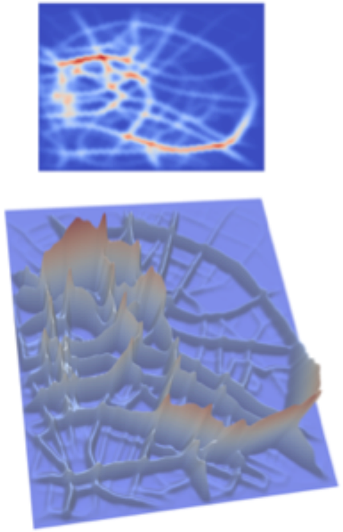
Graph Reconstruction by Discrete Morse Theory

Presented by Jiayuan Wang
Department of Computer Science and Engineering ,
The Ohio State University

Joint work with Tamal K. Dey and Yusu Wang

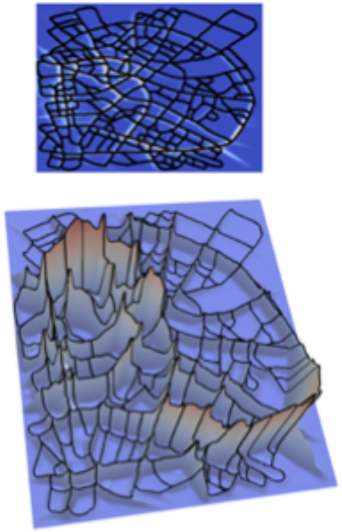
Motivation

- Recovering hidden structures from noisy data
  - Filamentary structures reconstruction [Sousbie, 2015]
  - Road network reconstruction from GPS trajectories [Wang, Li, Wang, SIGSPATIAL 2015]



Reconstructing 2-dimensional spaces

Recovering hidden structures



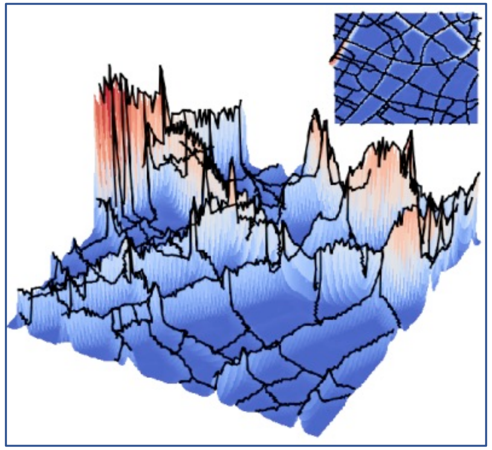
Input: a density field Sampled from GPS trajectories

Output: black lines, road network

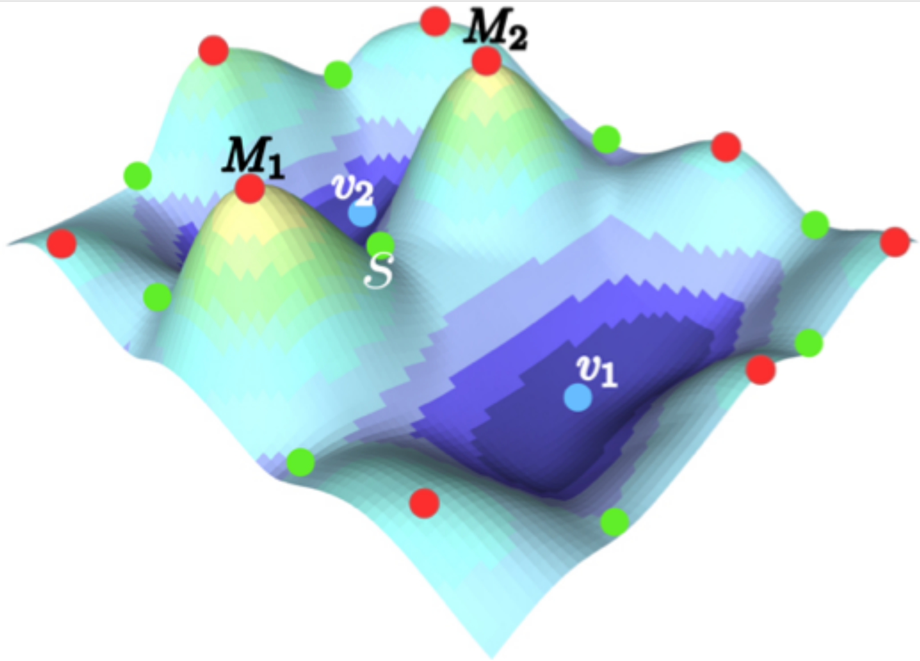
Our contributions

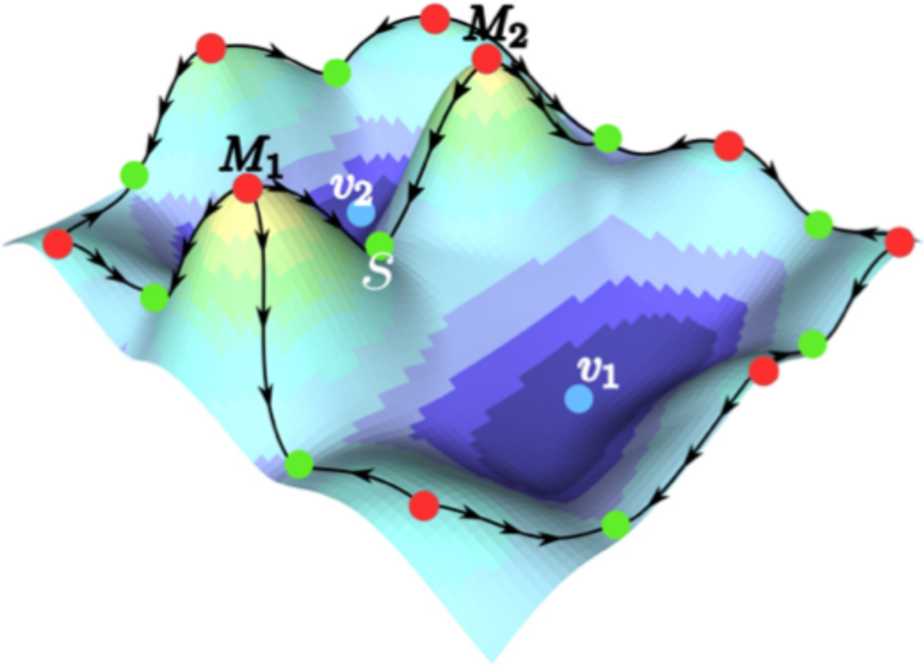
- Develop a simple persistence-guided discrete Morse-based graph reconstruction algorithm
- Obtain theoretical guarantees under a simple but reasonable noise model

Smooth case



- Input: a density field sampled from a graph
- Capture the mountain ridged by 1-stable manifold from Morse theory



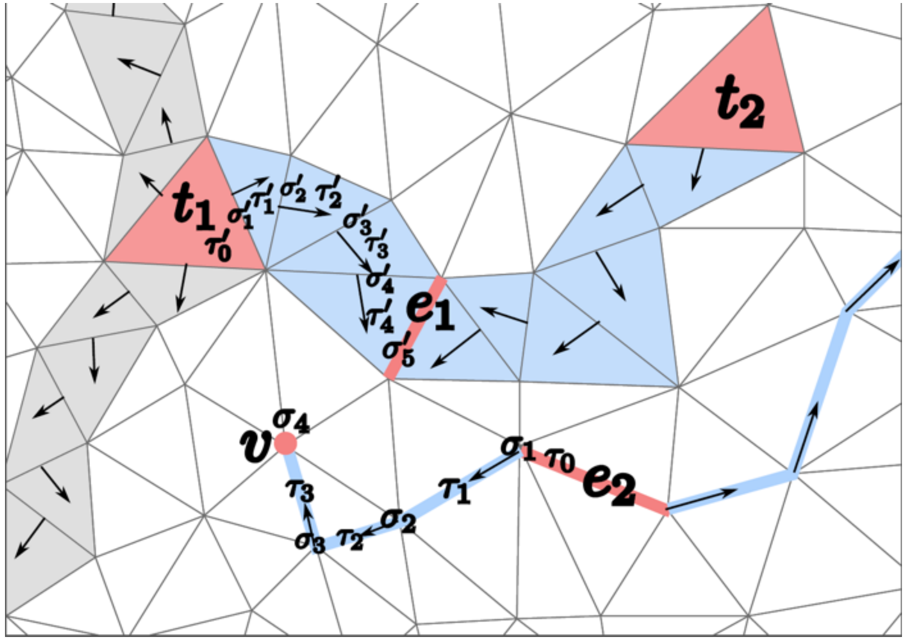


1-Stable manifold of a saddle s: integral lines flow into s

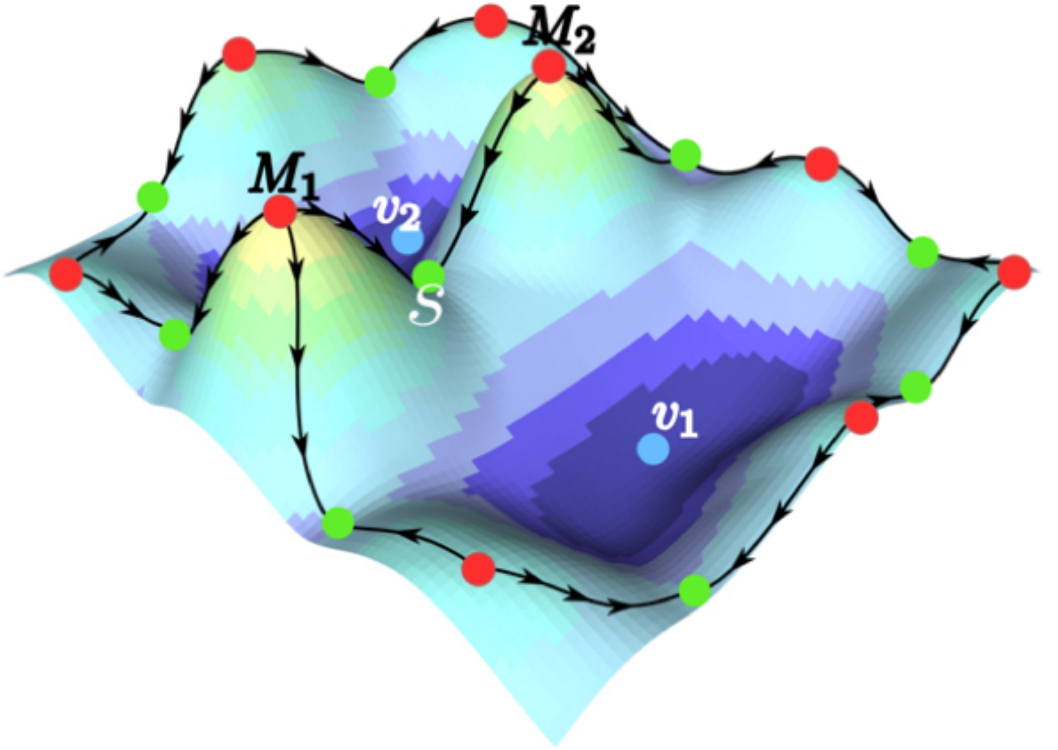
Discrete Morse theory

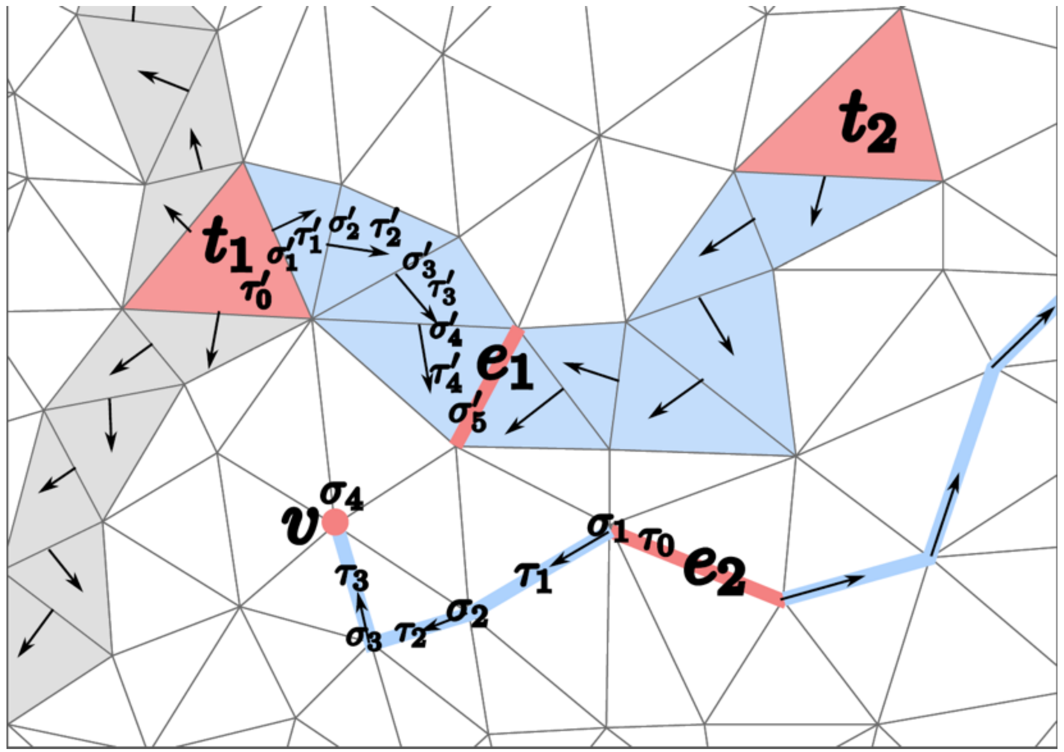
- Simplicial complex K
- A discrete (gradient) vector: (σ, τ) such that σ is a facet of τ
- A Morse pairing $M(K) = \{(\sigma, \tau)\}$, each simplex appears in at most one vector
  - A simplex is critical if it is not in any vector
- A V-path in $M(K)$
  - $\tau_0, \sigma_1, \tau_1, \sigma_2, \tau_2, \ldots, \sigma_k, \tau_k, \sigma_{k+1}$
  - cyclic: if $k > 0$, and $(\sigma_{k+1}, \tau_0) \in M(K)$
  - Acyclic -> a gradient path
- No cyclic V-path in $M(K)$ ->

A discrete gradient vector field



Smooth case and discrete case



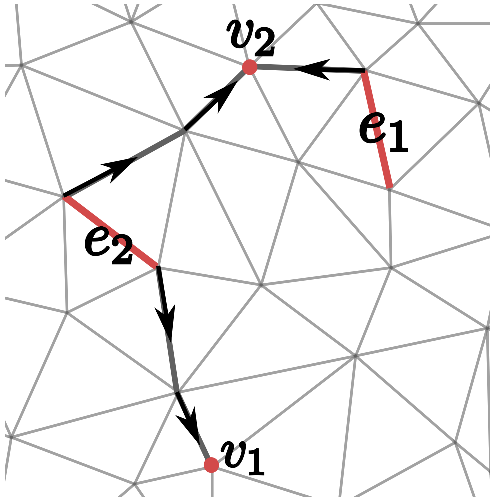


Stable manifold of a saddle s: integral lines flow into s
Saddles for function on R^2 ≈ critical edges

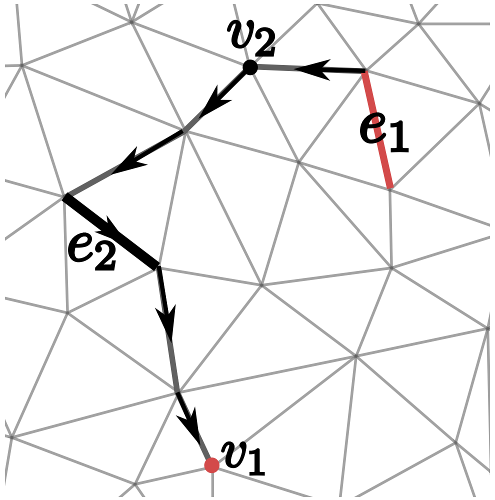
Stable manifold of an edge e: union of edge-triangle gradient paths that ends at e.
Unstable manifold of an edge e: union of vertex-edge gradient paths that begins with e.

Morse cancellation

- Simplify $M(K)$ (reducing # critical simplices)
- $< \sigma, \tau >$ is cancellable if there $\exists$ a unique gradient path connecting them
- Cancel by reversing the gradient path



Before cancellation of $< v_2, e_2 >$

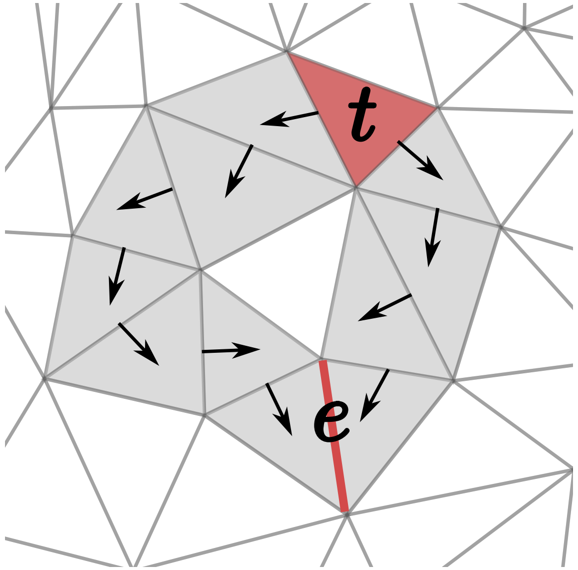


After cancellation of $< v_2, e_2 >$

After cancellation, the path between $< v_2, e_2 >$ is reversed, giving rise to a gradient path from e_1 to v_1

Morse cancellation

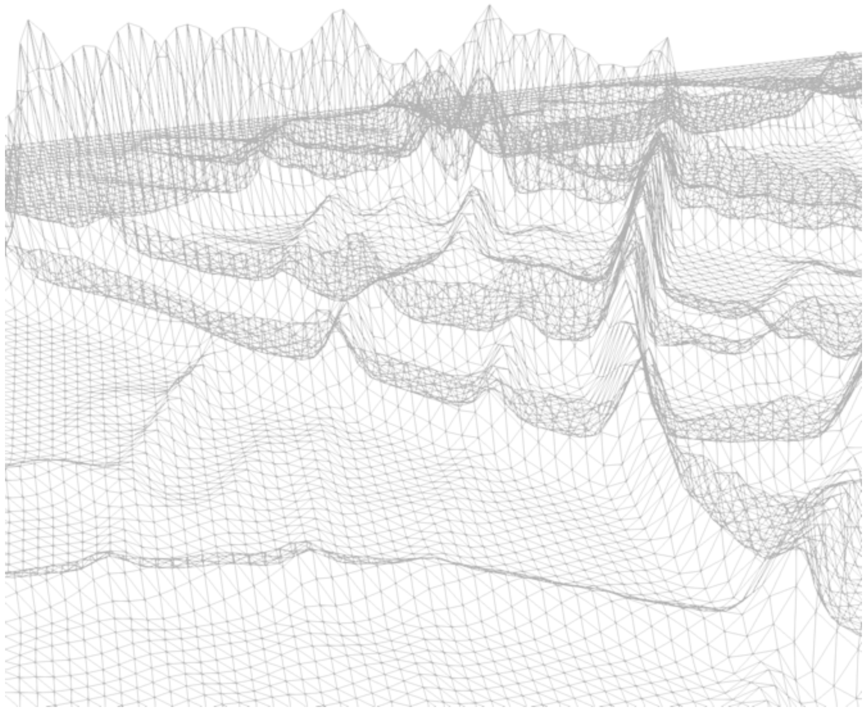
- Simplify $M(K)$ (reducing # critical simplices)
- $< \sigma, \tau >$ is cancellable if there ∃ a unique gradient path connecting them
- Cancel by reversing the gradient path



$< e, t >$ is not cancellable as there are two gradient paths between them

Existing algorithm with cancellation

- Input:
  - Triangulation K of domain $I \subset R^d$, function $f: K \to R$, threshold δ
- Initialize discrete gradient vector field W on K

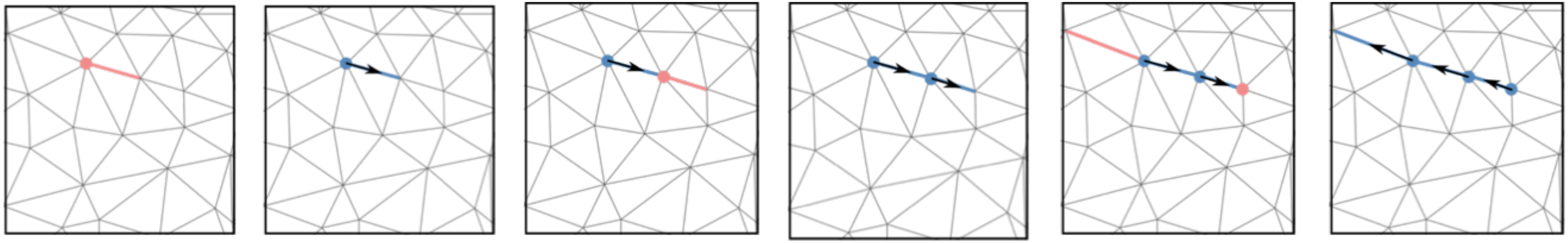


Existing algorithm with cancellation

- Input:
  - Triangulation K of domain $I \subset R^d$, function $f: K \to R$, threshold δ
- Initialize discrete gradient vector field W on K
- Step 1: *persistence computation*
  - Compute persistence pairings $P(K)$ induced by function $-f$

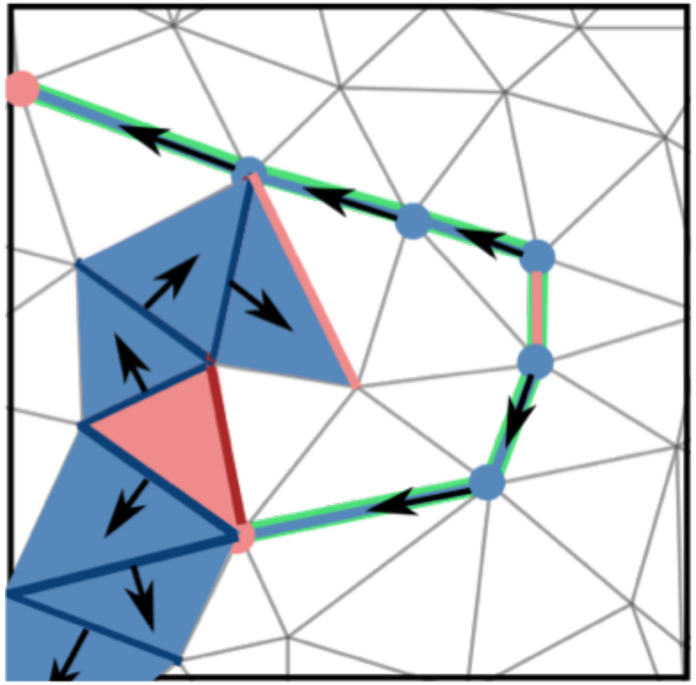
Existing algorithm with cancellation

- Input:
  - Triangulation K of domain $I \subset R^d$, function $f: K \to R$, threshold δ
- Initialize discrete gradient vector field W on K
- Step 1: *persistence computation*
- Step 2: *Morse simplification*
  - Simplify W by performing Morse cancellation for all critical pairs from $P(K)$ with persistence $\leq \delta$, if possible



Existing algorithm with cancellation

- Input:
  - Triangulation K of domain $I \subset R^d$, function $f: K \to R$, threshold δ
- Initialize discrete gradient vector field W on K
- Step 1: *persistence computation*
- Step 2: *Morse simplification*
- Step 3: *collect output*
  - For all remaining critical edges with persistence $> \delta$
  - Collect their 1-unstable manifolds and output



Existing algorithm with cancellation

- Input:
  - Triangulation K of domain $I \subset R^d$, function $f: K \to R$, threshold δ
- Initialize discrete gradient vector field W on K
- Step 1: *persistence computation*
  - Compute persistence pairings $P(K)$ induced by function $-f$
- Step 2: *Morse simplification*
  - Simplify W by performing Morse cancellation for all critical pairs from $P(K)$ with persistence $\leq \delta$, if possible
- Step 3: *collect output*
  - For all remaining critical edges with persistence $> \delta$
  - collect their 1-unstable manifolds and output

Simplified algorithm

- Step 1: *persistence computation*
- Step 2: (Morse simplification) is replaced by

Procedure PerSimpTree$(P(K), \delta)$

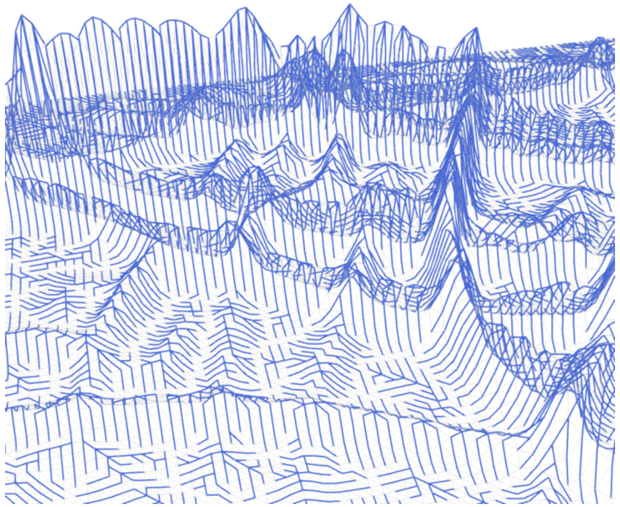
$\Pi :=$ the set of vertex-edge persistence pairs from $P = P(K)$

Set $\Pi_{\leq\delta} \subseteq \Pi$ to be $\Pi_{\leq\delta} = \{(v, e) \in \Pi \mid \mathrm{pers}(v, e) \leq \delta\}$

$\mathcal{T} := \bigcup_{(v,\sigma)\in\Pi_{\leq\delta}} \{\sigma = \langle u_1, u_2\rangle, u_1, u_2\}$

return $(\mathcal{T})$

- Step 3: *collect output*



$\Pi_{\leq\delta} = \{(v, e) \in \Pi \mid \mathrm{pers}(v, e) \leq \delta\}$

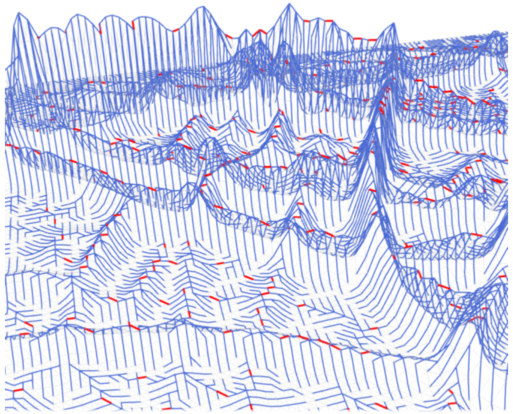
Spanning forest

Simplified Algorithms

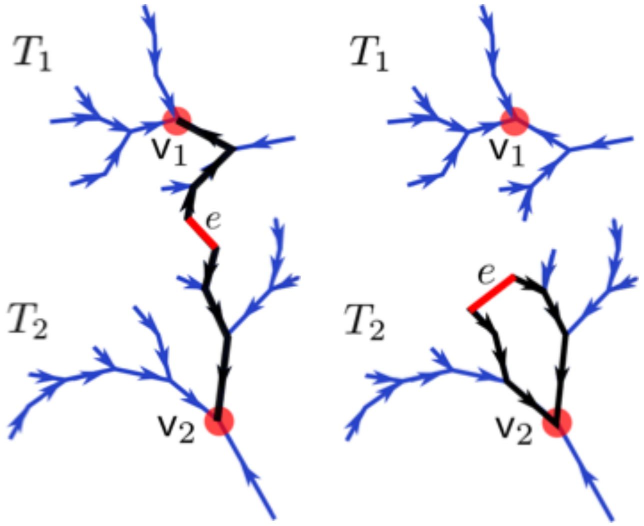
- Step 1: *persistence computation*
- Step 2: build Spanning forest
- Step 3: *collect output*

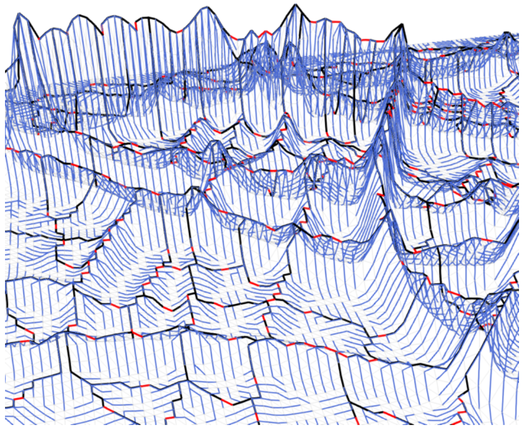
Procedure Treebased-OutputG($\mathcal{T}$)

for *each critical edge* $e = \langle u, v \rangle$ *with* $\mathrm{pers}(e) \geq \delta$ **do**
  Let $\pi(u)$ be the unique path from u to the sink of the tree T_i containing u
  Define $\pi(v)$ similarly; Set $\widehat{G} = \widehat{G} \cup \pi(u) \cup \pi(v) \cup \{e\}$



Red edges: $pers(e) > \delta$





Output: red and black edges

Existing algorithm and Simplified algorithm

Algorithm 1: MorseRecon(K,ρ, δ)

Data: Triangulation K of Ω, density function $\rho : K \to \mathbb{R}$, threshold δ

n: total number of vertices and edges in K

Result: Reconstructed graph $\widehat{G}$

begin

- Compute persistence pairings $P(K)$ by the lower-star filtration of K w.r.t $g_\rho = -\rho$
- M =PerSimpVF($P(K), \delta$) $O(n^2)$
- $\widehat{G}$ = CollectOutputG(M)
- **return** $\widehat{G}$

The same output

Algorithm 2: MorseReconSimp(K,ρ, δ)

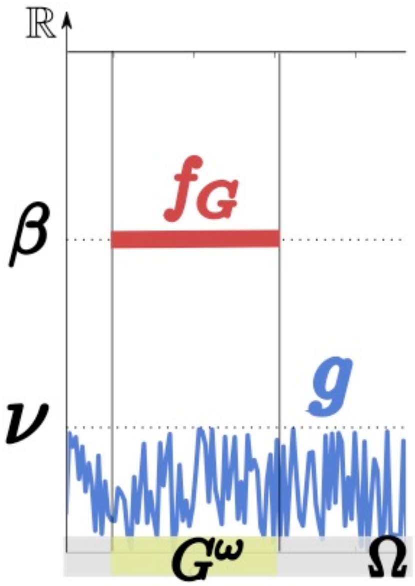
Data: Triangulation K of Ω, density function $\rho : K \to \mathbb{R}$, threshold δ

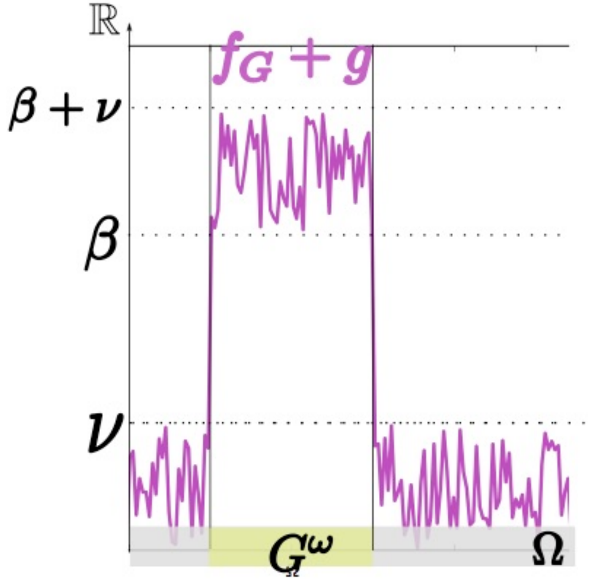
Result: Reconstructed graph $\widehat{G}$

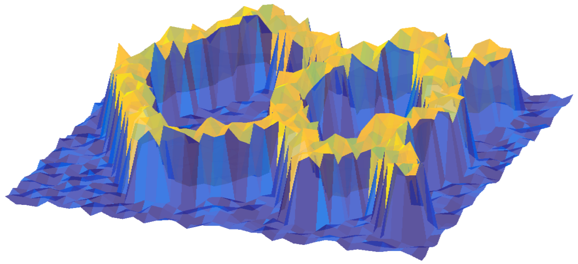
begin

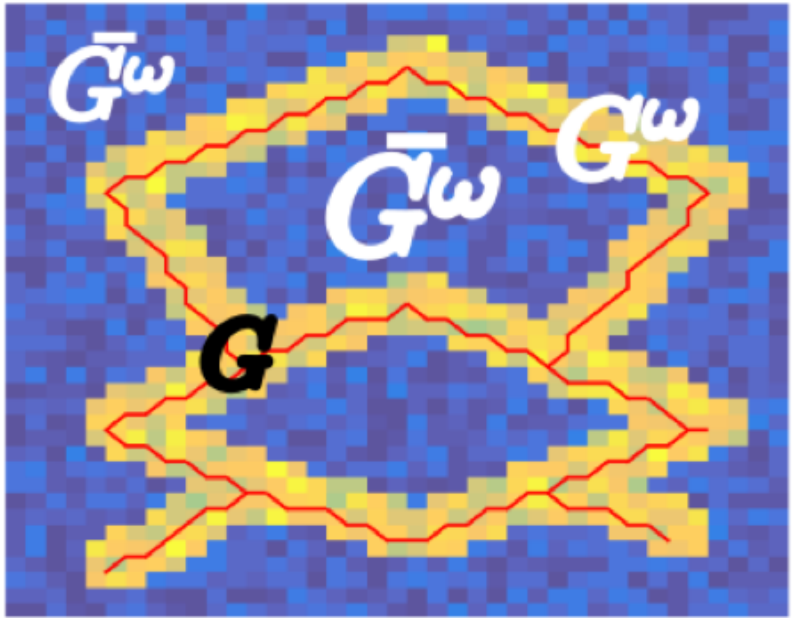
- Compute persistence pairings $P(K)$ by the lower-star filtration of K w.r.t $g_\rho = -\rho$
- $\mathcal{T}$=PerSimpTree($P(K), \delta$) $O(n)$
- $\widehat{G}$ = Treebased-OutputG($\mathcal{T}$)
- **return** $\widehat{G}$

Noise Model









Definition A density function $\rho: \Omega \to \mathbb{R}$ is a $(\beta, \nu, \omega) - approximation$ of a connected graph G if the following holds:

C-1 There is a ω-neighborhood G^ω of G such that G^ω deformation retracts to G.

C-2 $\rho(x) \in [\beta, \beta + \nu]$ for $x \in G^\omega$; and $\rho(x) \in [0, \nu]$ otherwise. Furthermore, $\beta > 2\nu$.

Theoretical results- any dimension

Under our noise model:

1. The input is a $(\beta, \nu, \omega) - approximation$ density field w.r.t G
2. $\delta \in [\nu, \beta - \nu)$

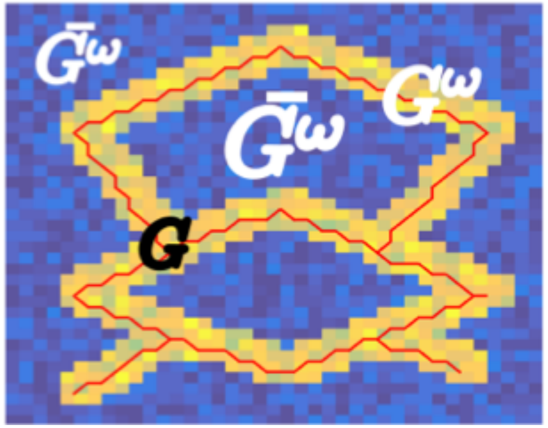
$\hat{G}$: the output of the reconstruction.

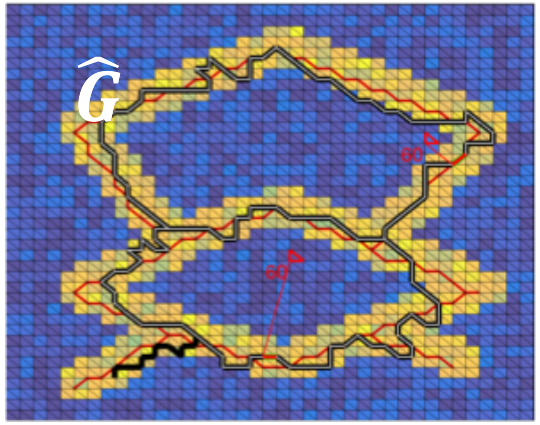
$\hat{G}$ is geometrically close to G:

Theorem Under our noise model, the output graph satisfies $\hat{G} \subseteq G^{\omega}$.

$\hat{G}$ is topologically close to G:

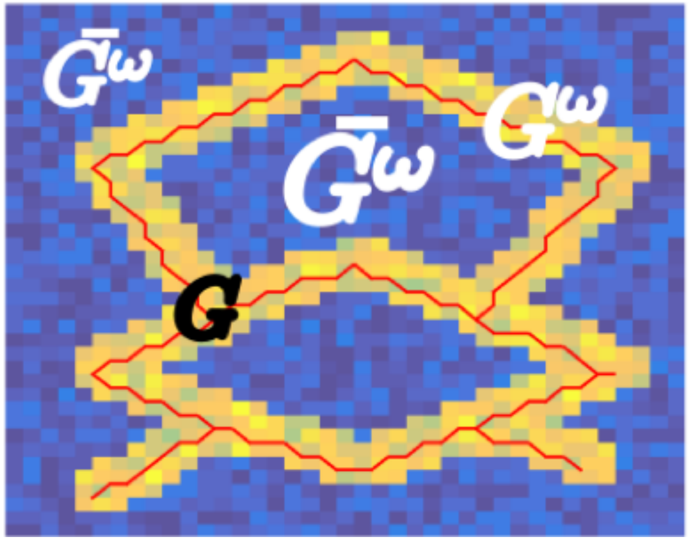
Proposition Under our noise model, $\hat{G}$ is homotopy equivalent to G.

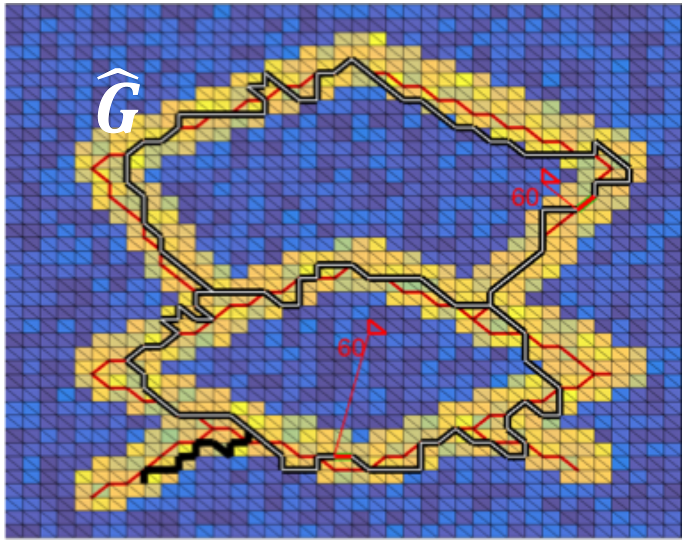




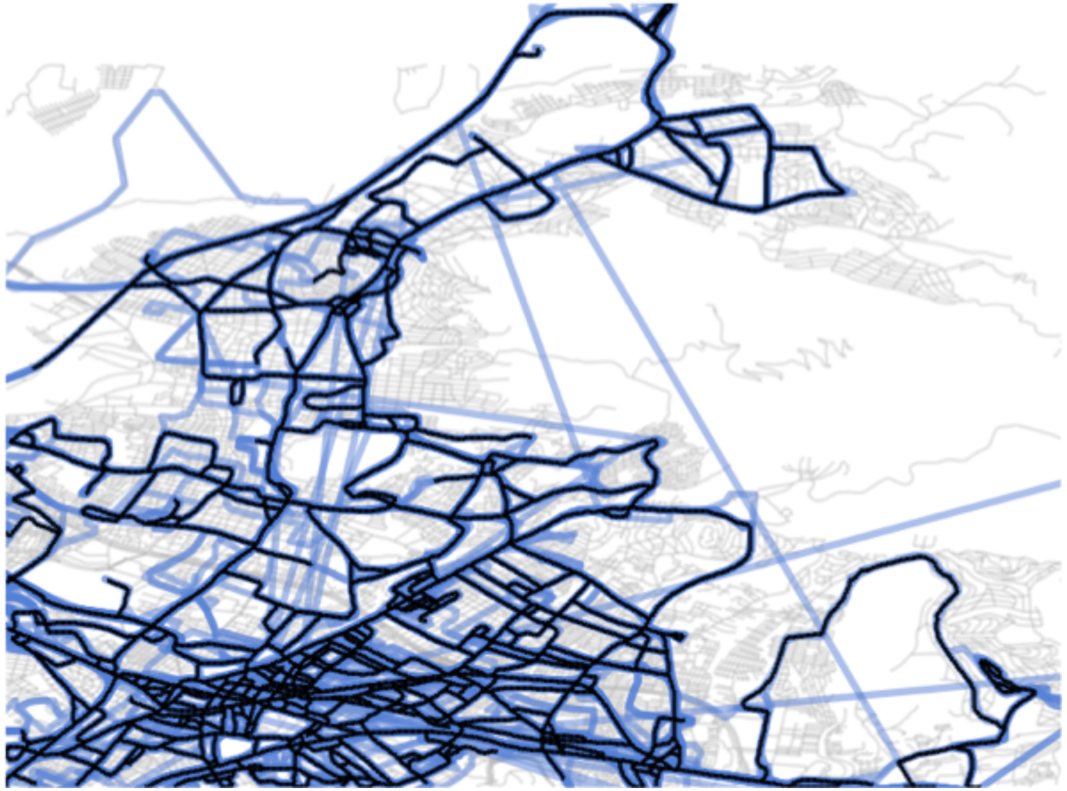
Theoretical results- 2D

Theorem Under our noise model, G^{ω} deformation retracts to G and $\widehat{G}$.

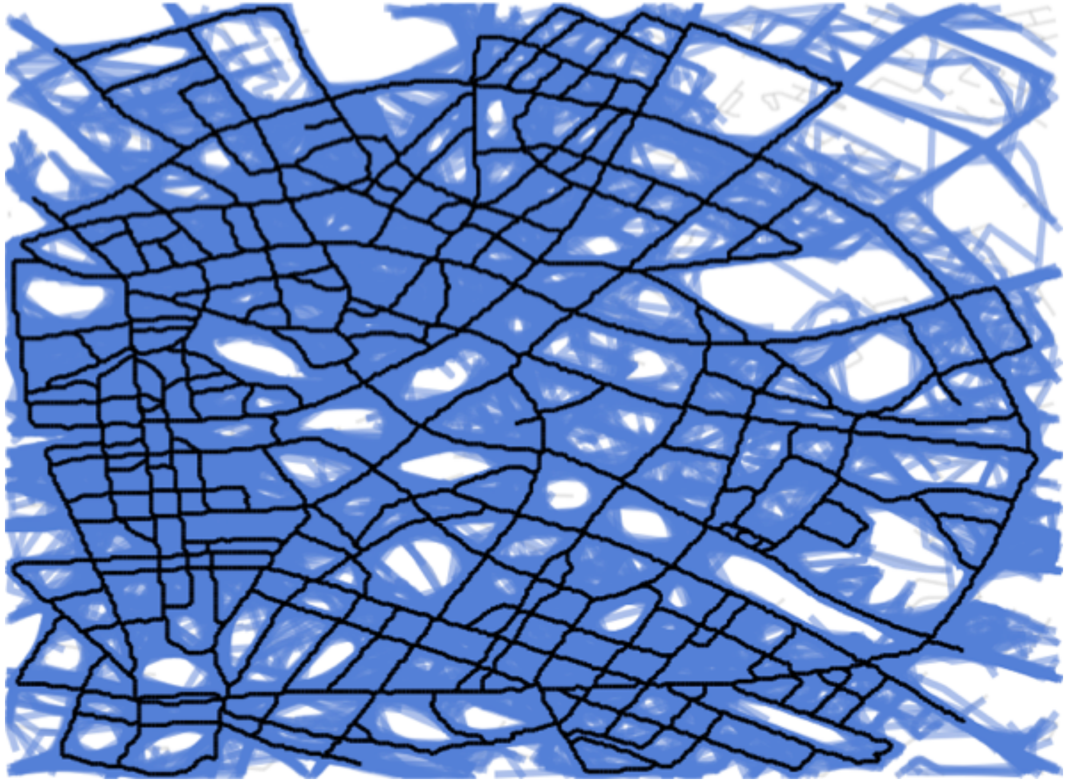




Experiment – Reconstruction 2D



Athens



Berlin

Blue lines: GPS trajectories
Black lines: reconstruction result

[Wang, Li, Wang, SIGSPATIAL 2015]

Experiment – Running Time

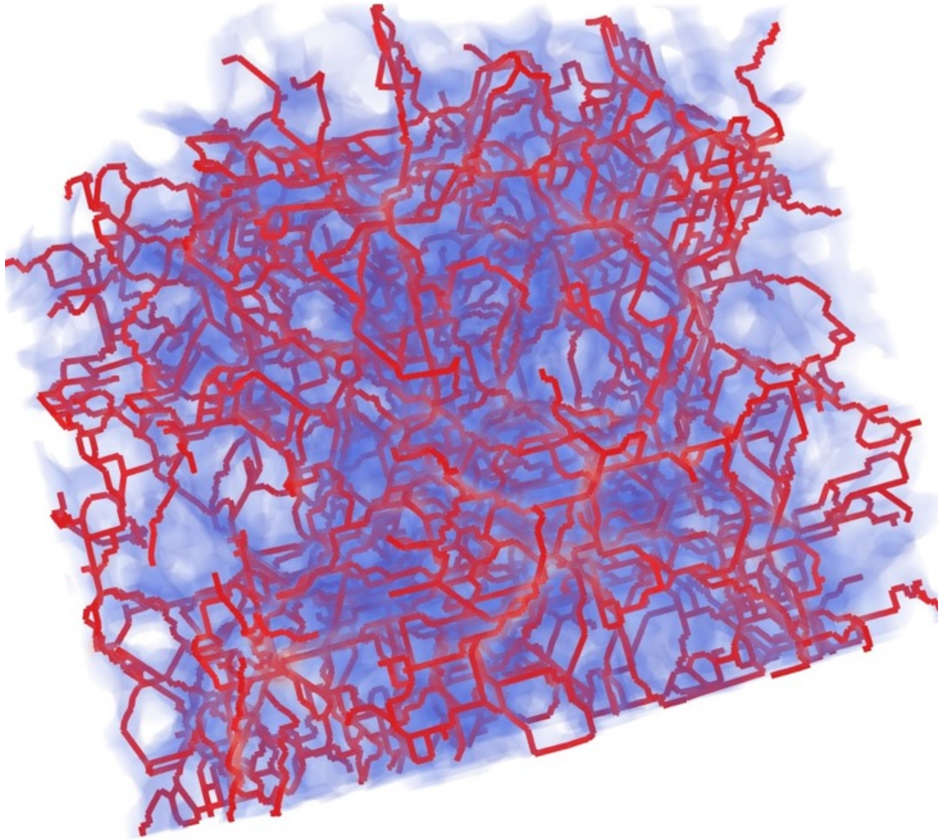
Name	#vertex	#edge	#triangle
Athens	444,600	1,331,111	886,512
Beijing	3,754,580	11,255,893	7,501,314
Berlin	80,741	241,084	160,344
ENZO	262,144	1,536,192	2,524,284
Bone	5,351,976	31,829,419	52,768,394

Computing the persistence pairing is the bottleneck.

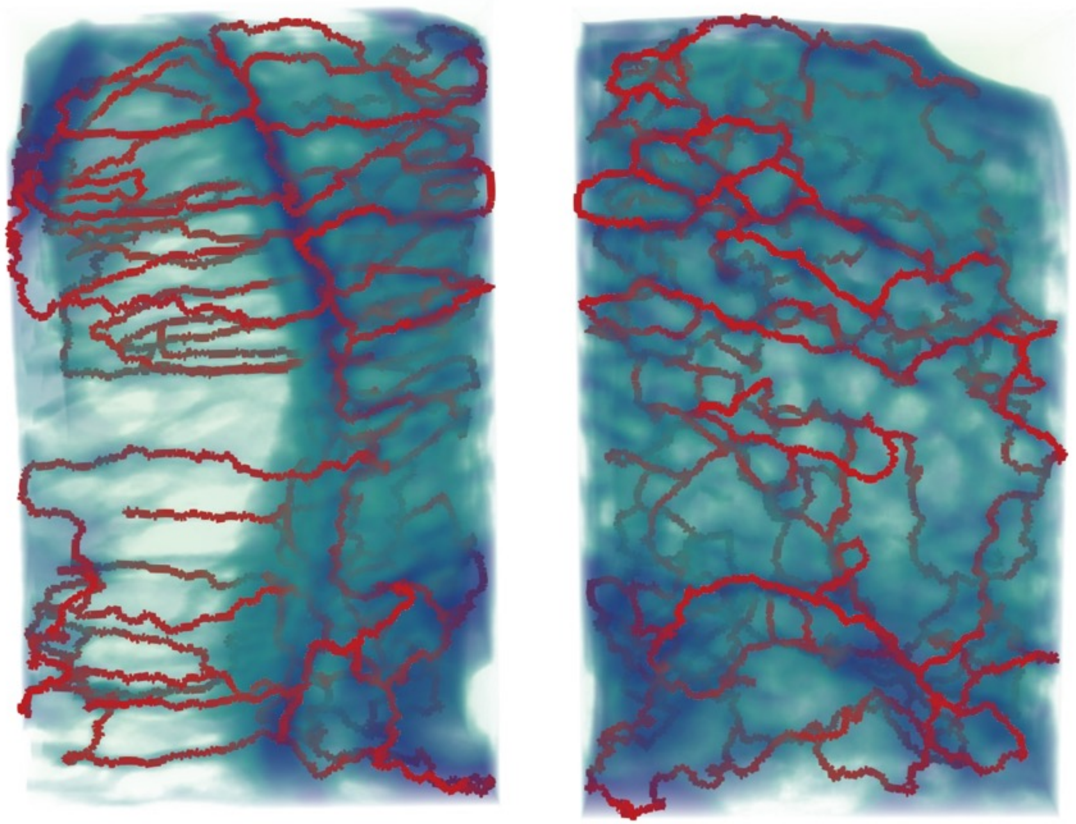
Name	δ	Pre-process	PerSimpVF$^+$ +CollectOutputG	PerSimpTree+Treebased-OutputG
Athens	0.01	12.3	1.2	0.5
Beijing	0.1	97.8	13.1	5.4
Berlin	10	2.0	0.25	0.17

Name	δ	Pre-process	PerSimpVF$^+$ +CollectOutputG	PerSimpTree+Treebased-OutputG
ENZO	50	26.5	1.0	0.38
Bone	40	869	21.6	8.2

Experiment – Reconstruction 3D



3D: cosmological structure



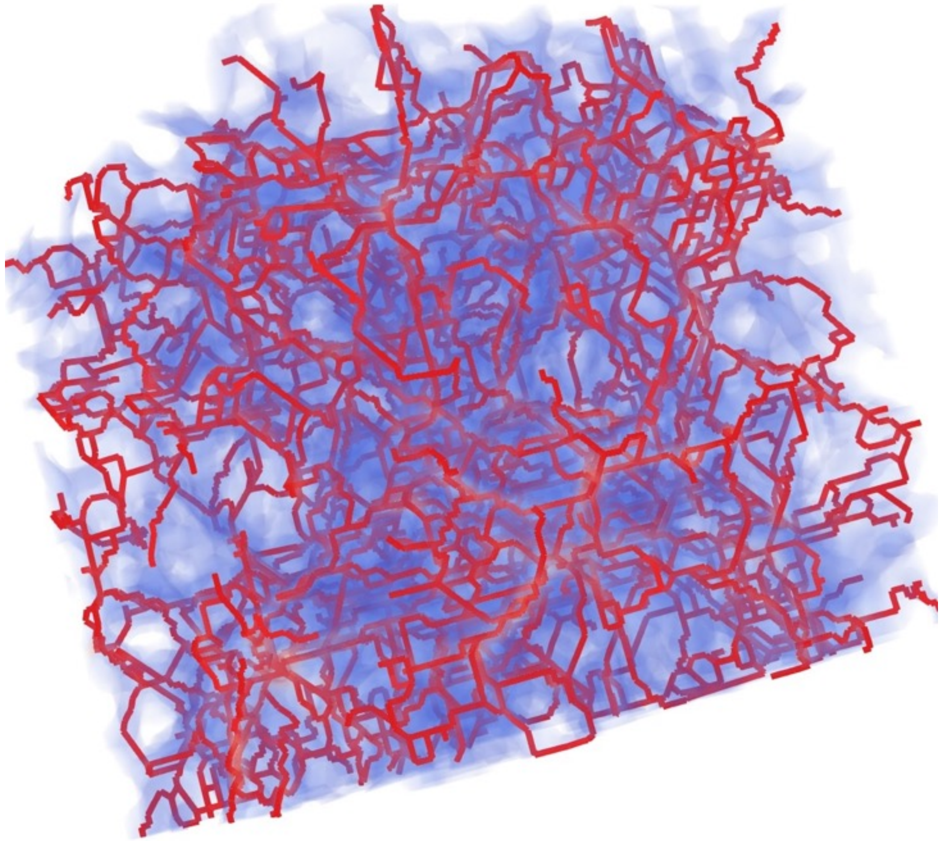
3D: micro CT images of bone

Red lines: reconstruction

Volume: volume rendering of the input density function

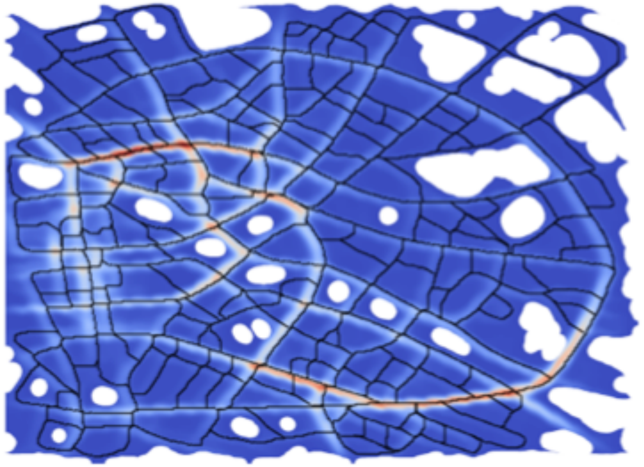
Open Questions

- Make the noise model less restricted
- Reconstruct branches with guarantees
- Reconstruct 2-dimensional spaces
- Further applications

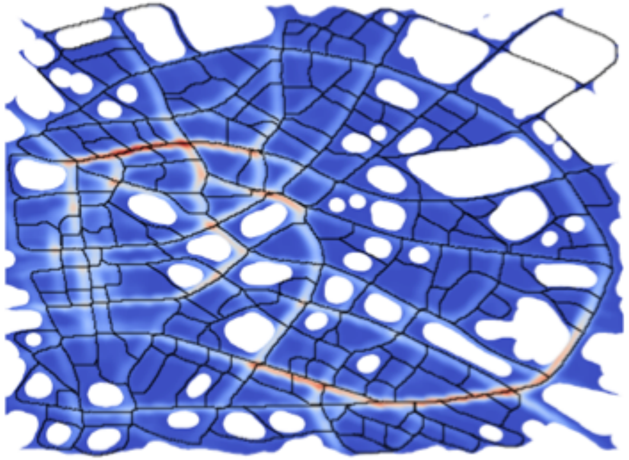


Experiment – Compare with Thresholding

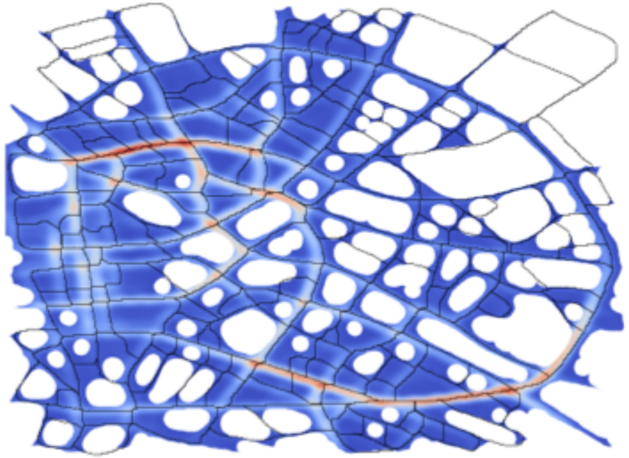
No single threshold value can capture all features due to the non-homogeneousness



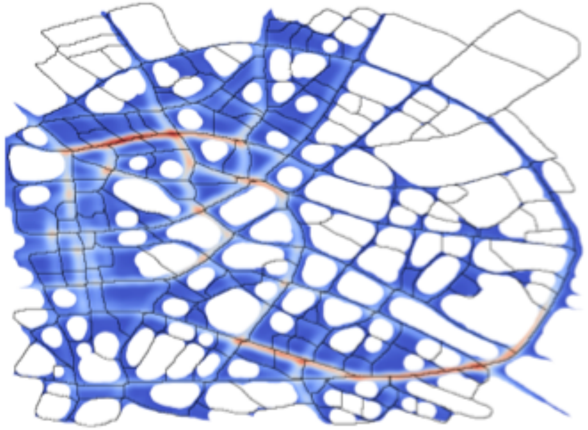
Keep 90% lowest density value



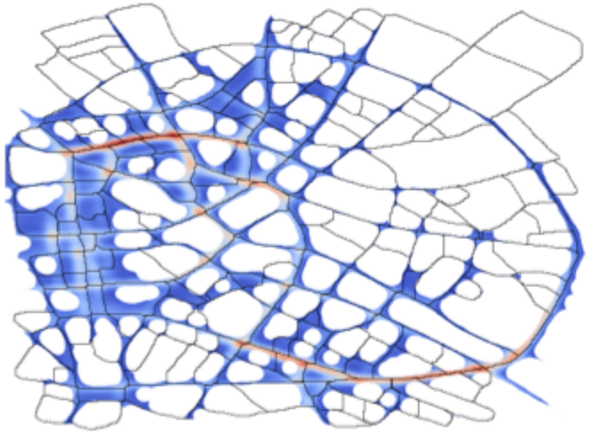
80%



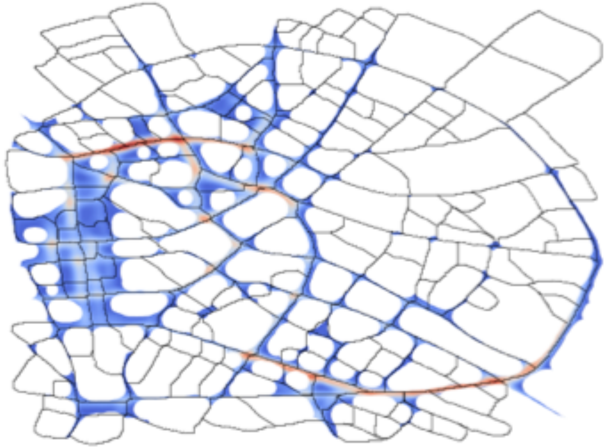
70%



60%



50%



40%