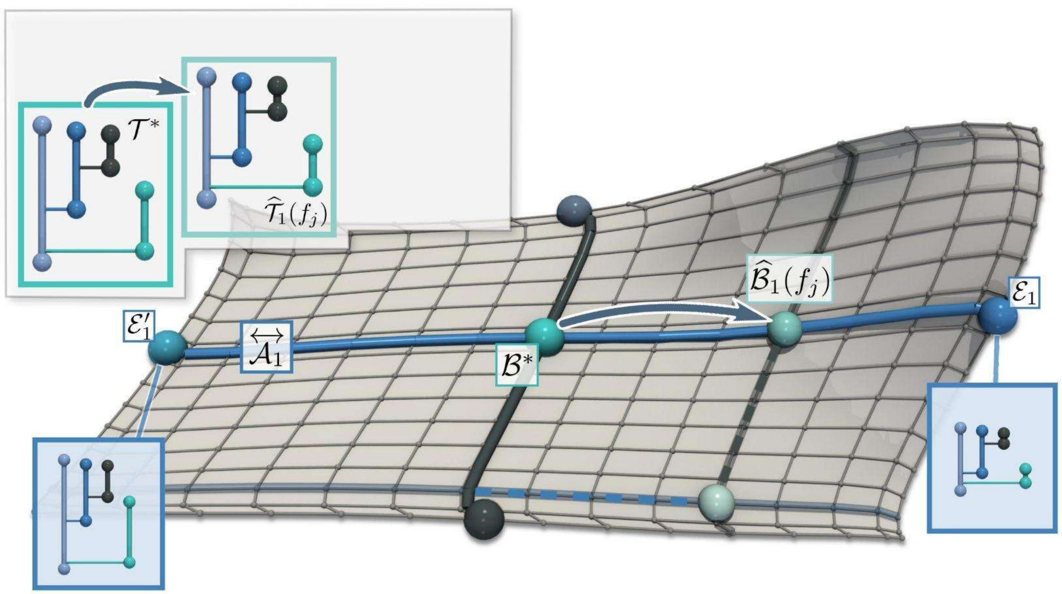
Axis projection



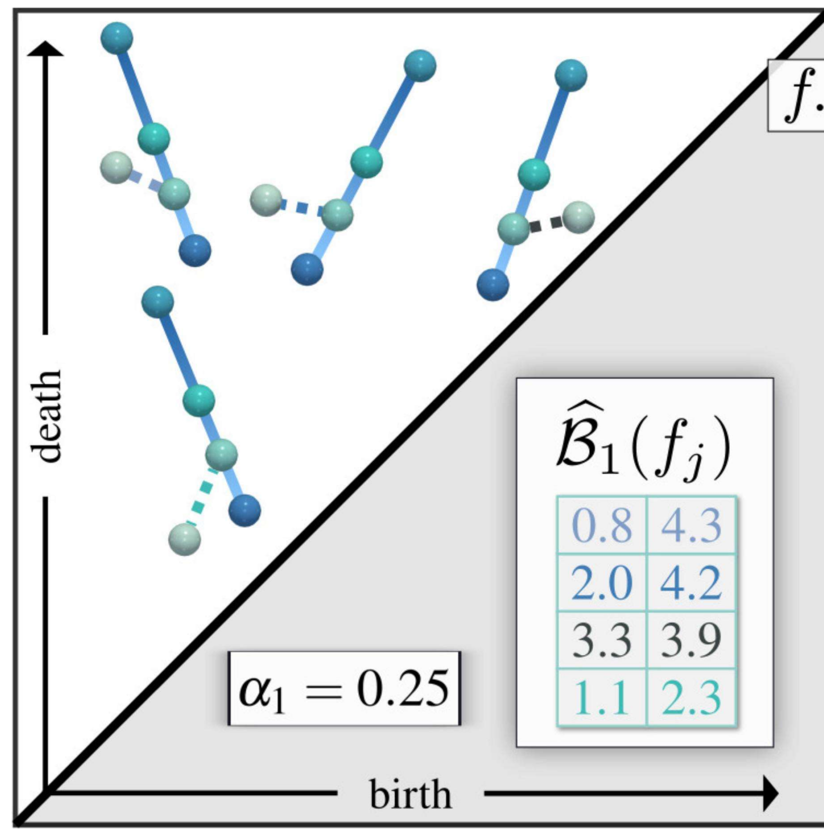
$$\mathcal{B}_{\overleftrightarrow{\mathcal{A}_i}} = \underset{\mathcal{B}' \in \overleftrightarrow{\mathcal{A}_i}}{\arg\min} \Big(W_2^{\mathcal{T}} \left(\mathcal{B}, \mathcal{B}' \right) \Big)$$



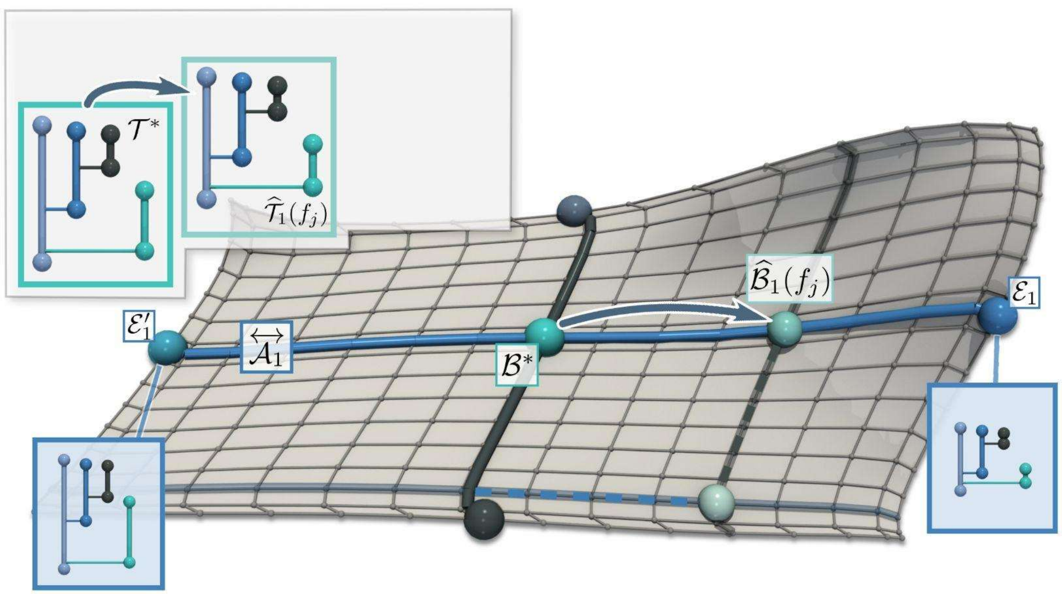
Axis projection

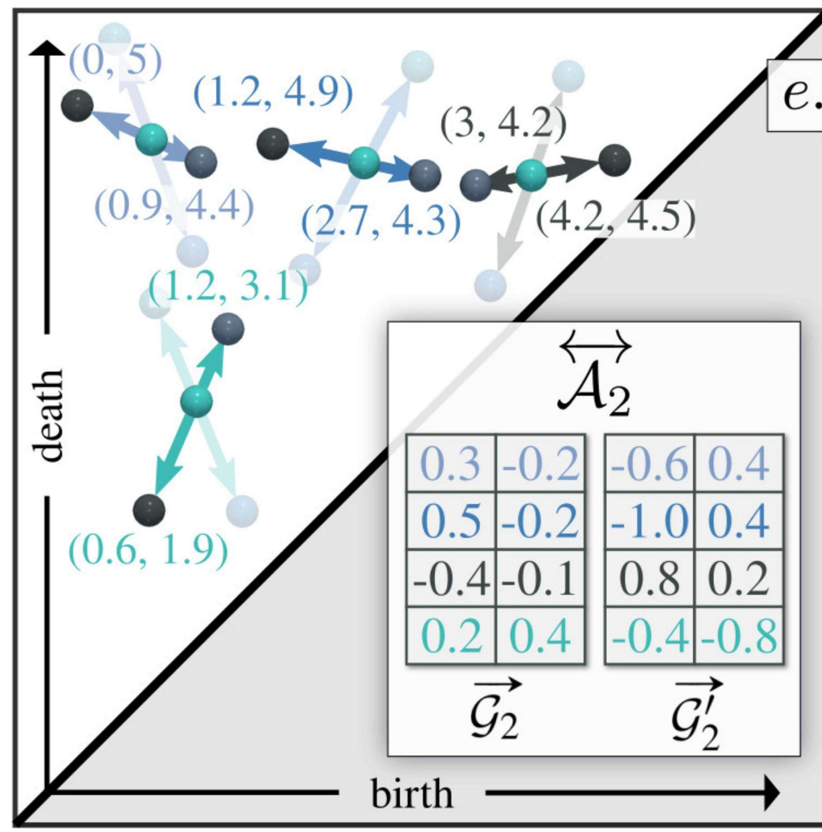


$$\mathcal{B} = \mathcal{E}_O + \overrightarrow{\mathcal{A}_i}(\mathcal{B}) = \mathcal{E}_O + \alpha_i \times \overrightarrow{\mathcal{G}_i} + (1 - \alpha_i) \times \overrightarrow{\mathcal{G}'_i}$$

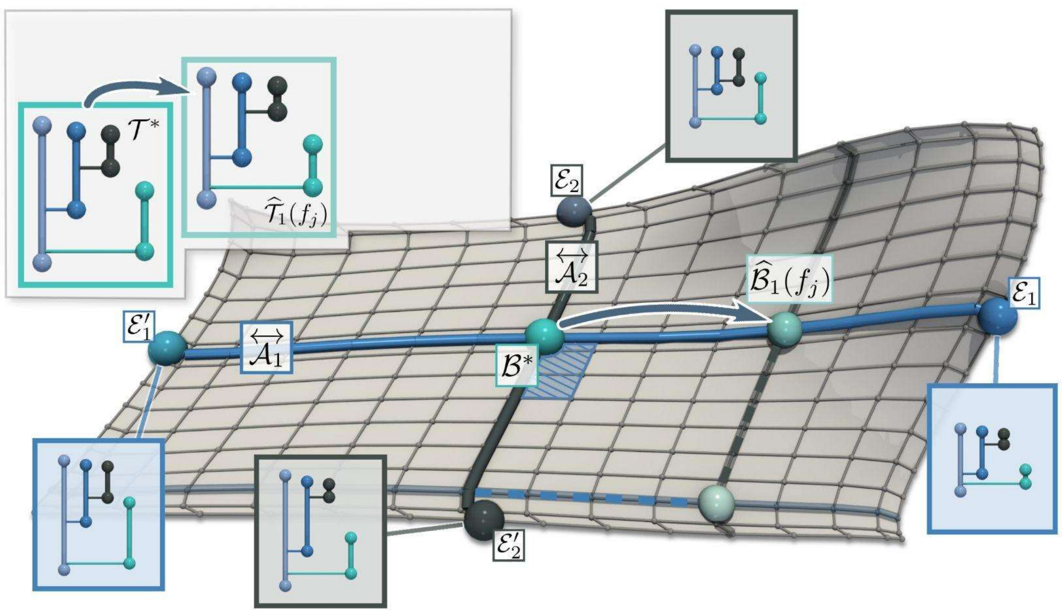


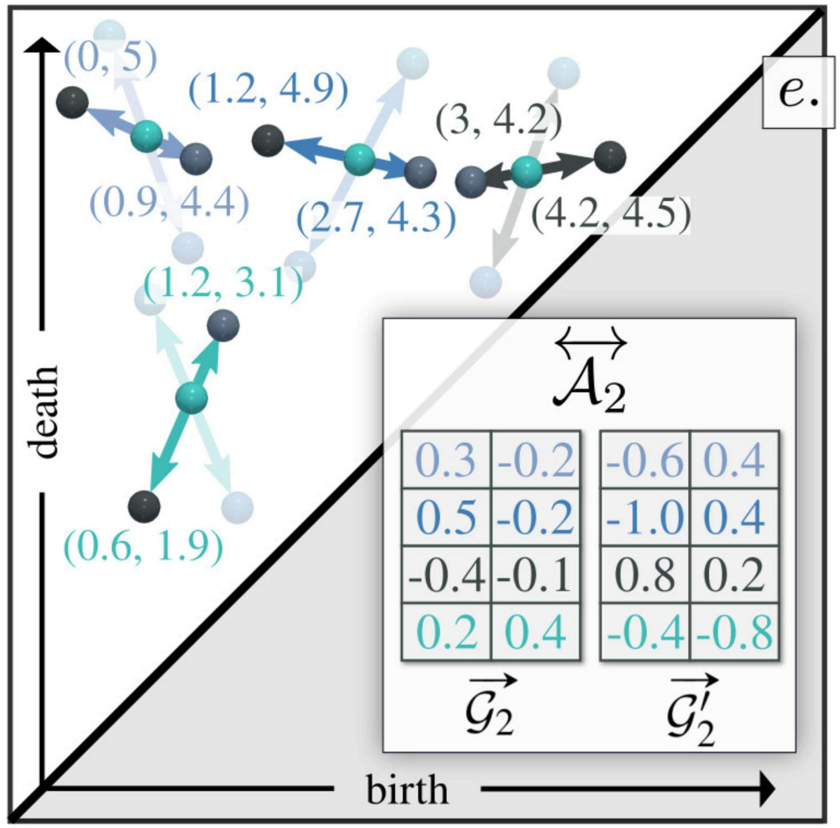
Orthogonal axis



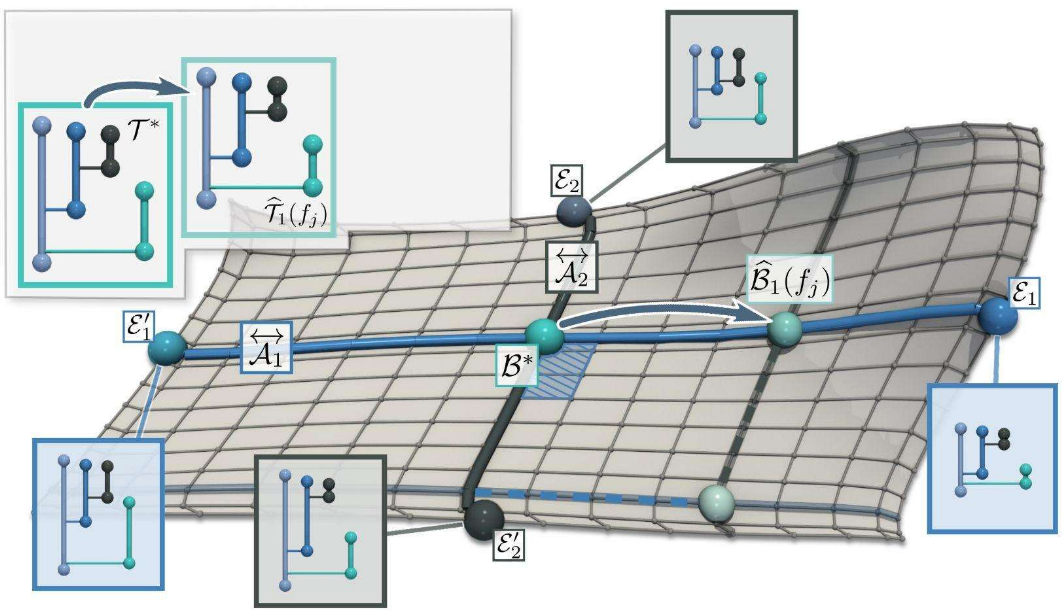


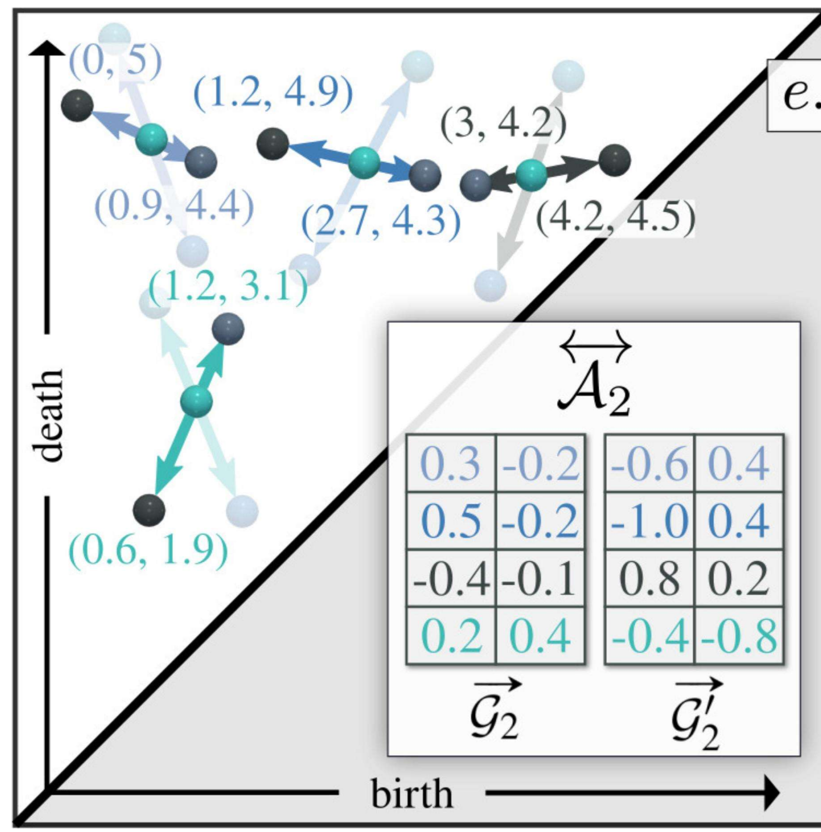
Orthogonal axis



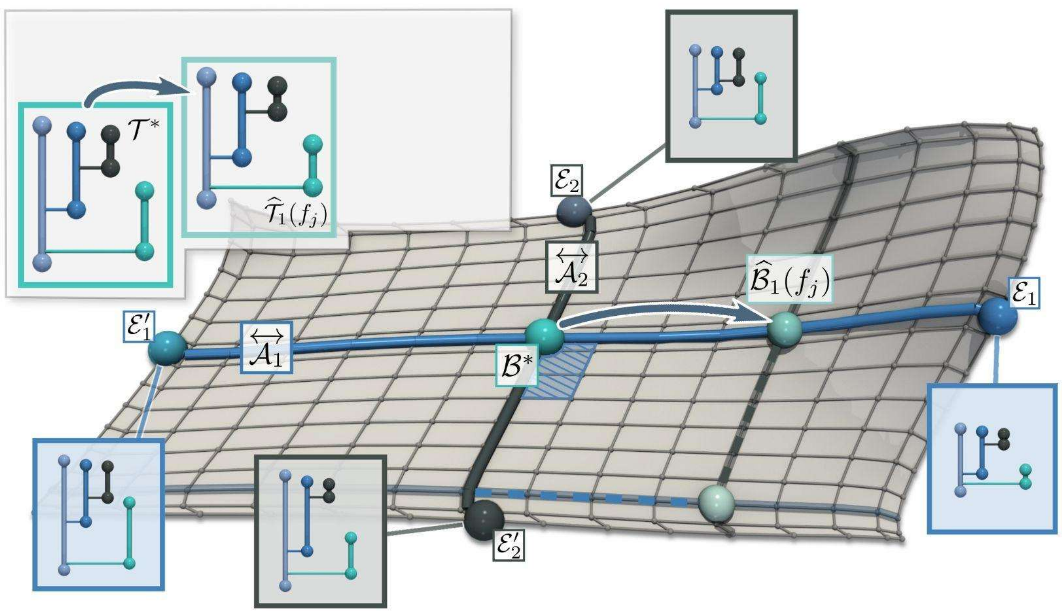


Orthogonal basis

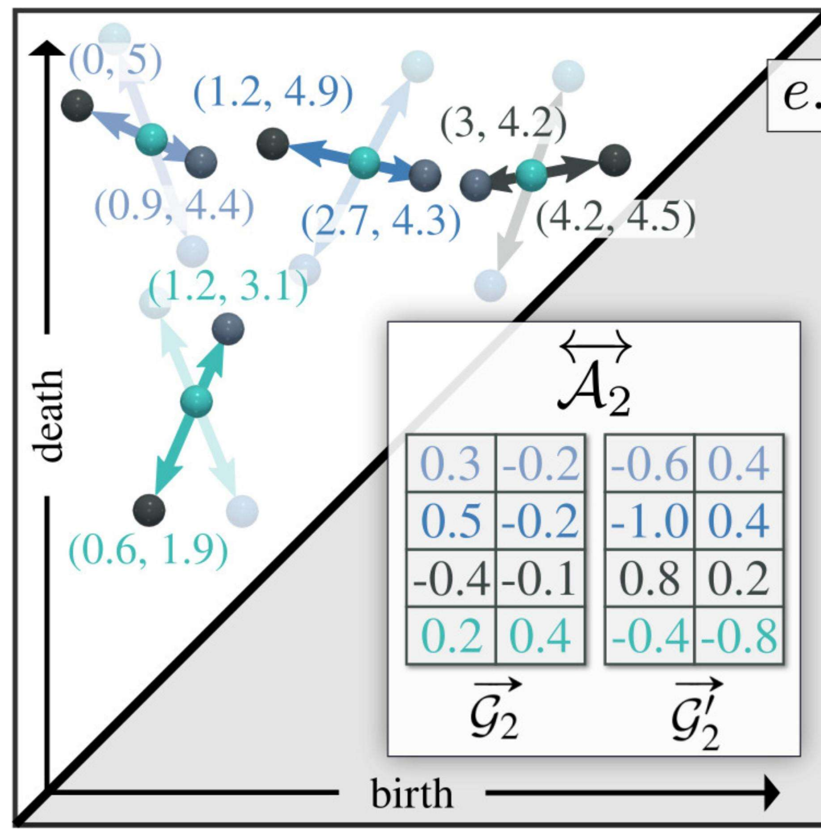




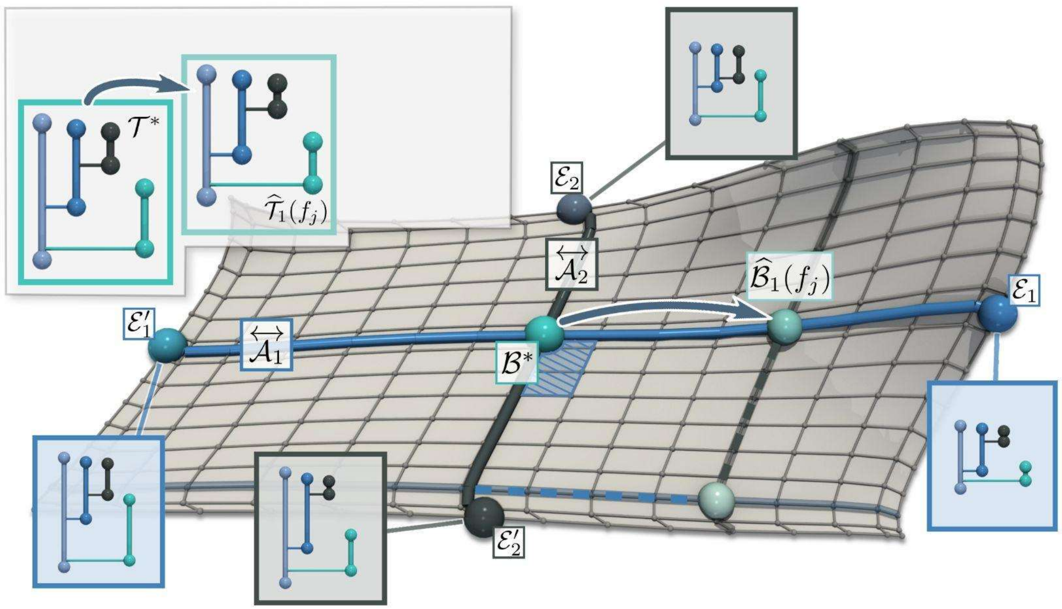
Orthogonal basis



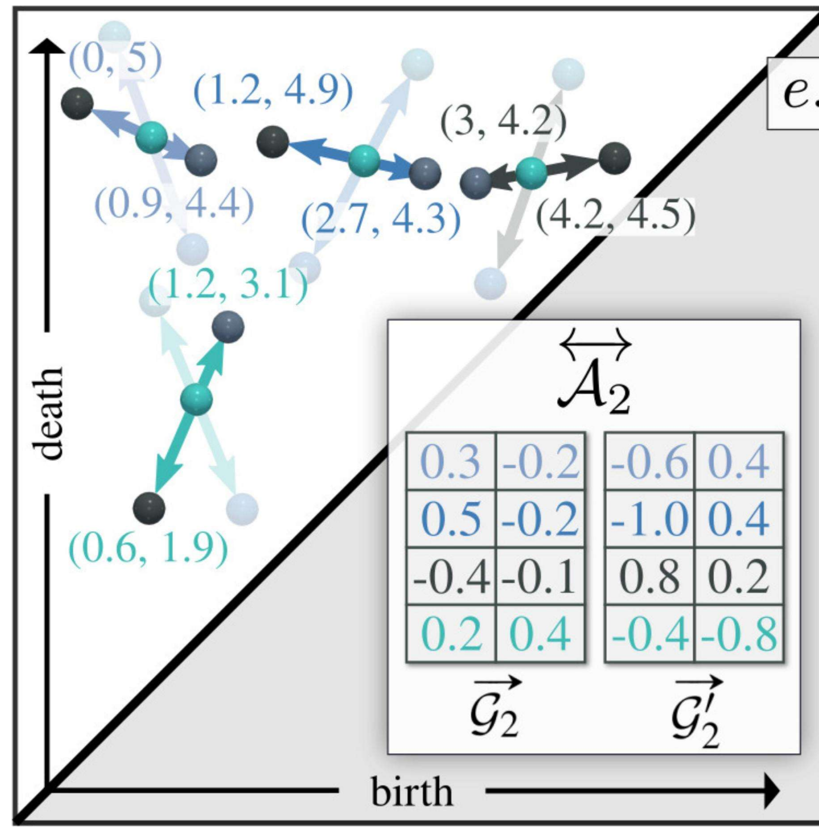
$$B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \ldots, \overleftrightarrow{\mathcal{A}_{d'}}\}$$



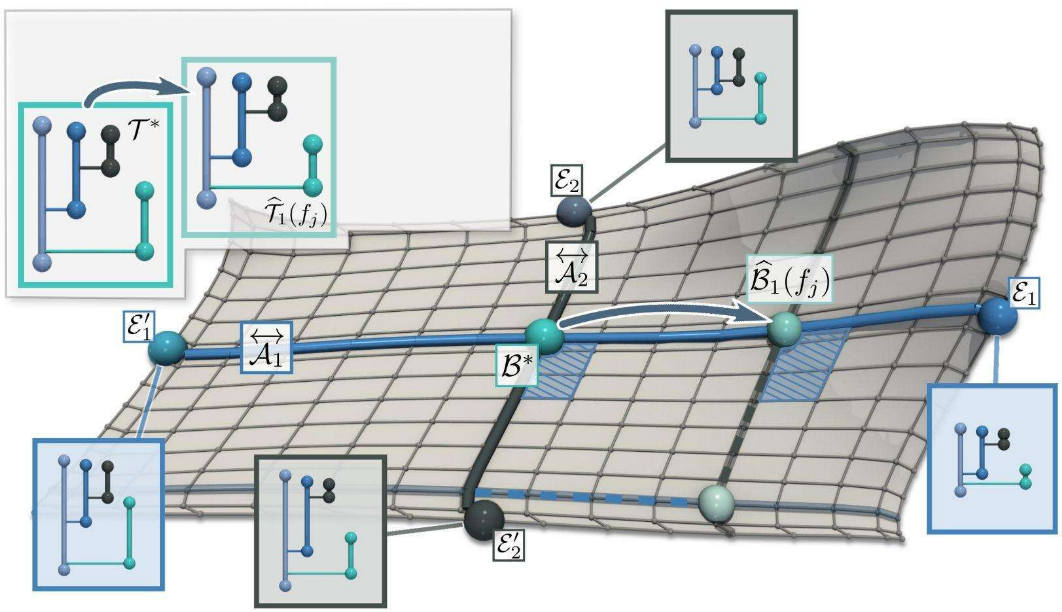
Orthogonal basis



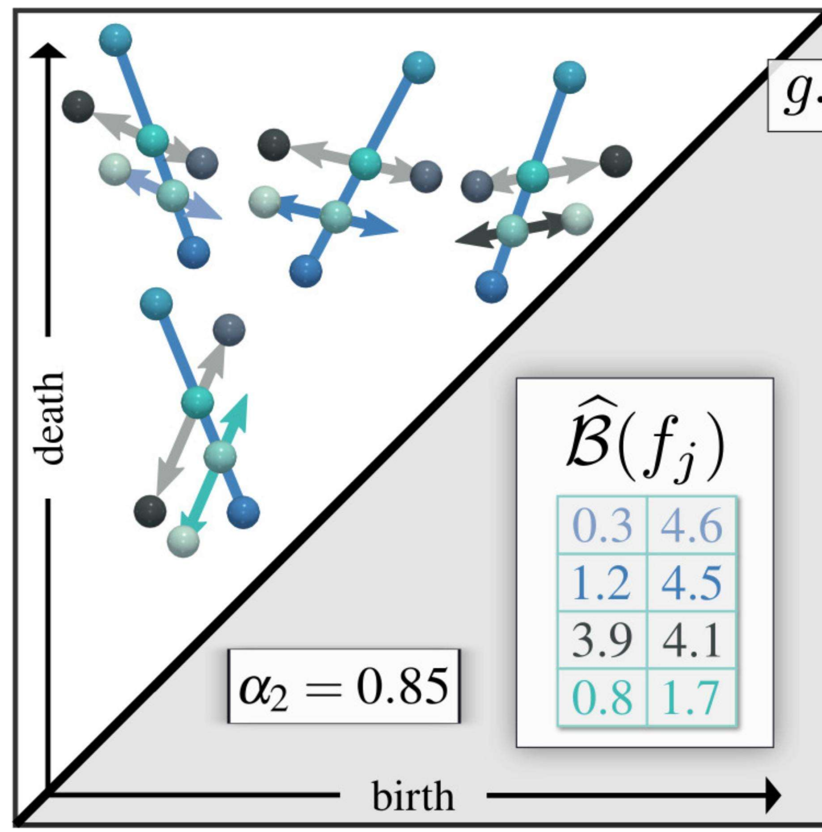
$$B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \ldots, \overleftrightarrow{\mathcal{A}_{d'}}\} \qquad \{\overrightarrow{\mathcal{V}_1}, \overrightarrow{\mathcal{V}_2}, \ldots, \overrightarrow{\mathcal{V}_{d'}}\}$$



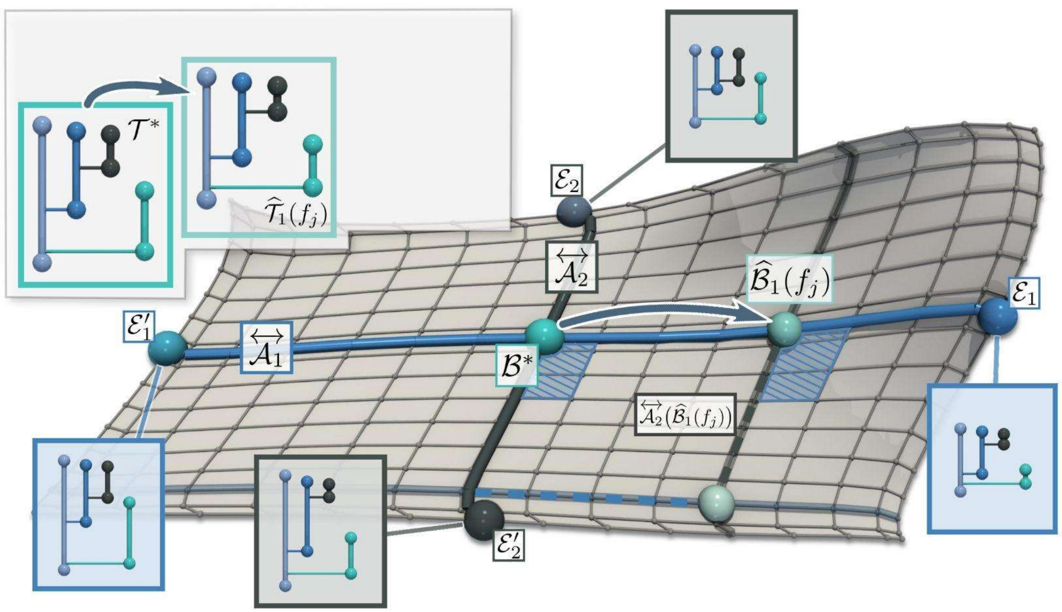
Axis translation



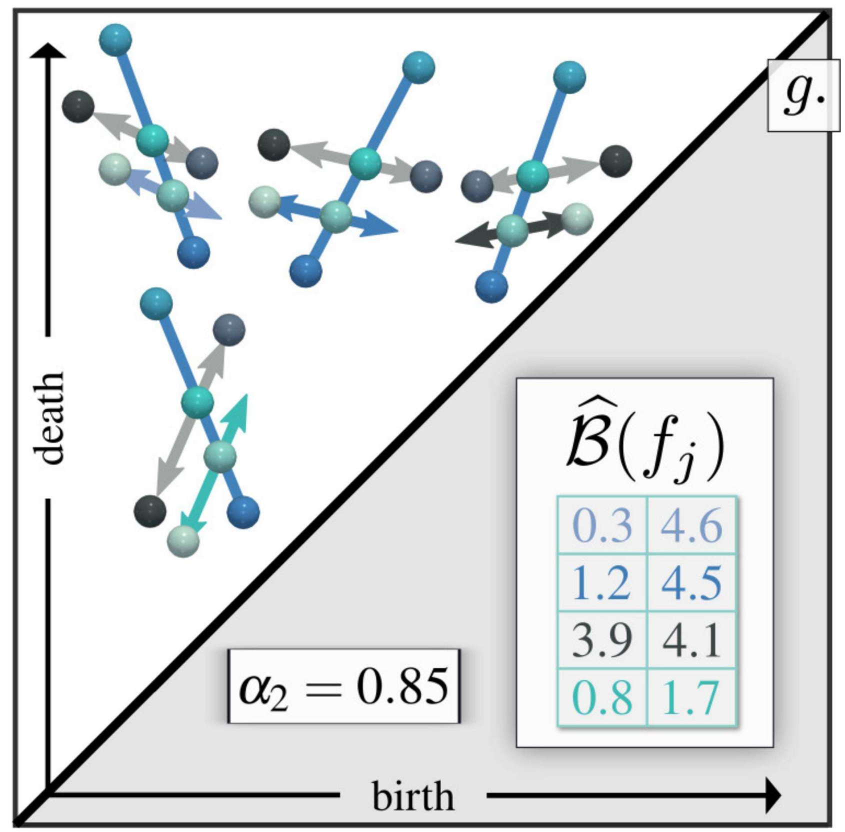
$$\overleftrightarrow{\mathcal{A}_j}(\mathcal{B}) = \left((\mathcal{B}, \mathcal{B} + \overrightarrow{\mathcal{G}_j}), (\mathcal{B}, \mathcal{B} + \overrightarrow{\mathcal{G}'_j})\right)$$



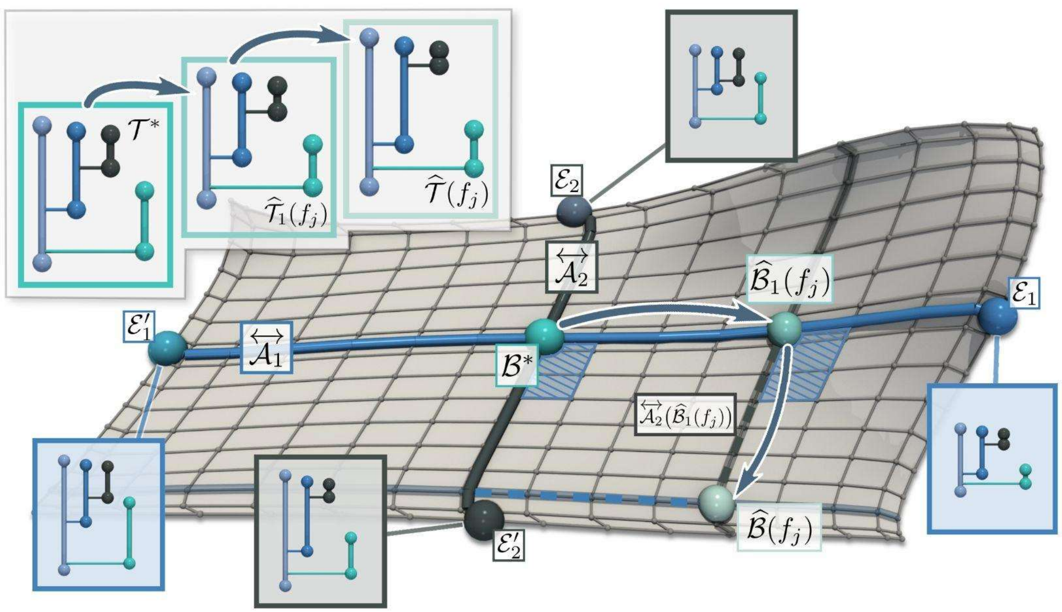
Axis translation

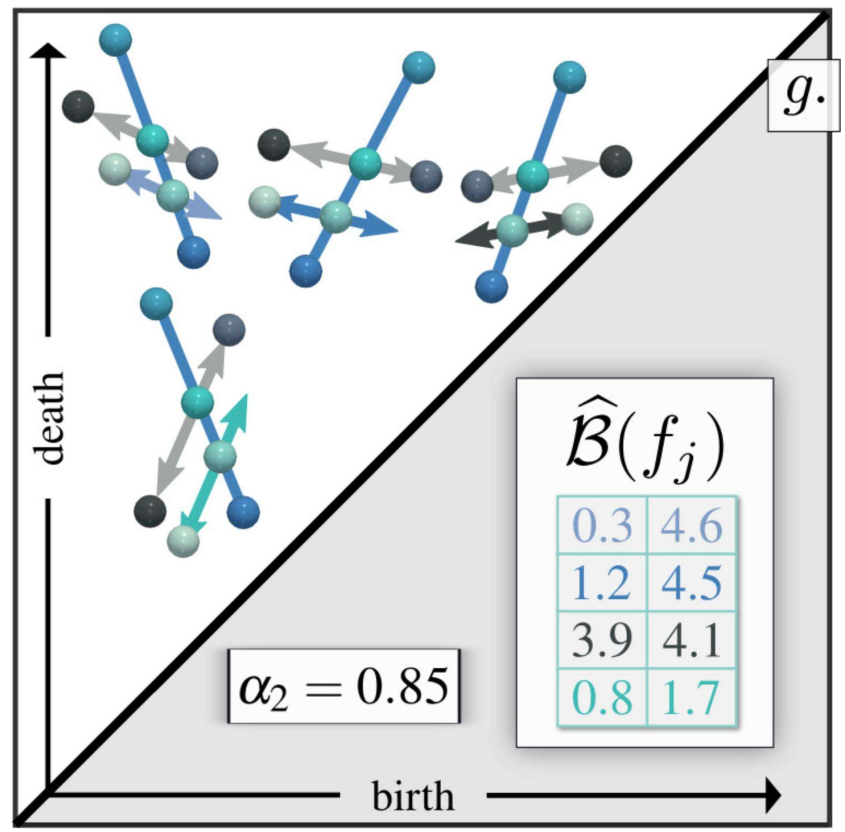


$$\overleftrightarrow{\mathcal{A}}_j(\mathcal{B}) = \left((\mathcal{B}, \mathcal{B} + \overrightarrow{\mathcal{G}_j}), (\mathcal{B}, \mathcal{B} + \overrightarrow{\mathcal{G}'_j})\right)$$



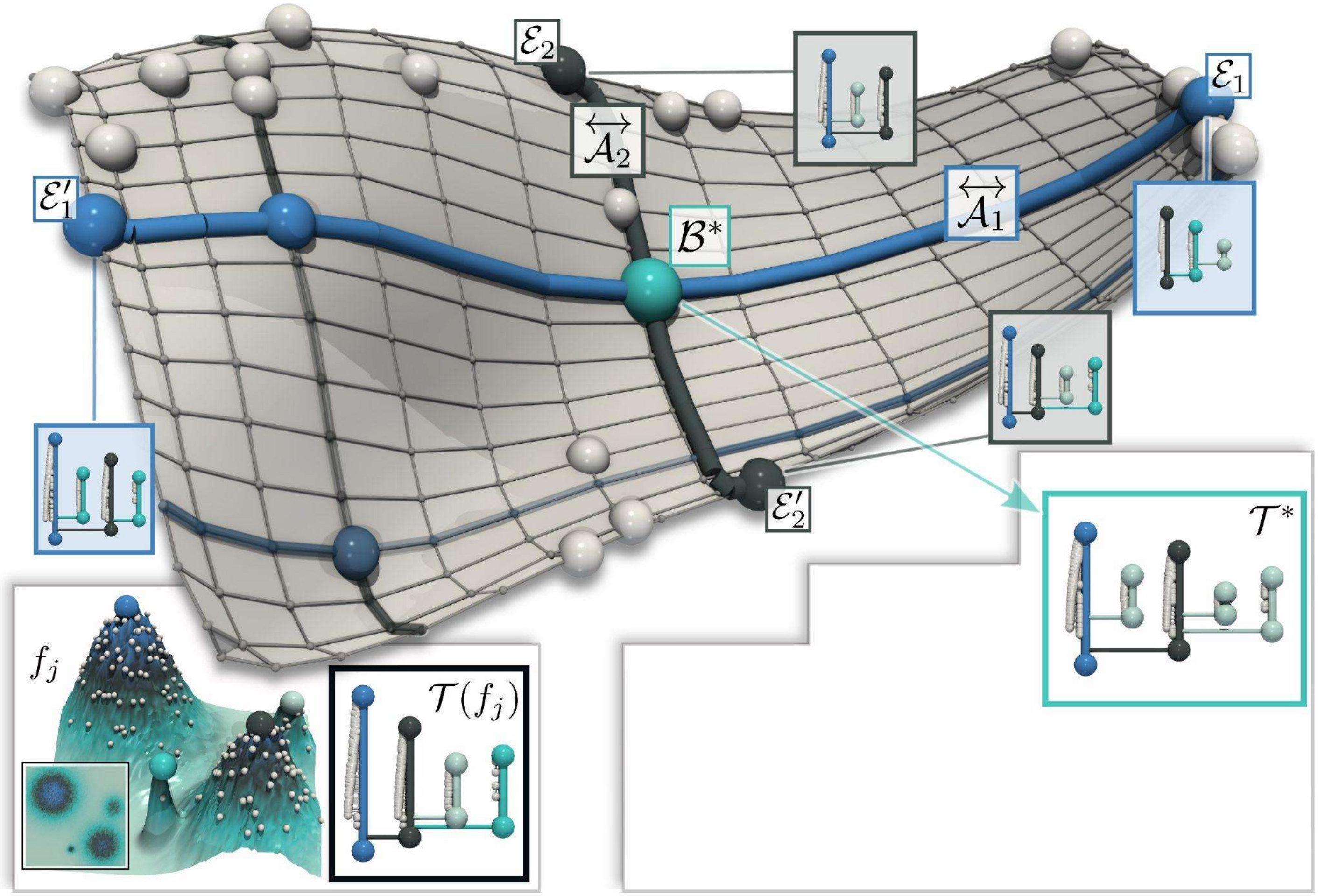
Axis translated projection



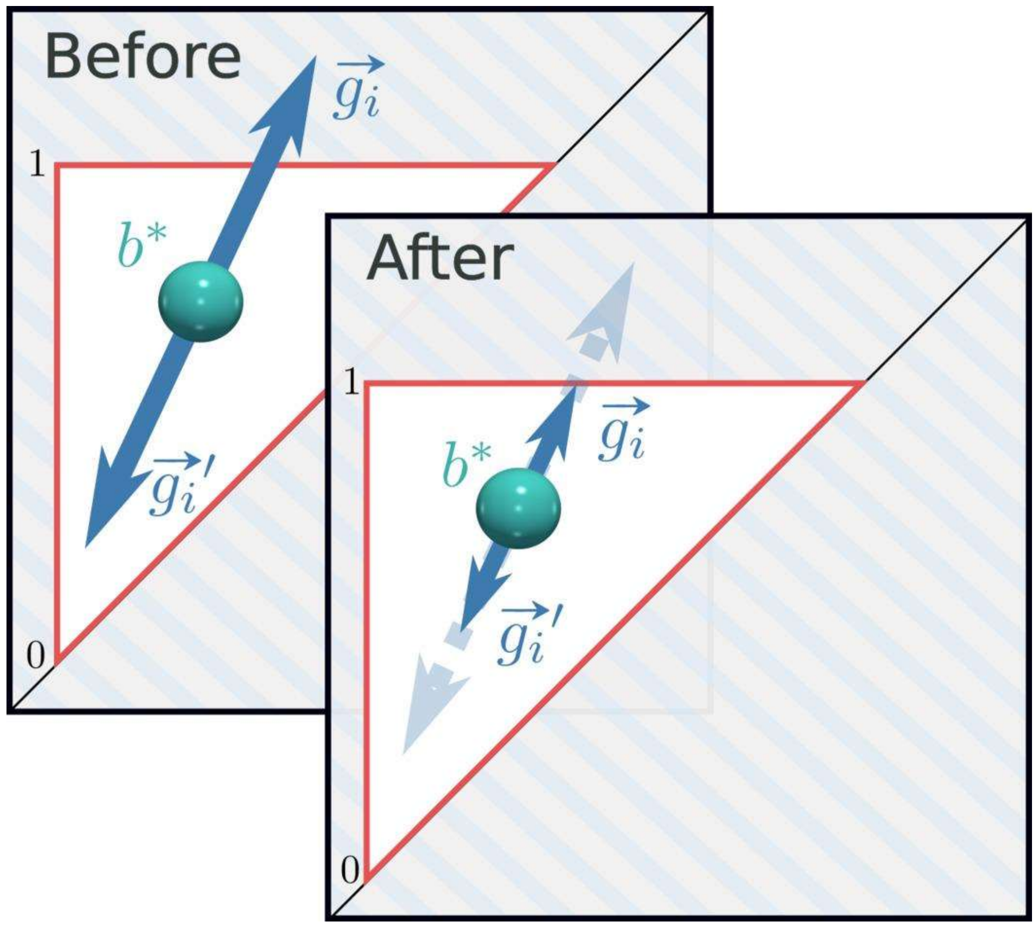


Overview

- **BDT Basis**
  - Barycenter
  - Geodesic axis
  - Orthogonal axes



From BDTs to MTs



Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_\mathcal{B} = \{\mathcal{B}(f_1), \ldots, \mathcal{B}(f_N)\}$.

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1, 2, \dots, N\}$).

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1, 2, \dots, N\}$).

$\mathcal{B}^* \leftarrow$WassersteinBarycenter$(\mathcal{S}_{\mathcal{B}})$;

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1, 2, \dots, N\}$).

$\mathcal{B}^* \leftarrow$WassersteinBarycenter($\mathcal{S}_{\mathcal{B}}$);

for $d' \in \{1, 2, \dots, d_{max}\}$ **do**

end for

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1,2,\dots,N\}$).

$\mathcal{B}^* \leftarrow$WassersteinBarycenter($\mathcal{S}_{\mathcal{B}}$);
for $d' \in \{1,2,\dots,d_{max}\}$ **do**
    $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$InitializeGeodesicAxis($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}$);
end for

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_\mathcal{B} = \{\mathcal{B}(f_1), \ldots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_\mathbb{B} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \ldots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_\mathbb{B}$ (with $j \in \{1, 2, \ldots, N\}$).

```
1:  B* ← WassersteinBarycenter(S_B);
2:  for d' ∈ {1, 2, ..., d_max} do
3:      A_d' ← InitializeGeodesicAxis(S_B, B*, B_𝔹);
4:      while E_{W_2^T}(B_𝔹) decreases do
5:          // Optimize the current geodesic axis A_d'
6:
7:
8:
9:
10:
11:
12:
13:
14:
15:     end while
16: end for
```

1: $\mathcal{B}^* \leftarrow$WassersteinBarycenter$(\mathcal{S}_\mathcal{B})$;
2: **for** $d' \in \{1, 2, \ldots, d_{max}\}$ **do**
3: $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$InitializeGeodesicAxis$(\mathcal{S}_\mathcal{B}, \mathcal{B}^*, B_\mathbb{B})$;
4: **while** $E_{W_2^\mathcal{T}}(B_\mathbb{B})$ decreases **do**
5: // Optimize the current geodesic axis $\overleftrightarrow{\mathcal{A}_{d'}}$
6:
7:
8:
9:
10:
11:
12:
13:
14:
15: **end while**
16: **end for**

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1,2,\dots,N\}$).

$\mathcal{B}^* \leftarrow$WassersteinBarycenter($\mathcal{S}_{\mathcal{B}}$);

for $d' \in \{1,2,\dots,d_{max}\}$ **do**

  $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$InitializeGeodesicAxis($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}$);

  while $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}})$ decreases **do**

    // Optimize the current geodesic axis $\overleftrightarrow{\mathcal{A}_{d'}}$

    $\alpha_{d'}^{j\in\{1,2,...N\}} \leftarrow$ProjectTrees($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j\in\{1,2,...N\}}$);

  end while

end for

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1, 2, \dots, N\}$).

$\mathcal{B}^* \leftarrow$WassersteinBarycenter($\mathcal{S}_{\mathcal{B}}$);
for $d' \in \{1, 2, \dots, d_{max}\}$ **do**
  $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$InitializeGeodesicAxis($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}$);
  while $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}})$ decreases **do**
    // Optimize the current geodesic axis $\overleftrightarrow{\mathcal{A}_{d'}}$
    $\alpha_{d'}^{j \in \{1,2,\dots N\}} \leftarrow$ProjectTrees($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}}$);
    $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}}) \leftarrow$EvaluateFittingEnergy($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}}$);
  end while
end for

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1, 2, \dots, N\}$).

$\mathcal{B}^* \leftarrow$WassersteinBarycenter$(\mathcal{S}_{\mathcal{B}})$;

for $d' \in \{1, 2, \dots, d_{max}\}$ **do**

  $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$InitializeGeodesicAxis$(\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}})$;

  while $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}})$ decreases **do**

    // Optimize the current geodesic axis $\overleftrightarrow{\mathcal{A}_{d'}}$

    $\alpha_{d'}^{j \in \{1,2,\dots N\}} \leftarrow$ProjectTrees$(\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}})$;

    $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}}) \leftarrow$EvaluateFittingEnergy$(\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}})$;

    $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$OptimizeFittingEnergy$(\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}})$;

  end while

end for

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1, 2, \dots, N\}$).

```
B* ←WassersteinBarycenter(S_B);
for d' ∈ {1,2,...,d_max} do
    A_d' ←InitializeGeodesicAxis(S_B, B*, B_𝔹);
    while E_{W2^T}(B_𝔹) decreases do
        // Optimize the current geodesic axis A_d'
        α_d'^{j∈{1,2,...N}} ←ProjectTrees(S_B, B*, B_𝔹, α^{j∈{1,2,...N}});
        E_{W2^T}(B_𝔹) ←EvaluateFittingEnergy(S_B, B*, B_𝔹, α^{j∈{1,2,...N}});
        A_d' ←OptimizeFittingEnergy(S_B, B*, B_𝔹, α^{j∈{1,2,...N}});
        // Enforce the constaints
        while A_d' evolves do
        end while
    end while
end for
```

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1,2,\dots,N\}$).

$\mathcal{B}^* \leftarrow$WassersteinBarycenter$(\mathcal{S}_{\mathcal{B}})$;
for $d' \in \{1,2,\dots,d_{max}\}$ **do**
    $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$InitializeGeodesicAxis$(\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}})$;
    while $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}})$ decreases **do**
        // Optimize the current geodesic axis $\overleftrightarrow{\mathcal{A}_{d'}}$
        $\alpha_{d'}^{j\in\{1,2,\dots N\}} \leftarrow$ProjectTrees$(\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j\in\{1,2,\dots N\}})$;
        $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}}) \leftarrow$EvaluateFittingEnergy$(\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j\in\{1,2,\dots N\}})$;
        $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$OptimizeFittingEnergy$(\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j\in\{1,2,\dots N\}})$;
        // Enforce the constaints
        while $\overleftrightarrow{\mathcal{A}_{d'}}$ evolves **do**
            $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$EnforceGeodesics$(\mathcal{B}^*, \overleftrightarrow{\mathcal{A}_{d'}})$;
        end while
    end while
end for

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1, 2, \dots, N\}$).

$\mathcal{B}^* \leftarrow$ WassersteinBarycenter($\mathcal{S}_{\mathcal{B}}$);
for $d' \in \{1, 2, \dots, d_{max}\}$ **do**
  $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$ InitializeGeodesicAxis($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}$);
  while $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}})$ decreases **do**
    // Optimize the current geodesic axis $\overleftrightarrow{\mathcal{A}_{d'}}$
    $\alpha_{d'}^{j \in \{1,2,\dots N\}} \leftarrow$ ProjectTrees($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}}$);
    $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}}) \leftarrow$ EvaluateFittingEnergy($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}}$);
    $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$ OptimizeFittingEnergy($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}}$);
    // Enforce the constaints
    while $\overleftrightarrow{\mathcal{A}_{d'}}$ evolves **do**
      $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$ EnforceGeodesics($\mathcal{B}^*, \overleftrightarrow{\mathcal{A}_{d'}}$);
      $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$ EnforceNegativeCollinearity($\overleftrightarrow{\mathcal{A}_{d'}}$);

    end while
  end while
end for

Overview

Algorithm 1 Merge Tree Principal Geodesic Analysis (MT-PGA) Algorithm

Input: Set of BDTs $\mathcal{S}_{\mathcal{B}} = \{\mathcal{B}(f_1), \dots, \mathcal{B}(f_N)\}$.

Output1: Basis origin $\mathcal{B}^*$;

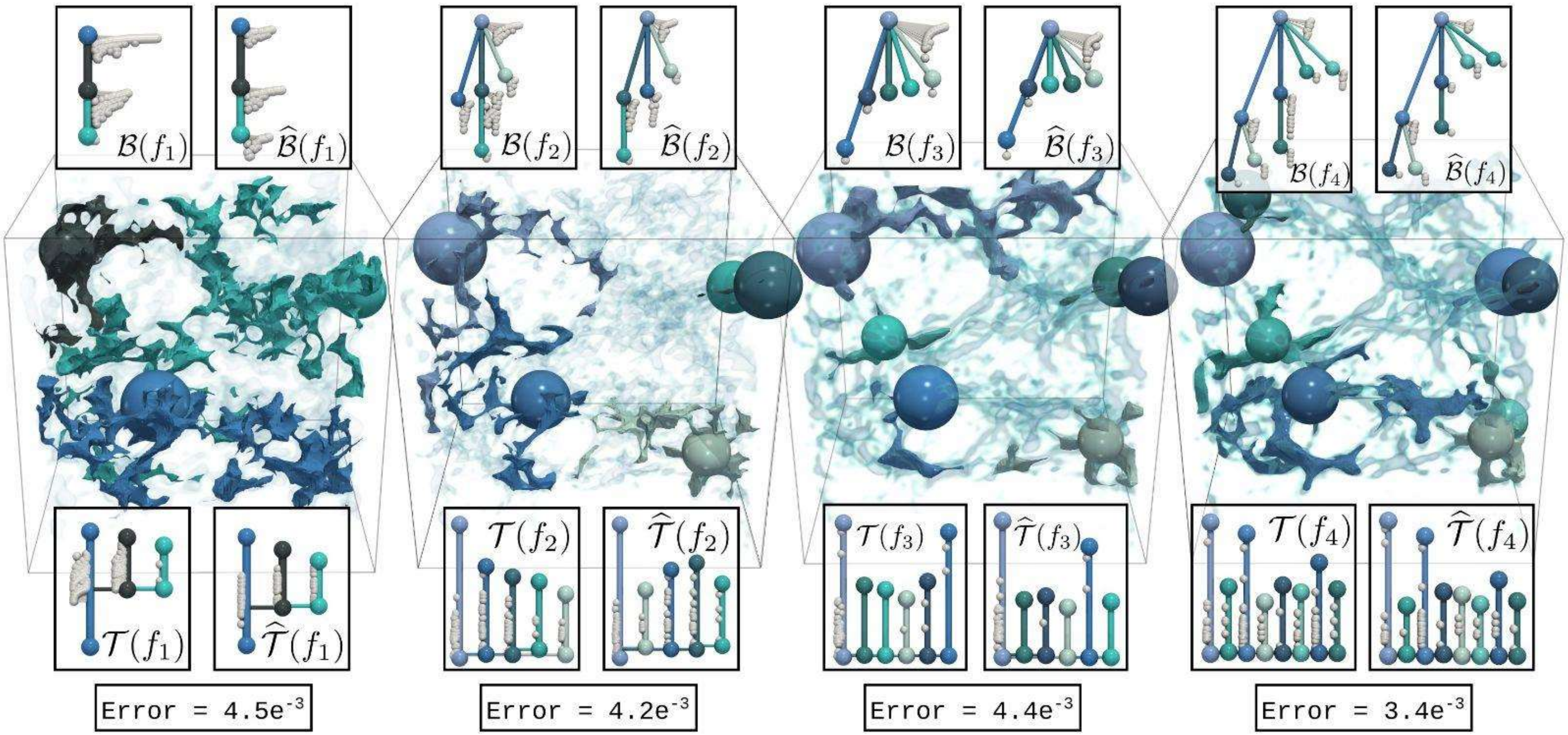
Output2: Basis geodesic axes $B_{\mathbb{B}} = \{\overleftrightarrow{\mathcal{A}_1}, \overleftrightarrow{\mathcal{A}_2}, \dots, \overleftrightarrow{\mathcal{A}_{d_{max}}}\}$;

Output3: Coordinates $\alpha^j \in [0,1]^{d_{max}}$ of the input BDTs in $B_{\mathbb{B}}$ (with $j \in \{1, 2, \dots, N\}$).

$\mathcal{B}^* \leftarrow$ WassersteinBarycenter($\mathcal{S}_{\mathcal{B}}$);
for $d' \in \{1, 2, \dots, d_{max}\}$ **do**
    $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$ InitializeGeodesicAxis($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}$);
    while $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}})$ decreases **do**
        // Optimize the current geodesic axis $\overleftrightarrow{\mathcal{A}_{d'}}$
        $\alpha_{d'}^{j \in \{1,2,\dots N\}} \leftarrow$ ProjectTrees($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}}$);
        $E_{W_2^{\mathcal{T}}}(B_{\mathbb{B}}) \leftarrow$ EvaluateFittingEnergy($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}}$);
        $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$ OptimizeFittingEnergy($\mathcal{S}_{\mathcal{B}}, \mathcal{B}^*, B_{\mathbb{B}}, \alpha^{j \in \{1,2,\dots N\}}$);
        // Enforce the constaints
        while $\overleftrightarrow{\mathcal{A}_{d'}}$ evolves **do**
            $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$ EnforceGeodesics($\mathcal{B}^*, \overleftrightarrow{\mathcal{A}_{d'}}$);
            $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$ EnforceNegativeCollinearity($\overleftrightarrow{\mathcal{A}_{d'}}$);
            $\overleftrightarrow{\mathcal{A}_{d'}} \leftarrow$ EnforceOrthogonality($B_{\mathbb{B}}, \overleftrightarrow{\mathcal{A}_{d'}}$);
        end while
    end while
end for

Data reduction

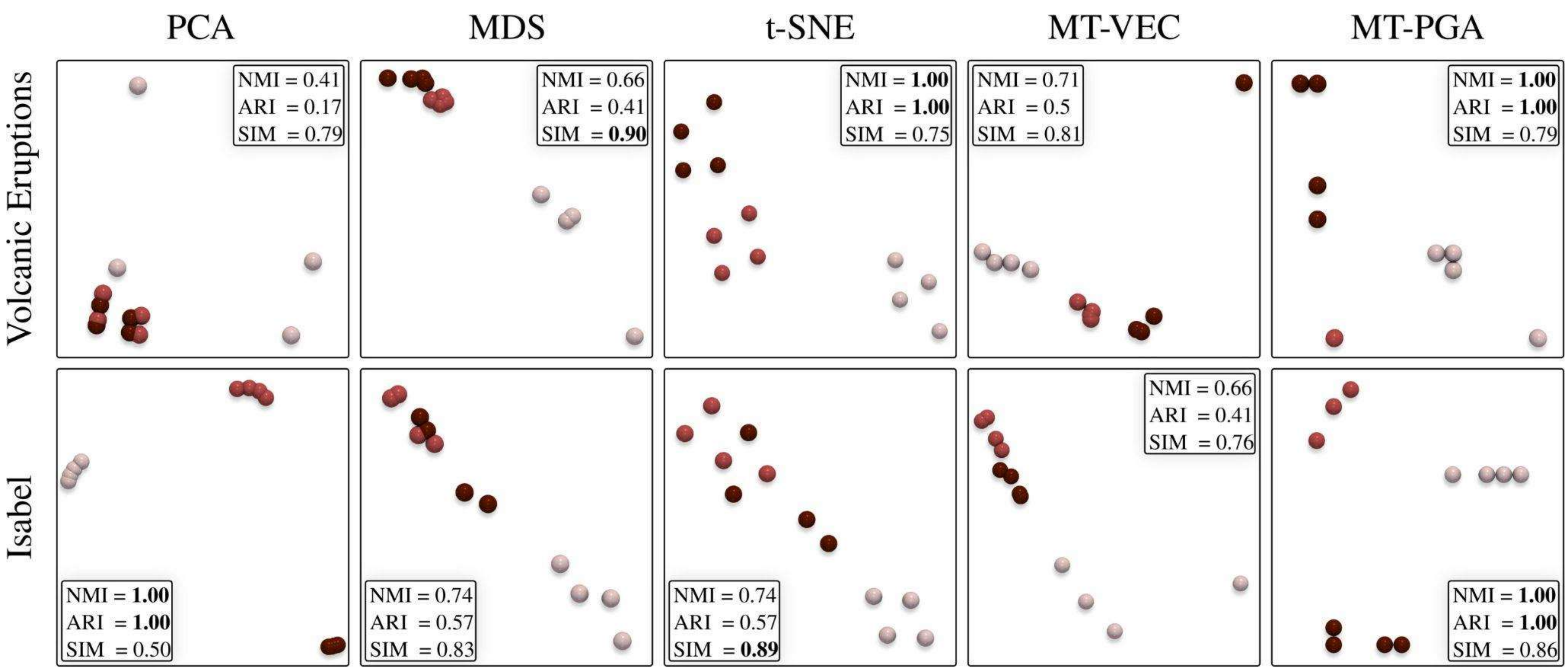
Compression factor: 19.27



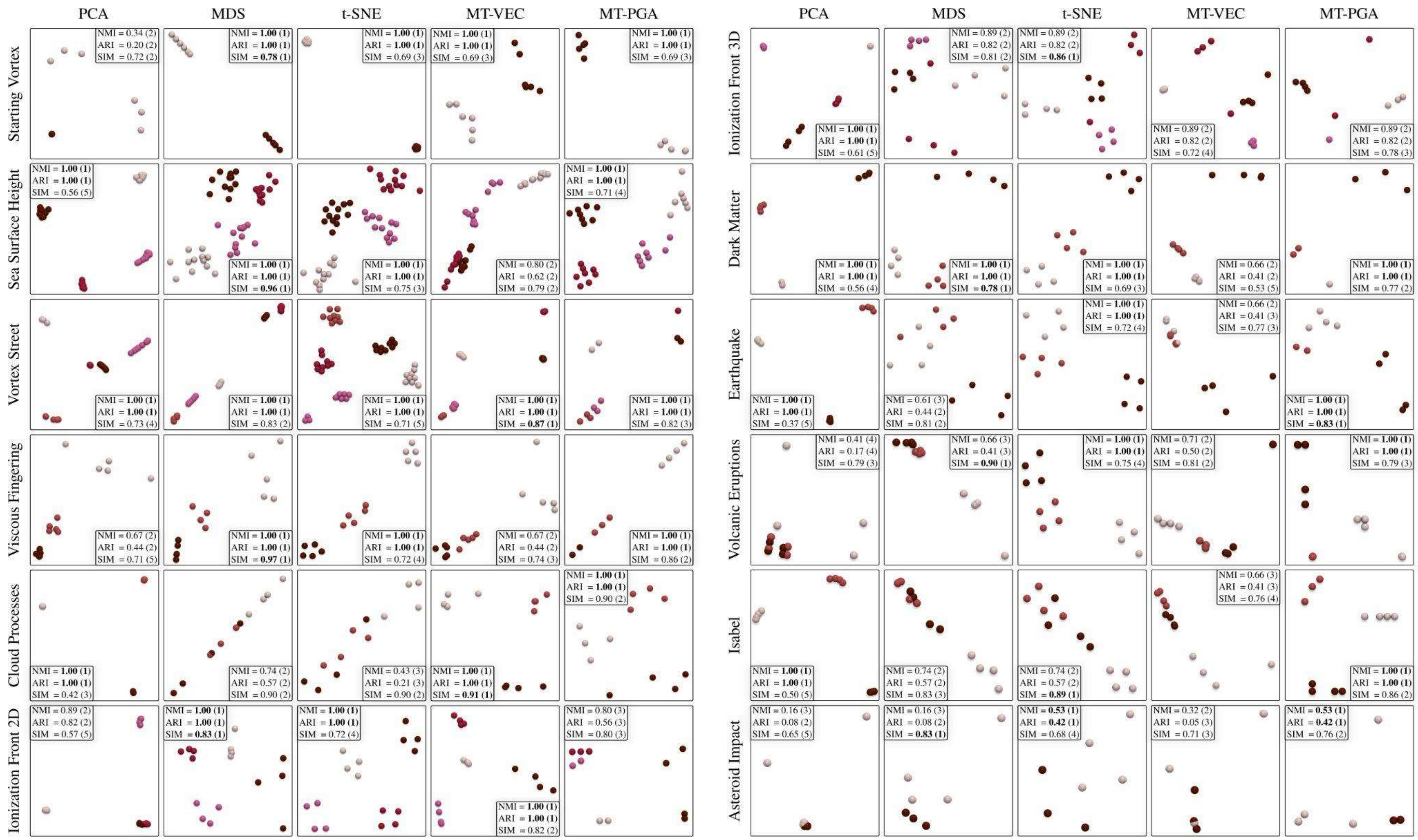
Data reduction

Dataset	N	$\|\mathcal{B}\|$	PD-PGA		MT-PGA	
			Factor	Error	Factor	Error
Asteroid Impact (3D)	7	1,295	2.97	0.07	4.84	0.22
Cloud processes (2D)	12	1,209	5.94	0.19	7.39	0.01
Viscous fingering (3D)	15	118	2.23	0.13	4.71	0.02
Dark matter (3D)	40	2,592	10.00	0.04	19.27	0.04
Volcanic eruptions (2D)	12	811	9.99	0.12	4.83	0.04
Ionization front (2D)	16	135	2.56	0.14	5.12	0.40
Ionization front (3D)	16	763	3.27	0.17	4.85	0.46
Earthquake (3D)	12	1,203	1.42	0.18	2.19	0.33
Isabel (3D)	12	1,338	5.49	0.27	9.25	0.05
Starting Vortex (2D)	12	124	1.76	0.07	4.42	0.01
Sea Surface Height (2D)	48	1,787	19.59	0.18	9.48	0.48
Vortex Street (2D)	45	23	1.86	0.04	11.84	0.02

Dimensionality reduction



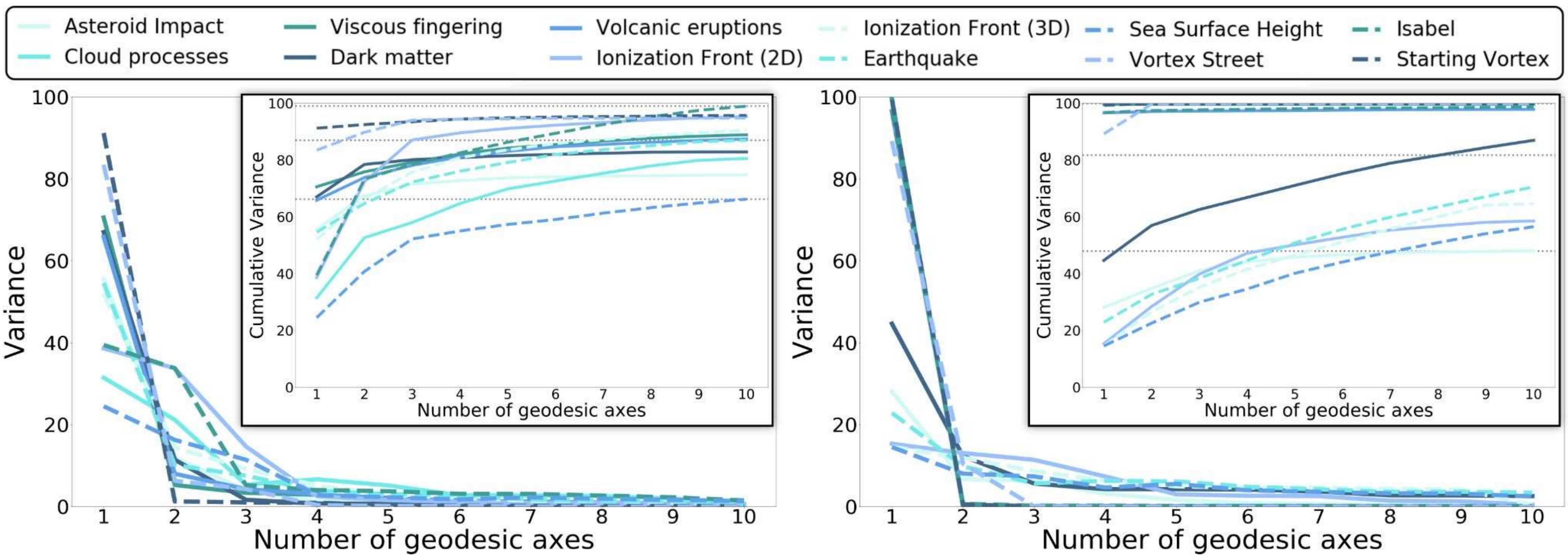
Dimensionality reduction



Dimensionality reduction

Indicator	PCA	MDS ($W_2^{\mathcal{T}}$) [66]	t-SNE ($W_2^{\mathcal{T}}$) [118]	MT-VEC	**MT-PGA**
NMI	0.79	0.82	0.88	0.78	**0.94**
ARI	0.71	0.73	0.84	0.64	**0.90**
SIM	0.60	**0.85**	0.75	0.76	0.80

Variance



PD-PGA

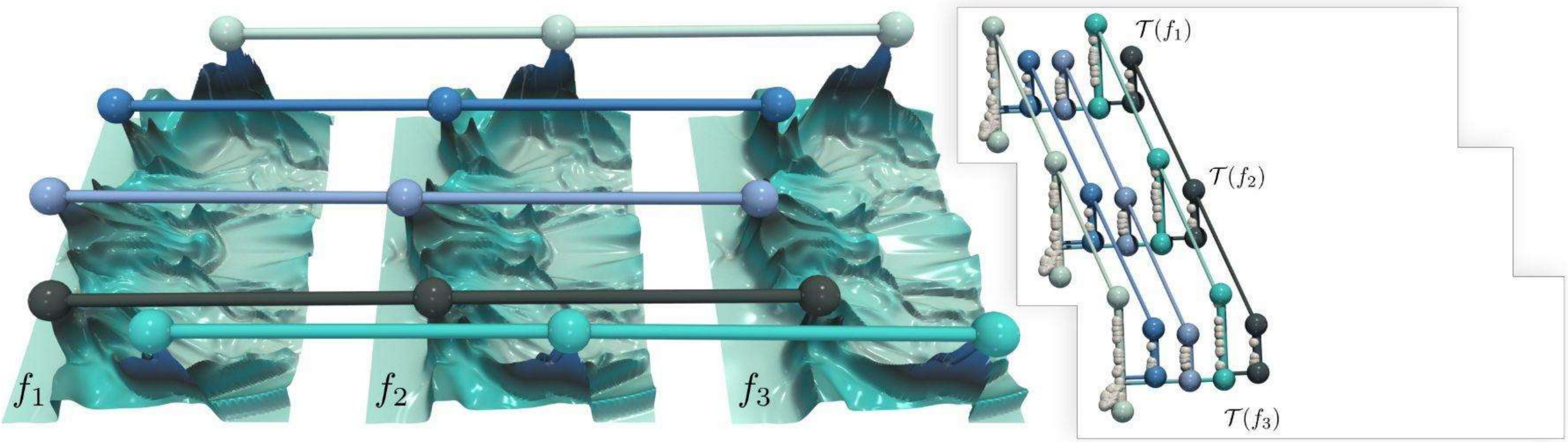
MT-PGA

Data reduction

- **Only store**
  - The basis
  - The coordinates

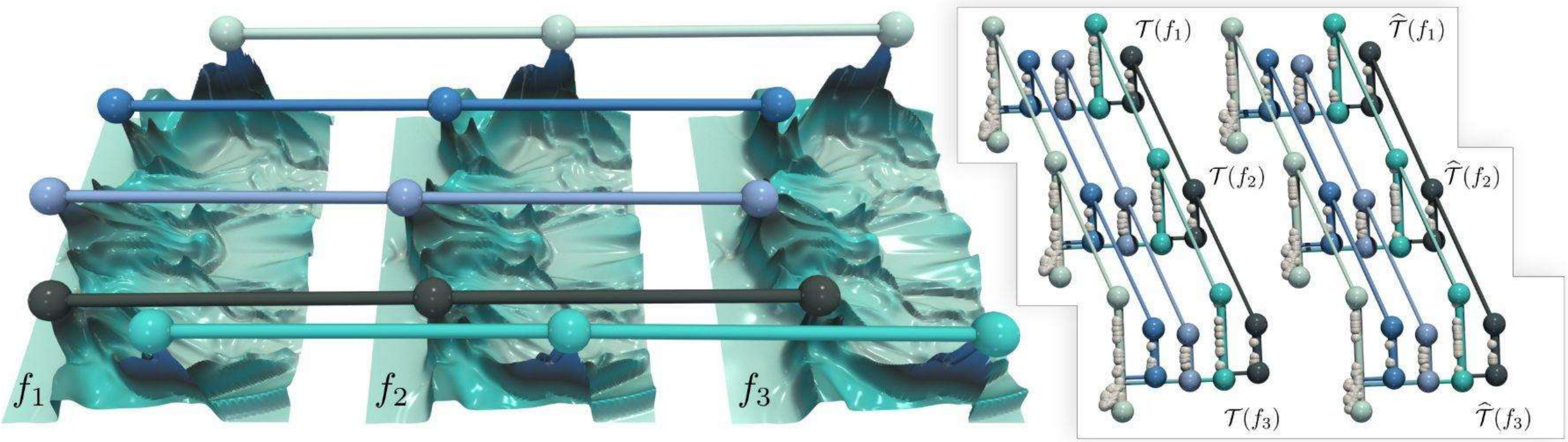
Data reduction

- **Only store**
  - The basis
  - The coordinates



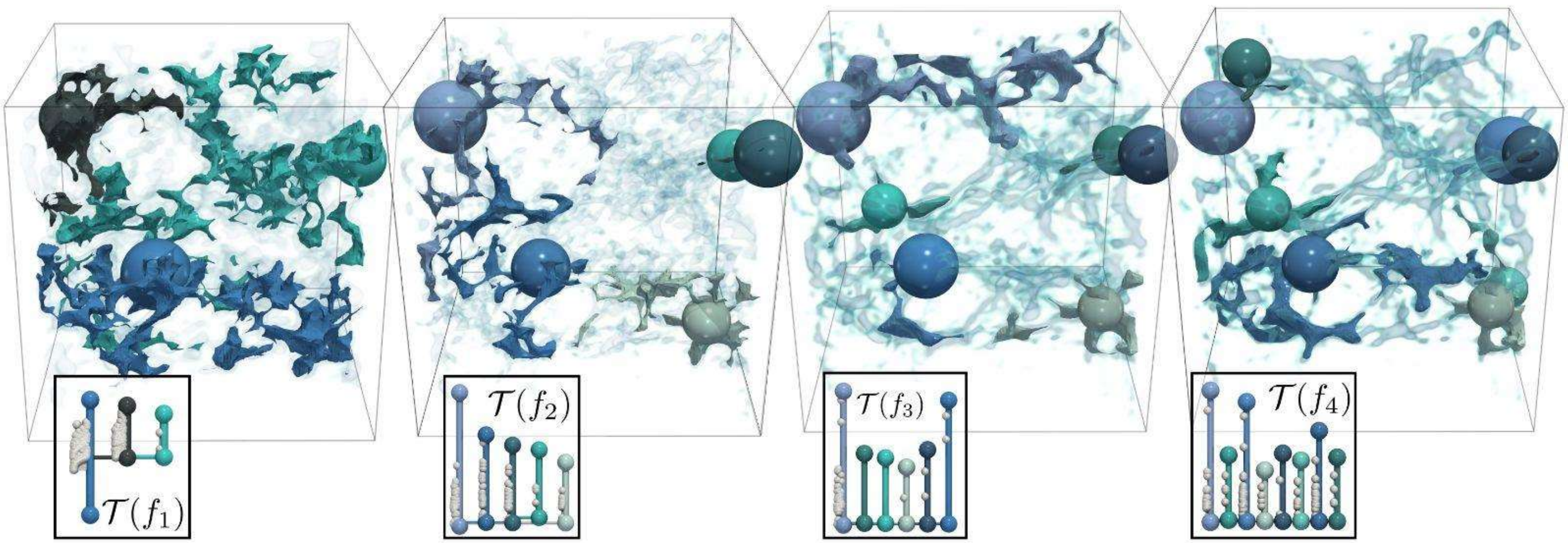
Data reduction

- **Only store**
  - The basis
  - The coordinates



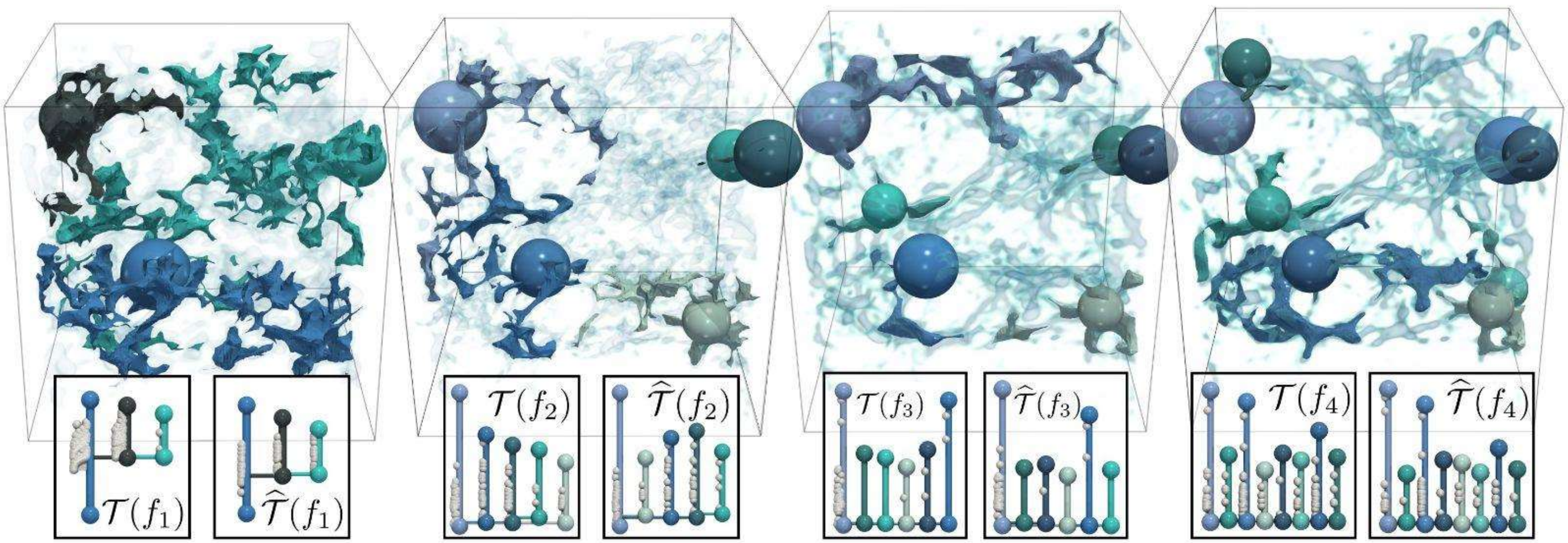
Compression factor: 5.12

Data reduction

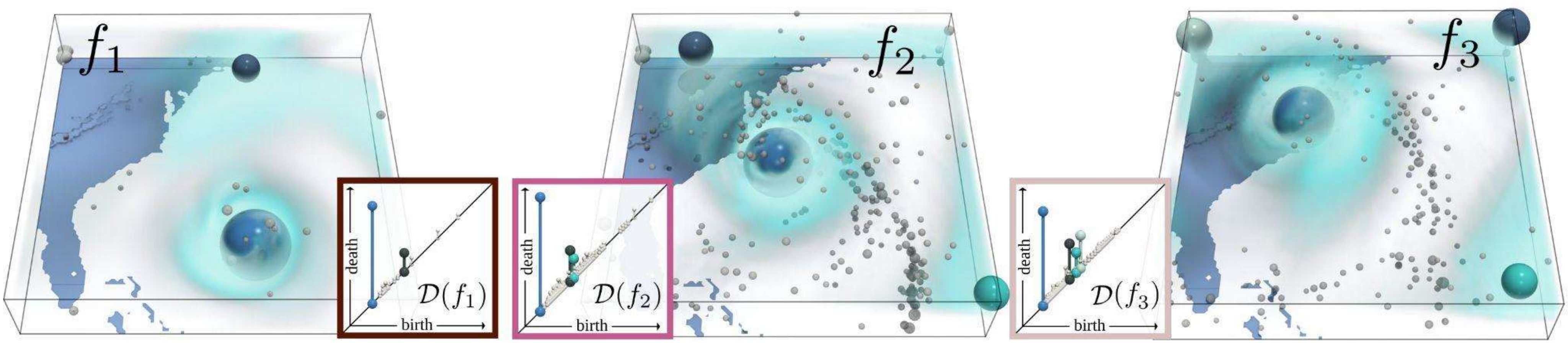


Data reduction

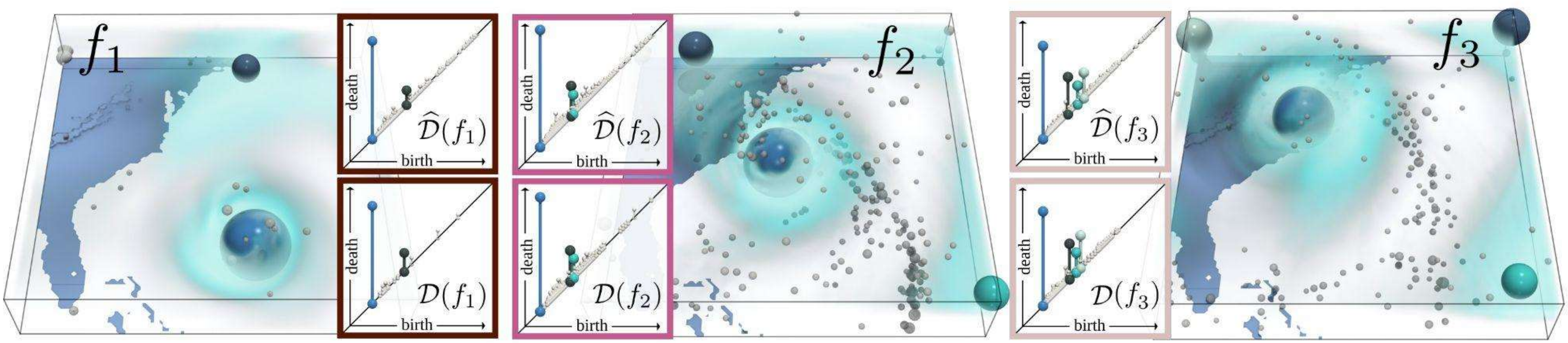
Compression factor: 19.27



PD-PGA

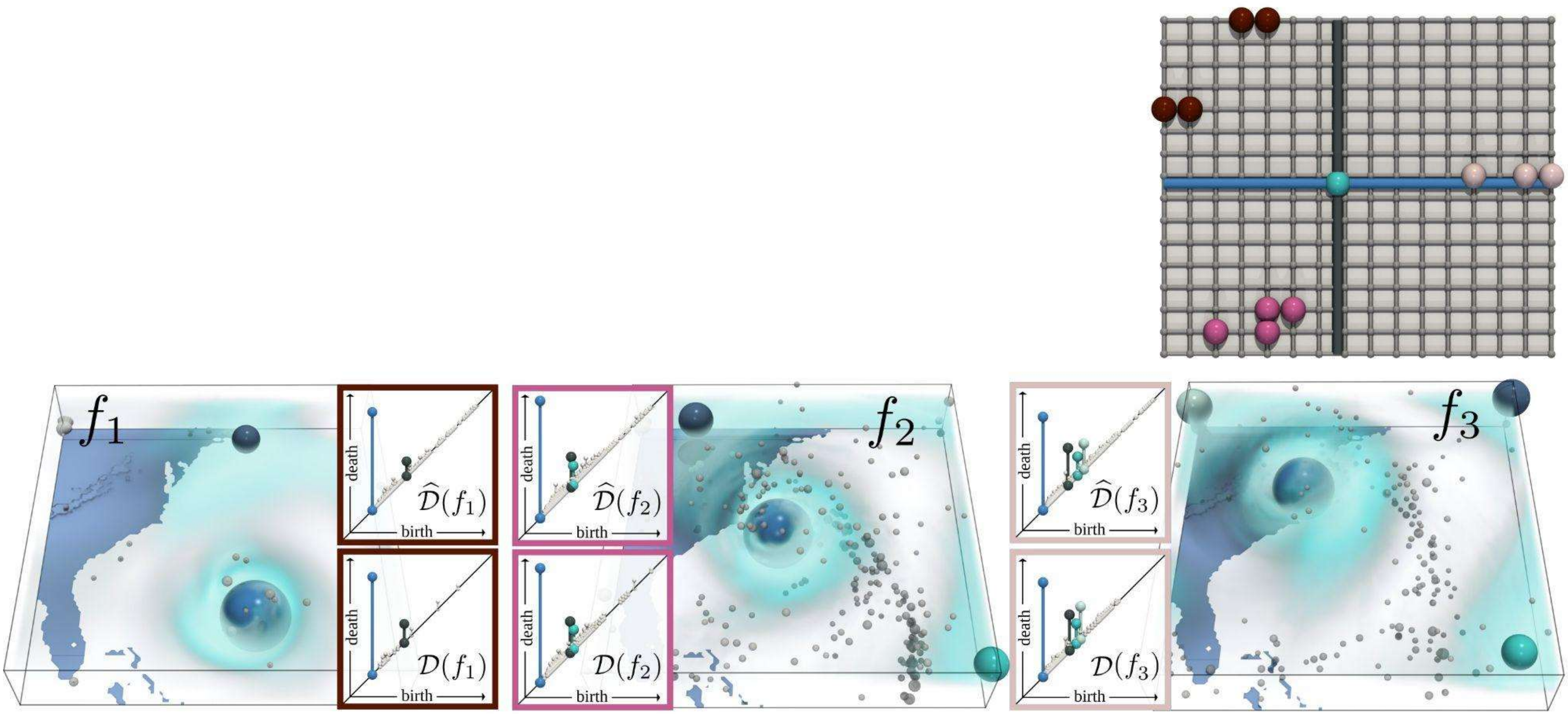


PD-PGA

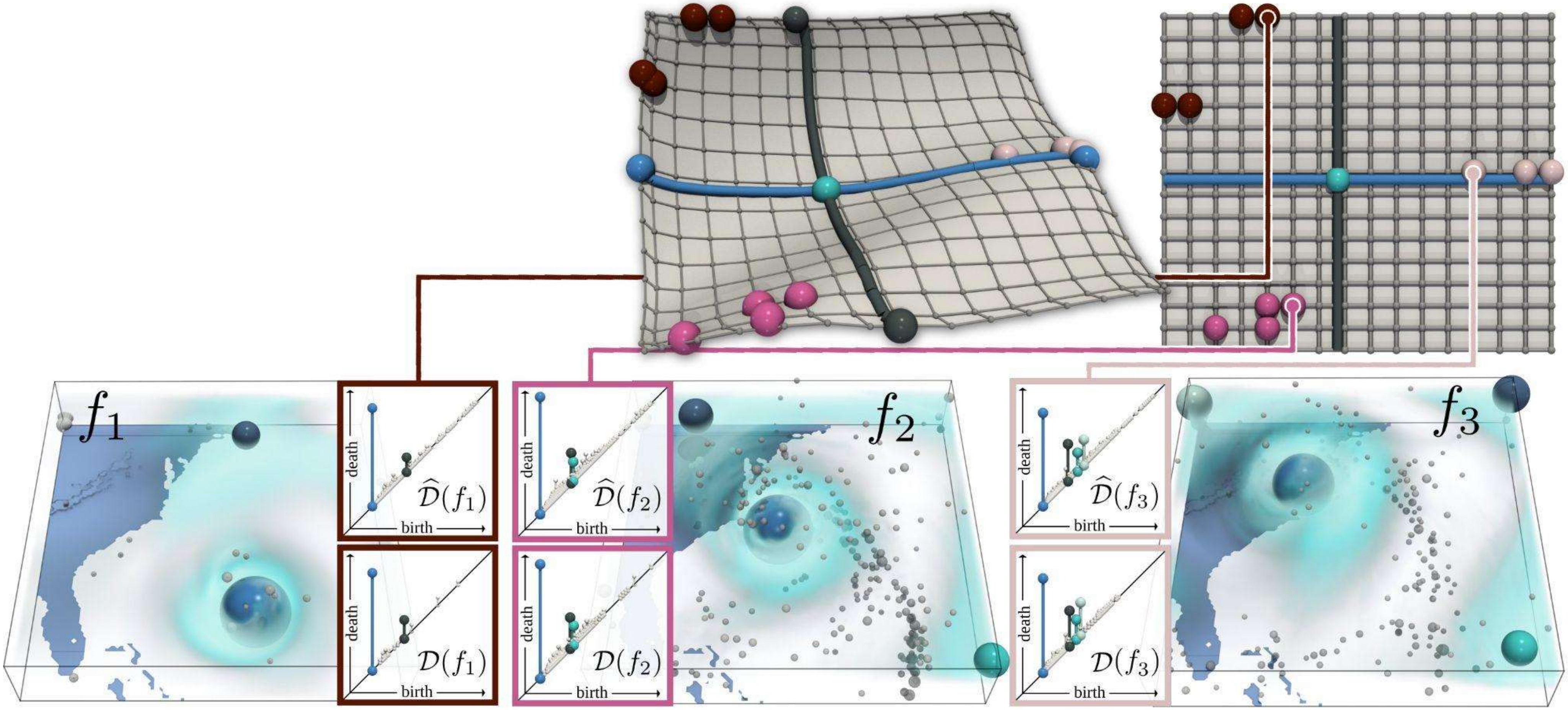


f_1
death
birth
$\widehat{\mathcal{D}}(f_1)$
$\mathcal{D}(f_1)$
$\widehat{\mathcal{D}}(f_2)$
$\mathcal{D}(f_2)$
f_2
$\widehat{\mathcal{D}}(f_3)$
$\mathcal{D}(f_3)$
f_3

PD-PGA

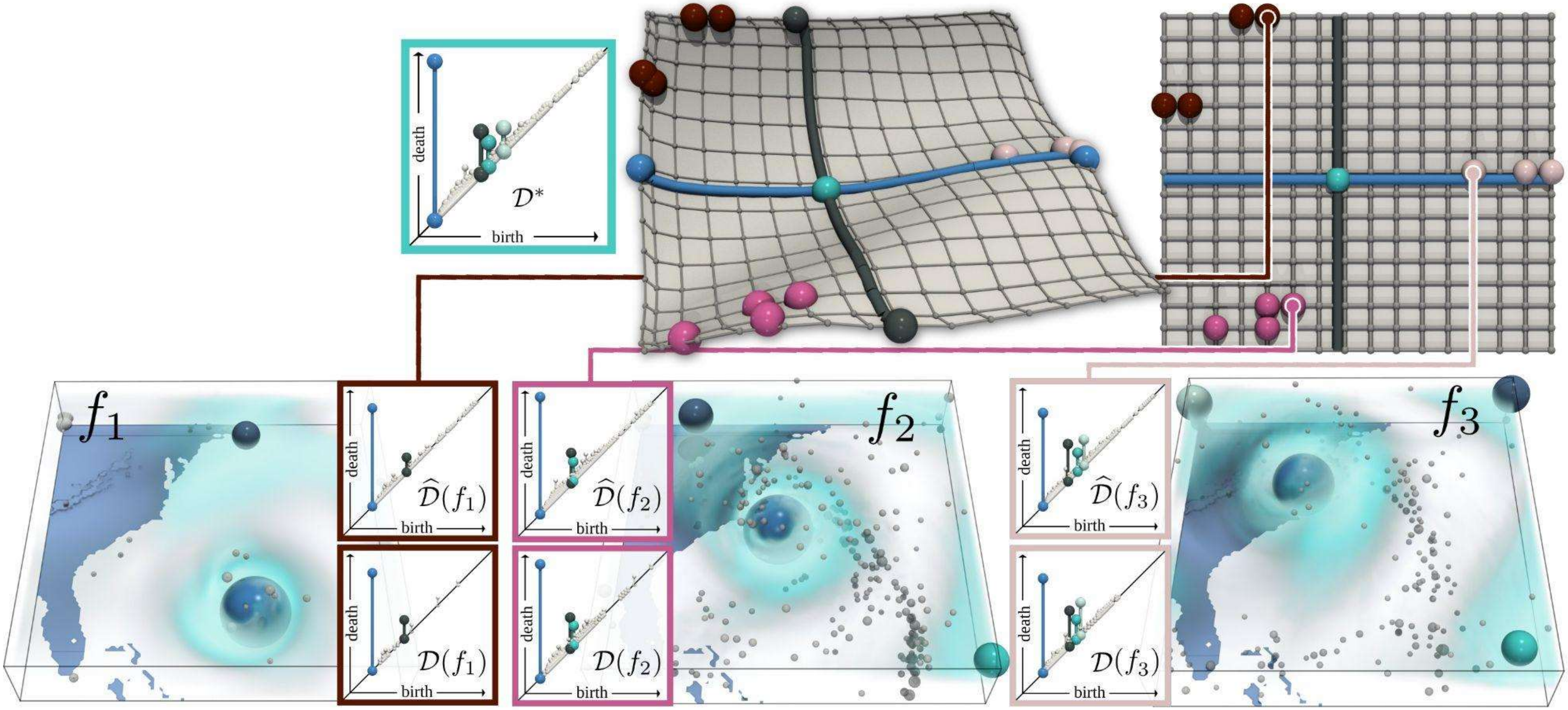

f_1
f_2
f_3
$\widehat{\mathcal{D}}(f_1)$
$\mathcal{D}(f_1)$
$\widehat{\mathcal{D}}(f_2)$
$\mathcal{D}(f_2)$
$\widehat{\mathcal{D}}(f_3)$
$\mathcal{D}(f_3)$
death
birth

PD-PGA



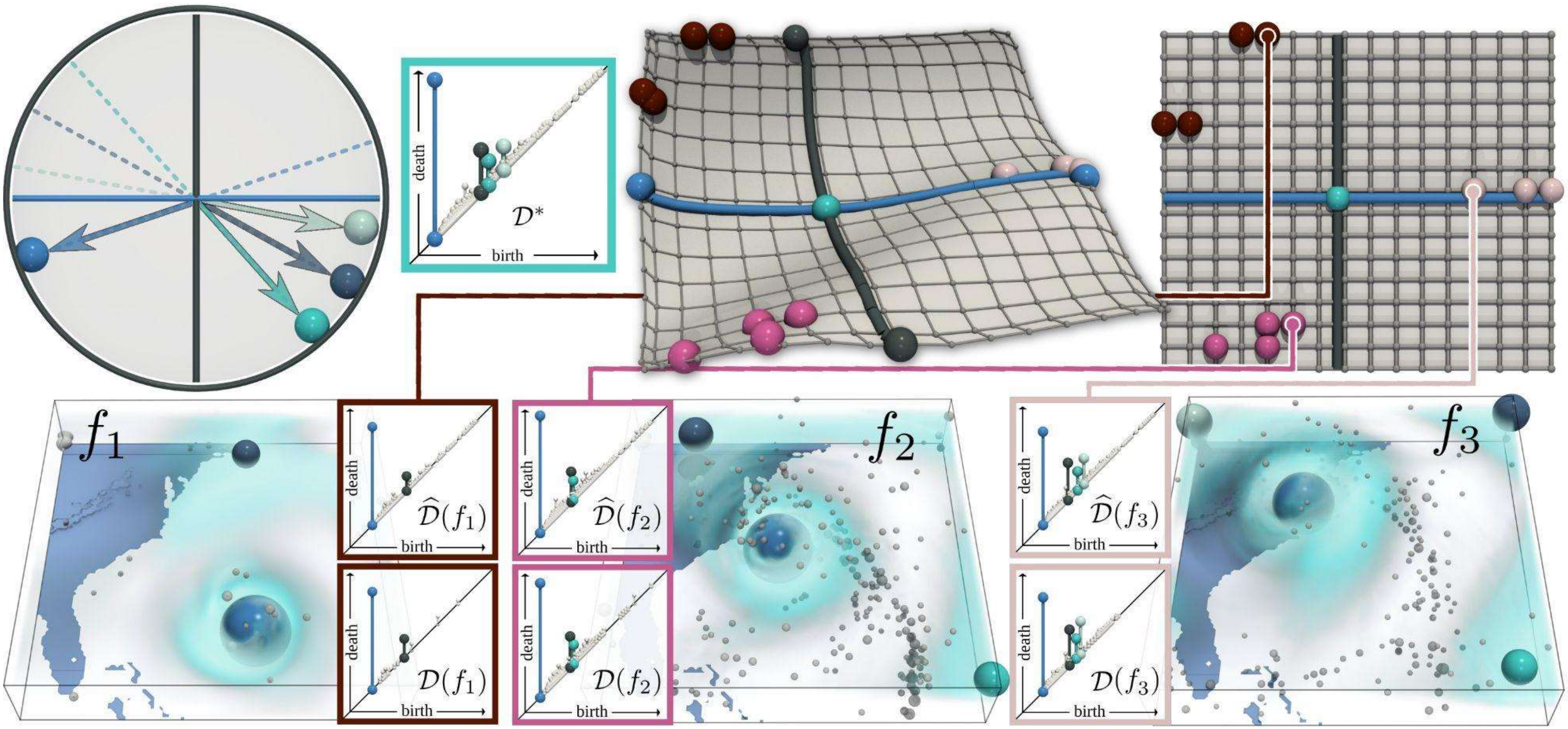
f_1
f_2
f_3
$\widehat{\mathcal{D}}(f_1)$
$\mathcal{D}(f_1)$
$\widehat{\mathcal{D}}(f_2)$
$\mathcal{D}(f_2)$
$\widehat{\mathcal{D}}(f_3)$
$\mathcal{D}(f_3)$
death
birth

PD-PGA



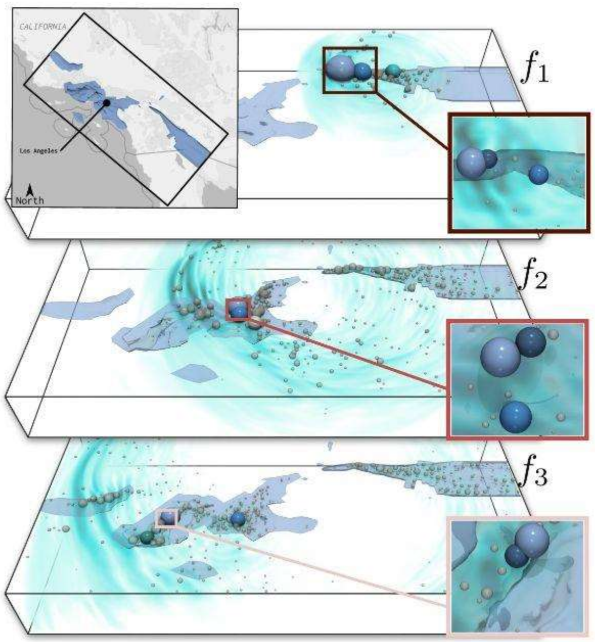
death
birth
$\mathcal{D}^*$
f_1
$\widehat{\mathcal{D}}(f_1)$
$\mathcal{D}(f_1)$
f_2
$\widehat{\mathcal{D}}(f_2)$
$\mathcal{D}(f_2)$
f_3
$\widehat{\mathcal{D}}(f_3)$
$\mathcal{D}(f_3)$

PD-PGA

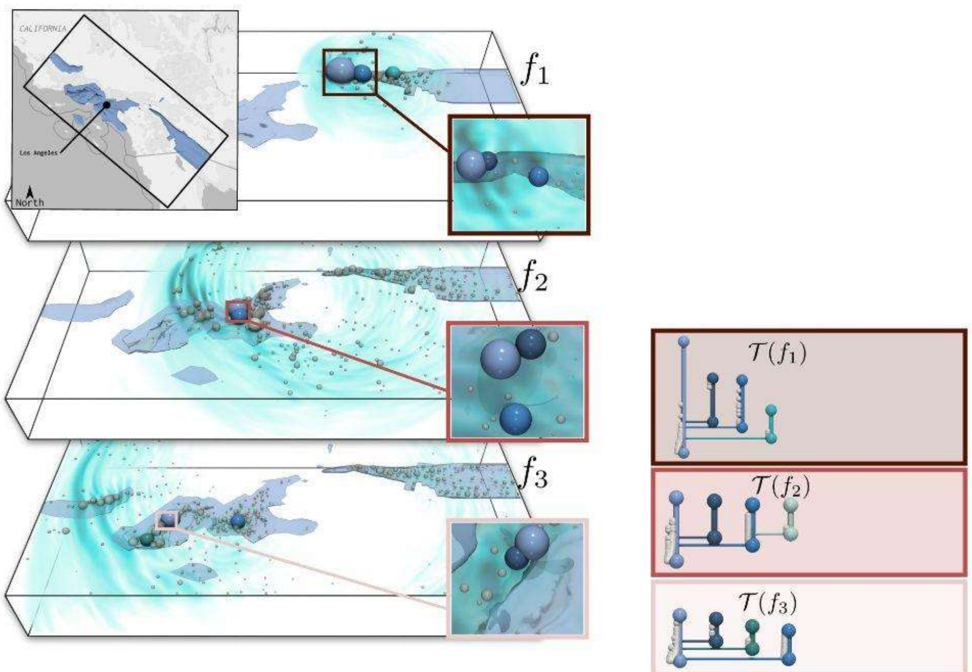


death
birth
$\mathcal{D}^*$
f_1
$\widehat{\mathcal{D}}(f_1)$
$\mathcal{D}(f_1)$
$\widehat{\mathcal{D}}(f_2)$
$\mathcal{D}(f_2)$
f_2
$\widehat{\mathcal{D}}(f_3)$
$\mathcal{D}(f_3)$
f_3

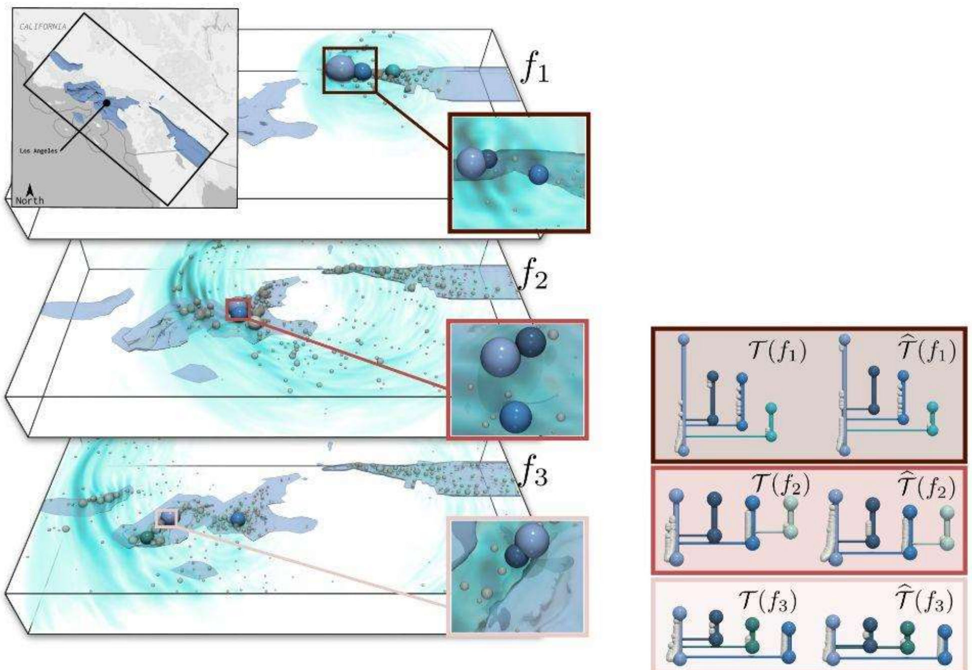
MT-PGA



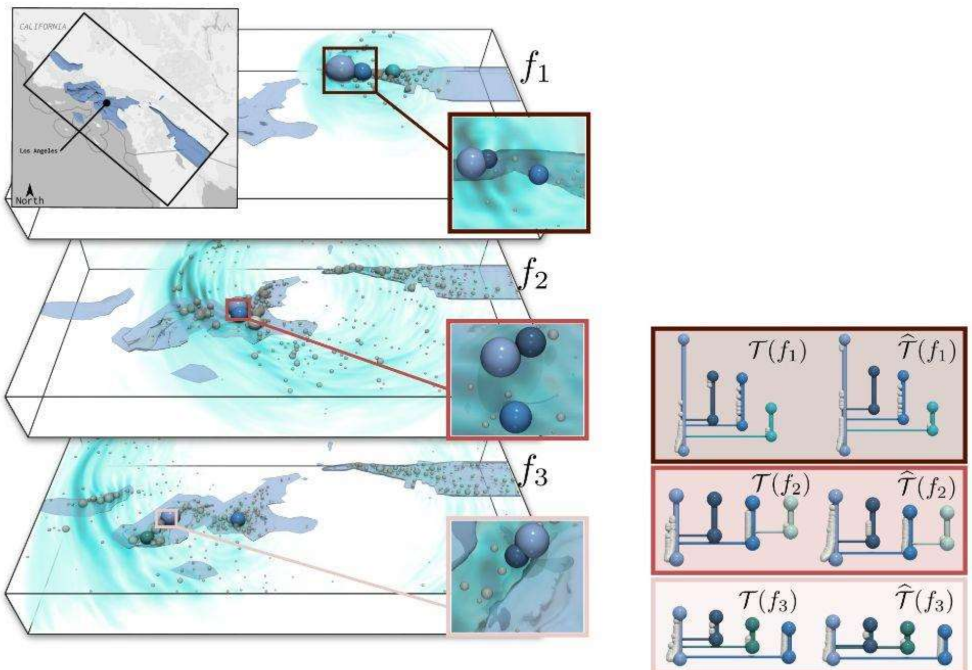
MT-PGA

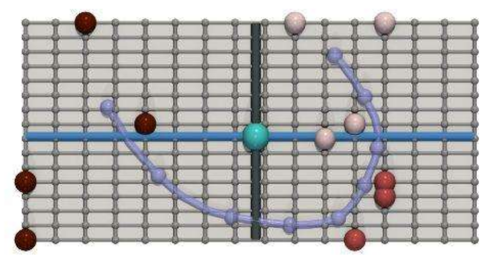

CALIFORNIA
Los Angeles
North
f_1
f_2
f_3
$\mathcal{T}(f_1)$
$\mathcal{T}(f_2)$
$\mathcal{T}(f_3)$

MT-PGA

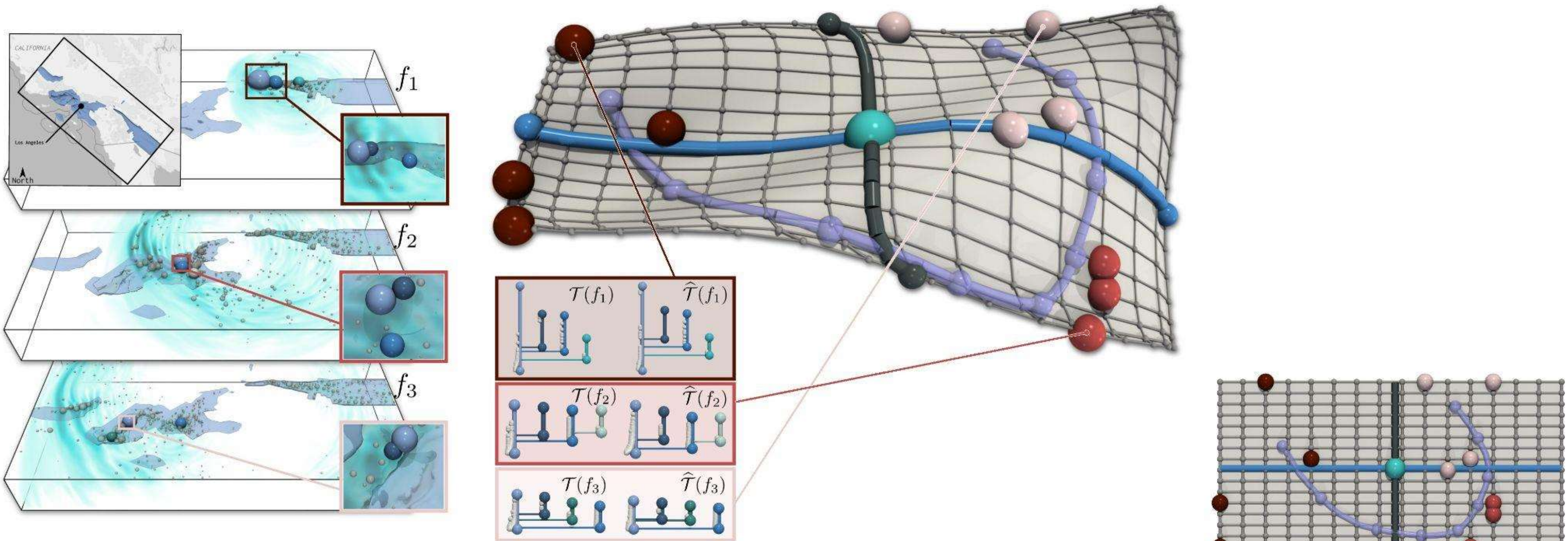

CALIFORNIA
Los Angeles
North
f_1
f_2
f_3
$\mathcal{T}(f_1)$
$\widehat{\mathcal{T}}(f_1)$
$\mathcal{T}(f_2)$
$\widehat{\mathcal{T}}(f_2)$
$\mathcal{T}(f_3)$
$\widehat{\mathcal{T}}(f_3)$

MT-PGA


CALIFORNIA
Los Angeles
North
f_1
f_2
f_3
$\mathcal{T}(f_1)$
$\widehat{\mathcal{T}}(f_1)$
$\mathcal{T}(f_2)$
$\widehat{\mathcal{T}}(f_2)$
$\mathcal{T}(f_3)$
$\widehat{\mathcal{T}}(f_3)$

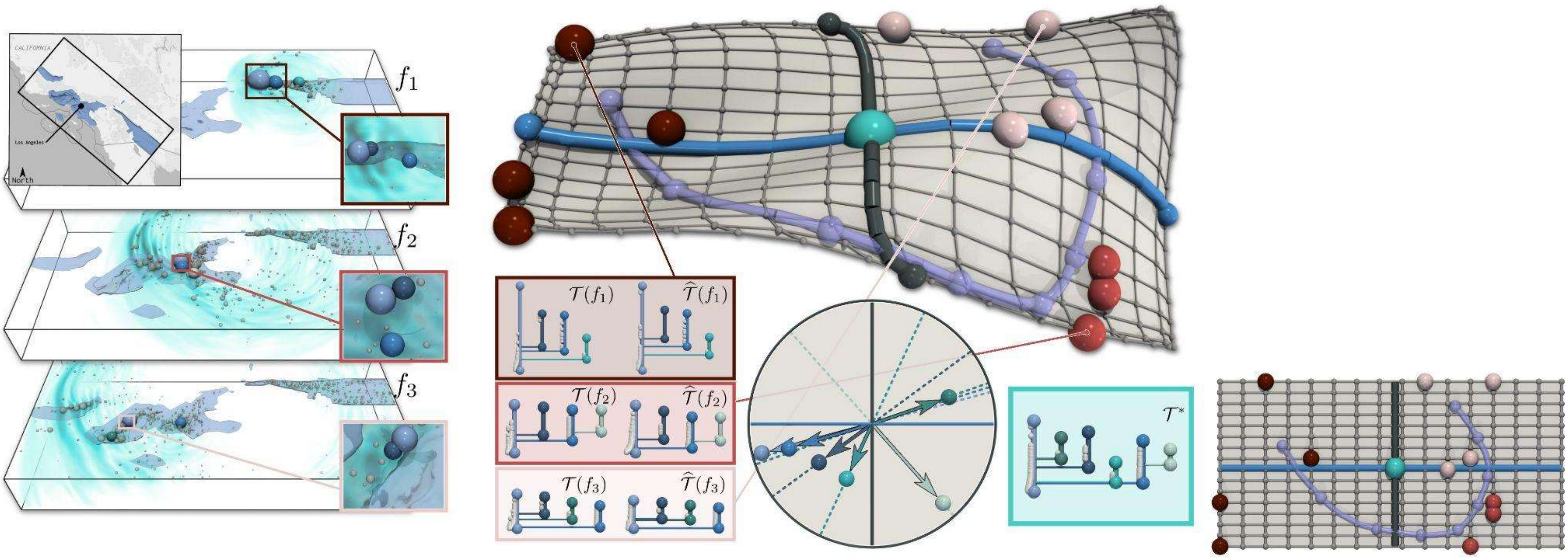


MT-PGA



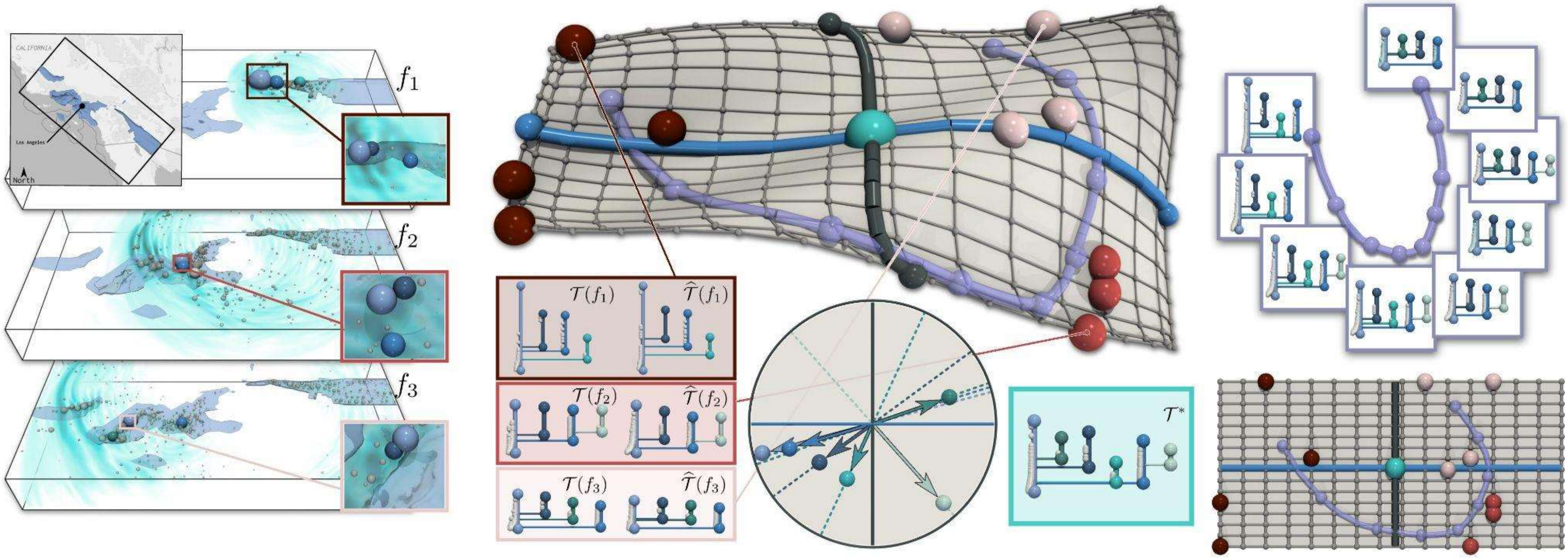
CALIFORNIA
Los Angeles
North
f_1
f_2
f_3
$\mathcal{T}(f_1)$
$\widehat{\mathcal{T}}(f_1)$
$\mathcal{T}(f_2)$
$\widehat{\mathcal{T}}(f_2)$
$\mathcal{T}(f_3)$
$\widehat{\mathcal{T}}(f_3)$

MT-PGA



CALIFORNIA
Los Angeles
North
f_1
f_2
f_3
$\mathcal{T}(f_1)$
$\widehat{\mathcal{T}}(f_1)$
$\mathcal{T}(f_2)$
$\widehat{\mathcal{T}}(f_2)$
$\mathcal{T}(f_3)$
$\widehat{\mathcal{T}}(f_3)$
$\mathcal{T}^*$

MT-PGA



CALIFORNIA
Los Angeles
North
f_1
f_2
f_3
$\mathcal{T}(f_1)$
$\widehat{\mathcal{T}}(f_1)$
$\mathcal{T}(f_2)$
$\widehat{\mathcal{T}}(f_2)$
$\mathcal{T}(f_3)$
$\widehat{\mathcal{T}}(f_3)$
$\mathcal{T}^*$