Dynamic Planar Voronoi Diagrams for General Distance Functions

Haim Kaplan
Tel Aviv University
Tel Aviv, Israel

Wolfgang Mulzer
Freie Universität
Berlin, Germany

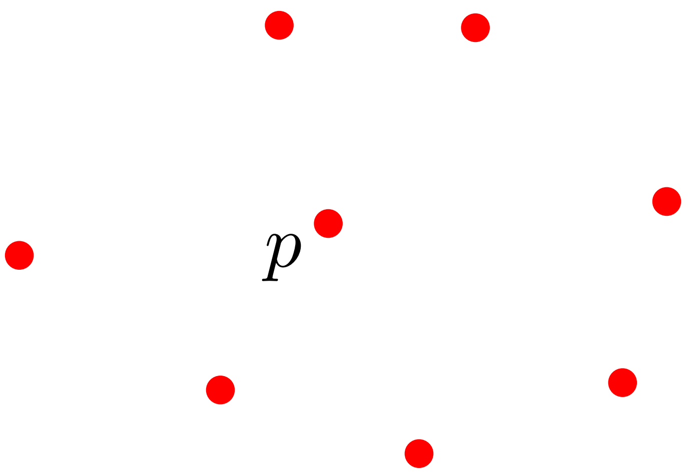
Liam Roditty
Bar-Ilan University
Tel Aviv, Israel

Paul Seiferth
Freie Universität
Berlin, Germany

Micha Sharir
Tel Aviv University
Tel Aviv, Israel

Dynamic Nearest Neighbor Search

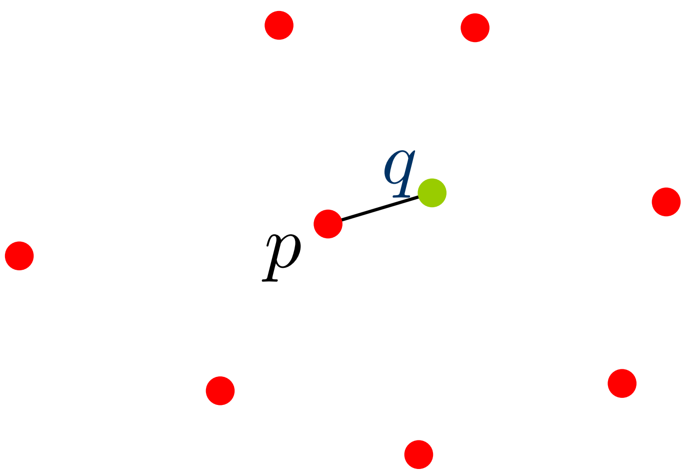
sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \rightarrow \mathbb{R}$ for all $p \in P$



Euclidean

Dynamic Nearest Neighbor Search

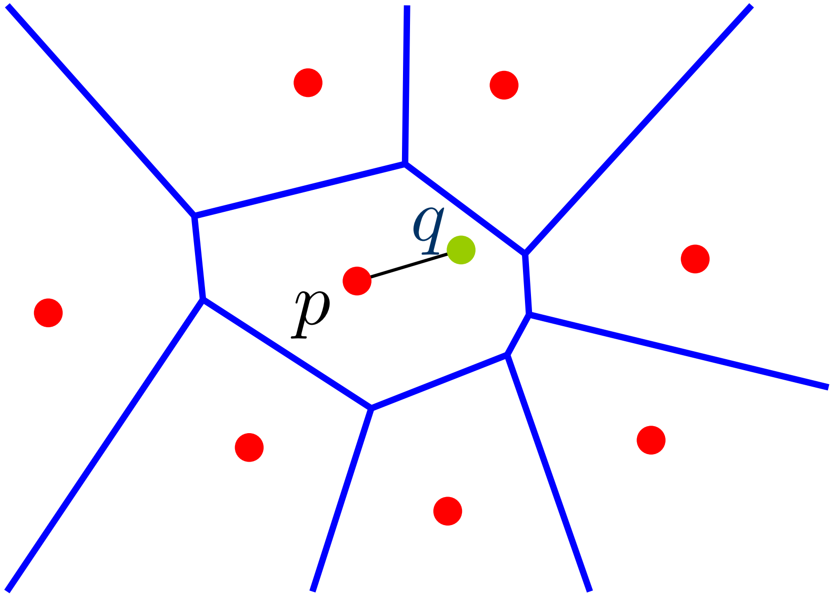
sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ for all $p \in P$



Euclidean

Dynamic Nearest Neighbor Search

sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ for all $p \in P$

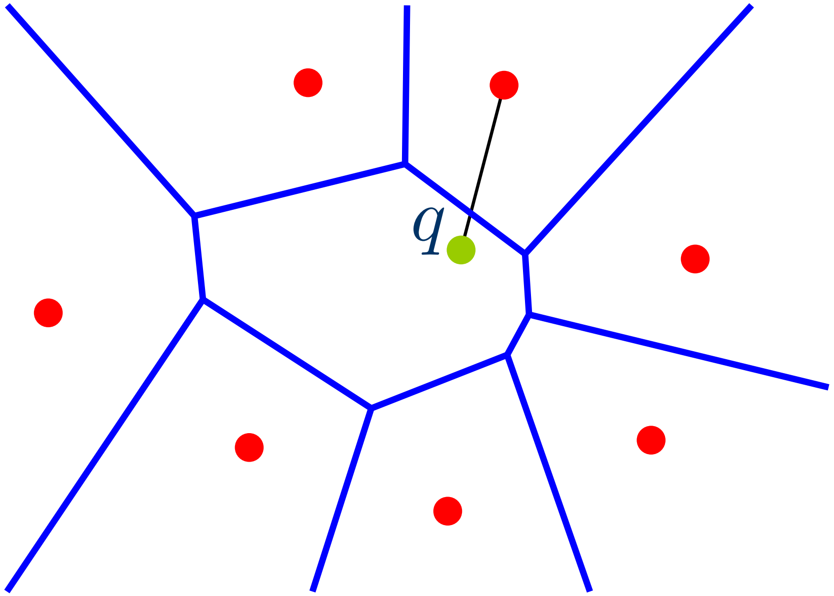


Euclidean

Dynamic Nearest Neighbor Search

sites P in $\mathbb{R}^2$ +

distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ for all $p \in P$

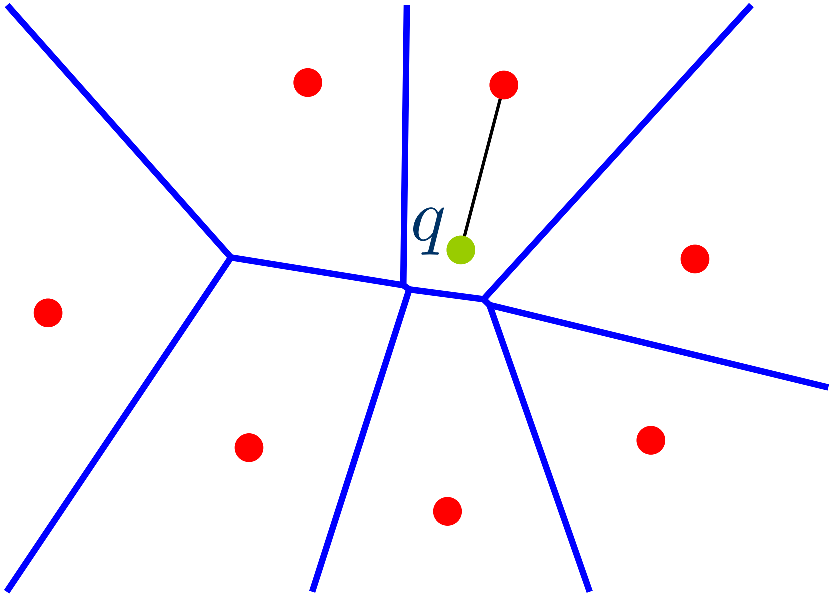


Euclidean

Dynamic Nearest Neighbor Search

sites P in $\mathbb{R}^2$ +

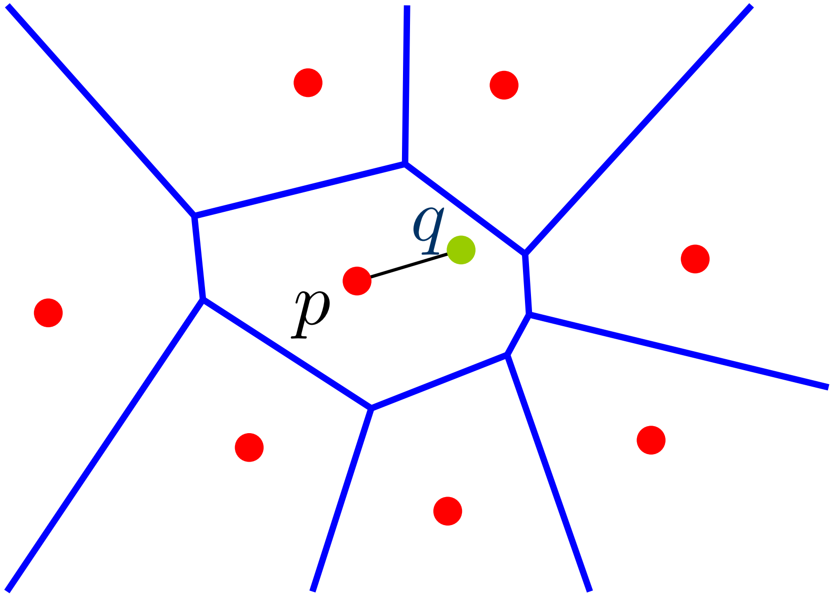
distance functions $\delta_p : \mathbb{R}^2 \rightarrow \mathbb{R}$ for all $p \in P$



Euclidean

Dynamic Nearest Neighbor Search

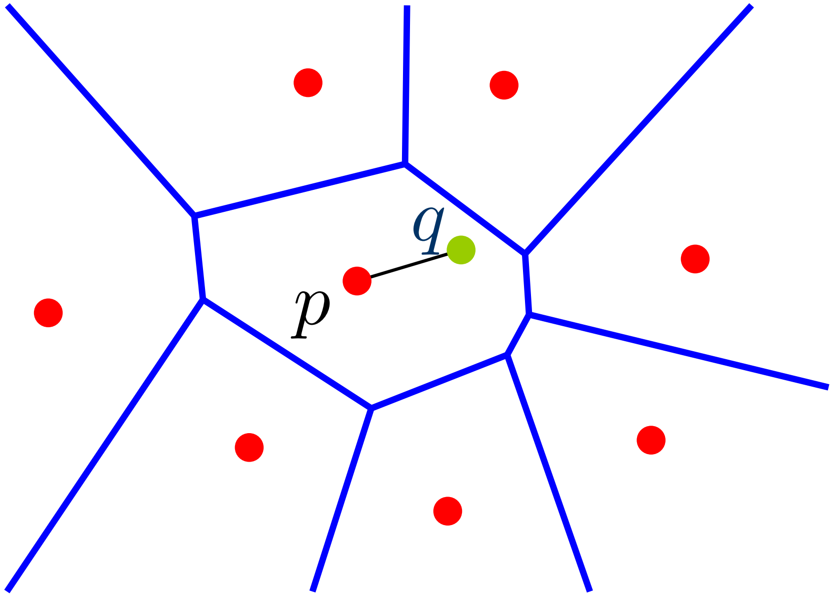
sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ for all $p \in P$



Euclidean

Dynamic Nearest Neighbor Search

sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ for all $p \in P$

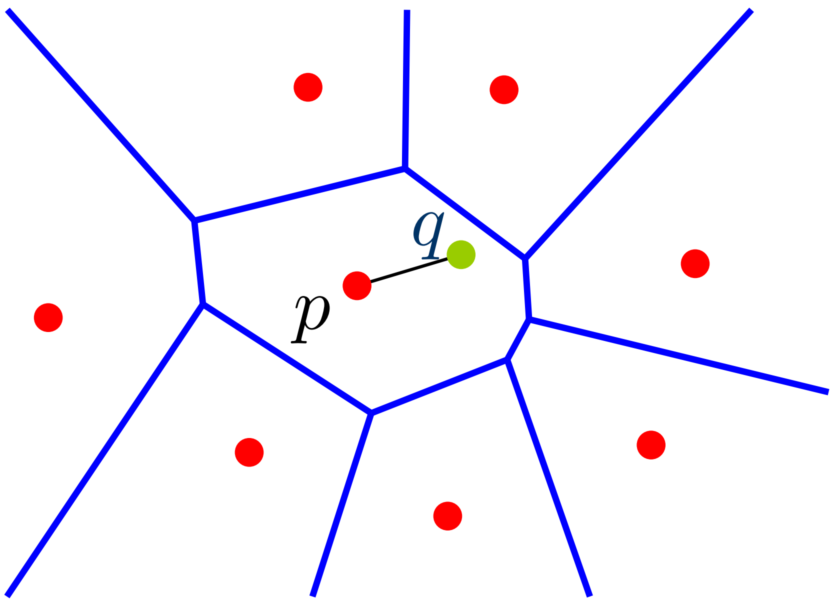


Euclidean

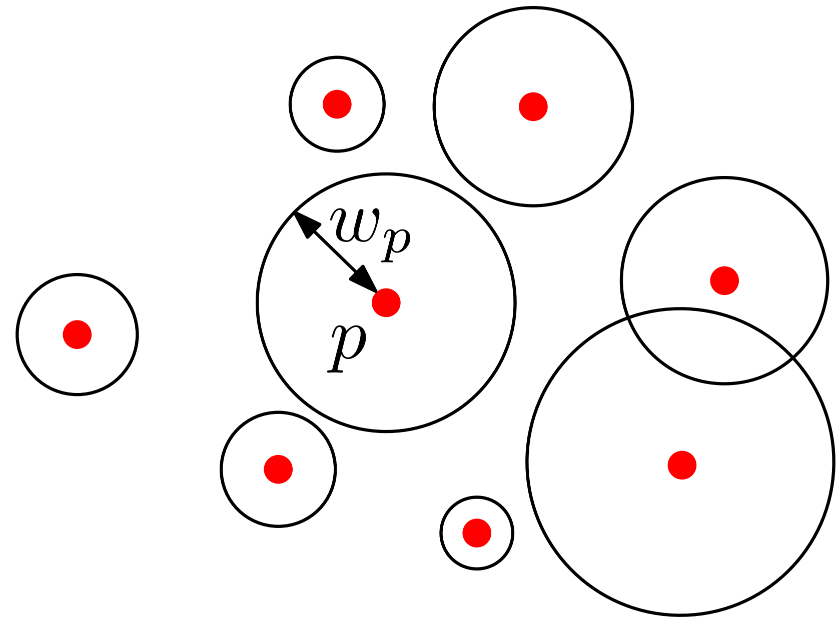
	updates	queries
[AM95]	n^ε	$\log n$
[Cha06]	$\log^6 n$	$\log^2 n$

Dynamic Nearest Neighbor Search

sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ for all $p \in P$



Euclidean

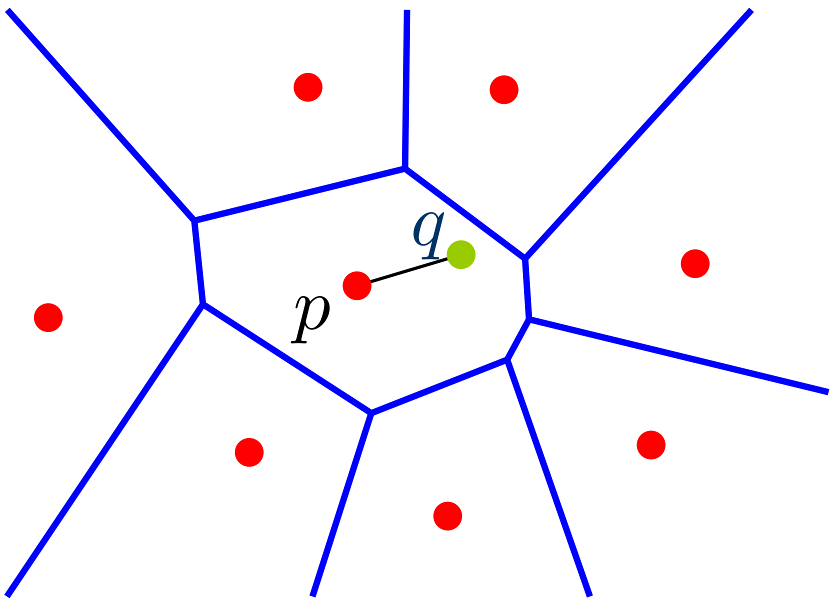


gen. metric
(e.g., add. weighted or any ℓ_p-norm)

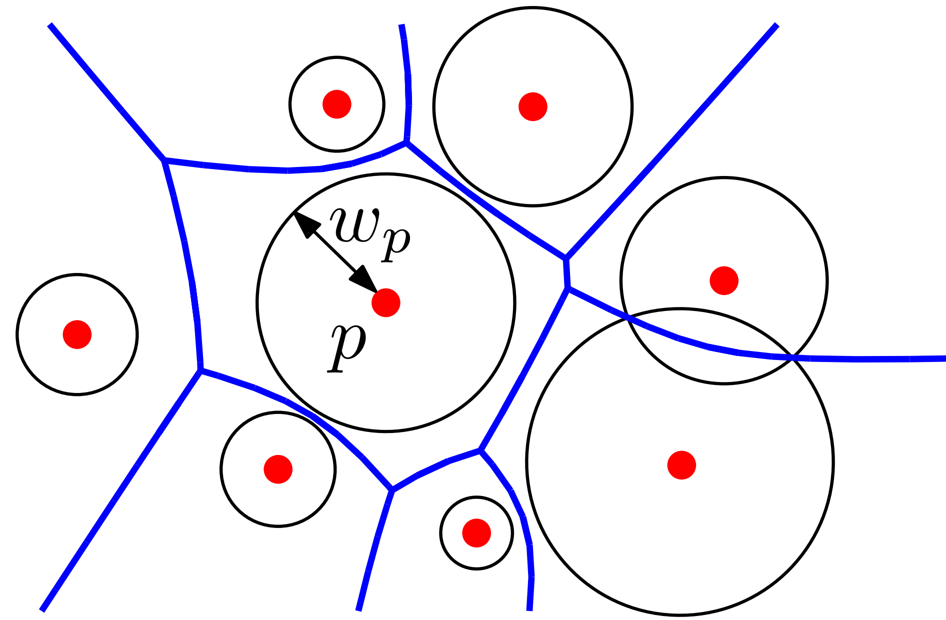
	updates	queries
[AM95]	n^ε	$\log n$
[Cha06]	$\log^6 n$	$\log^2 n$

Dynamic Nearest Neighbor Search

sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ for all $p \in P$



Euclidean

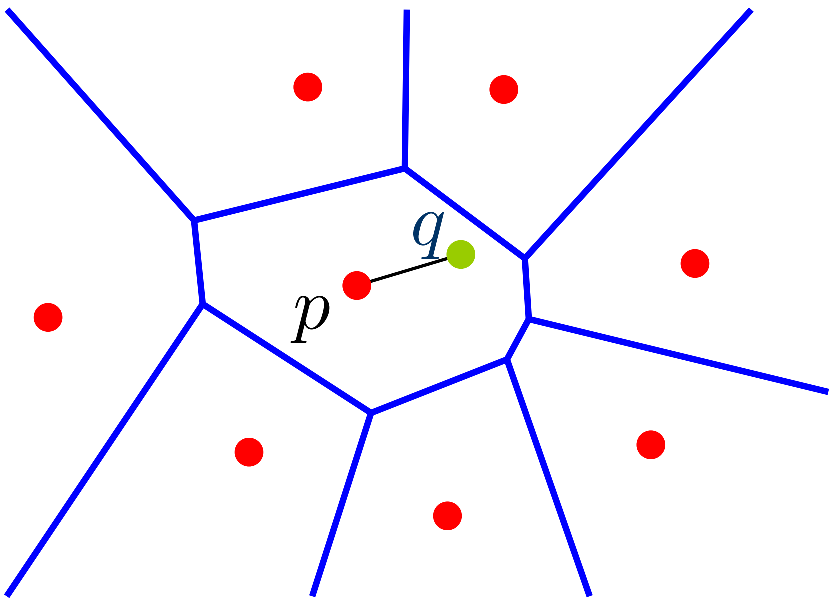


gen. metric
(e.g., add. weighted or any ℓ_p-norm)

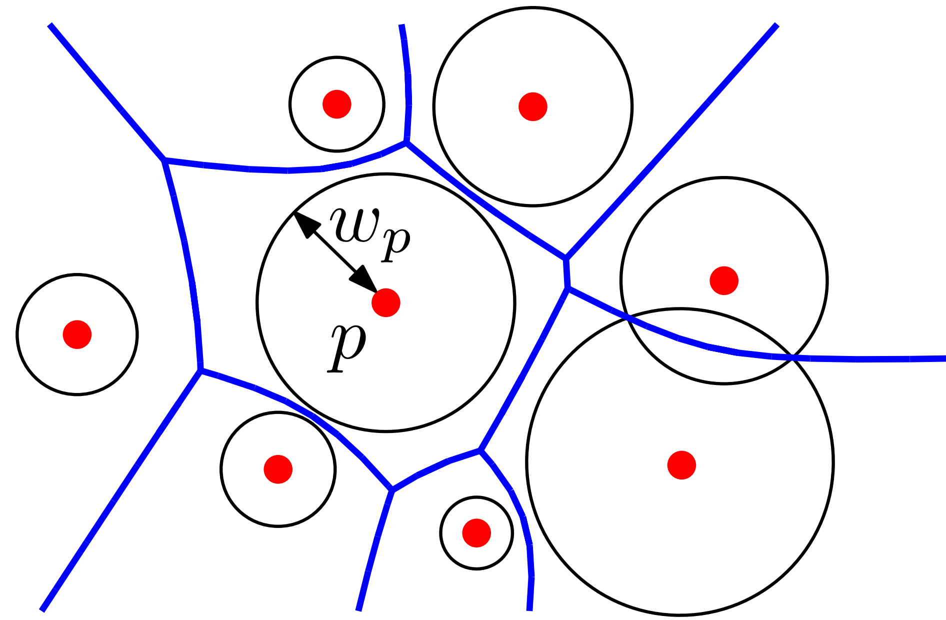
	updates	queries
[AM95]	n^{ε}	$\log n$
[Cha06]	$\log^6 n$	$\log^2 n$

Dynamic Nearest Neighbor Search

sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ for all $p \in P$



Euclidean



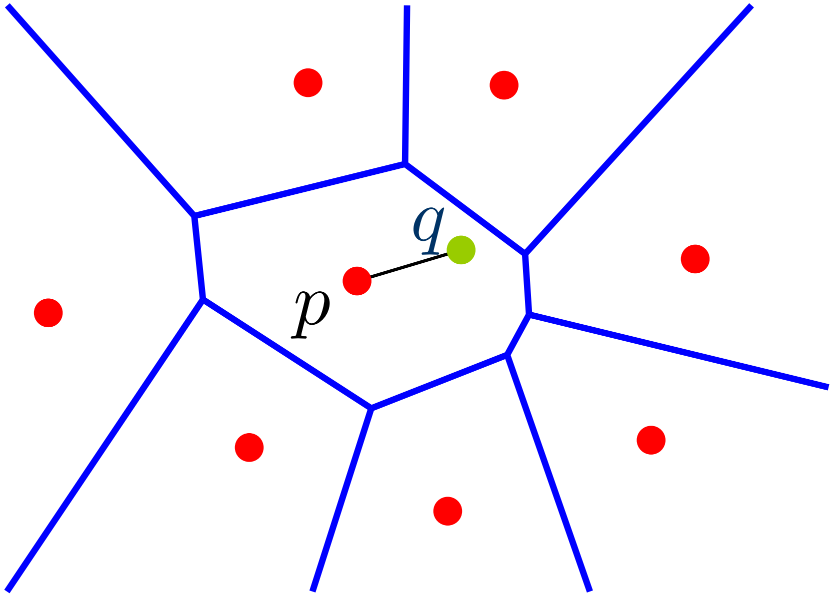
gen. metric
(e.g., add. weighted or any ℓ_p-norm)

	updates	queries
[AM95]	n^ε	$\log n$
[Cha06]	$\log^6 n$	$\log^2 n$

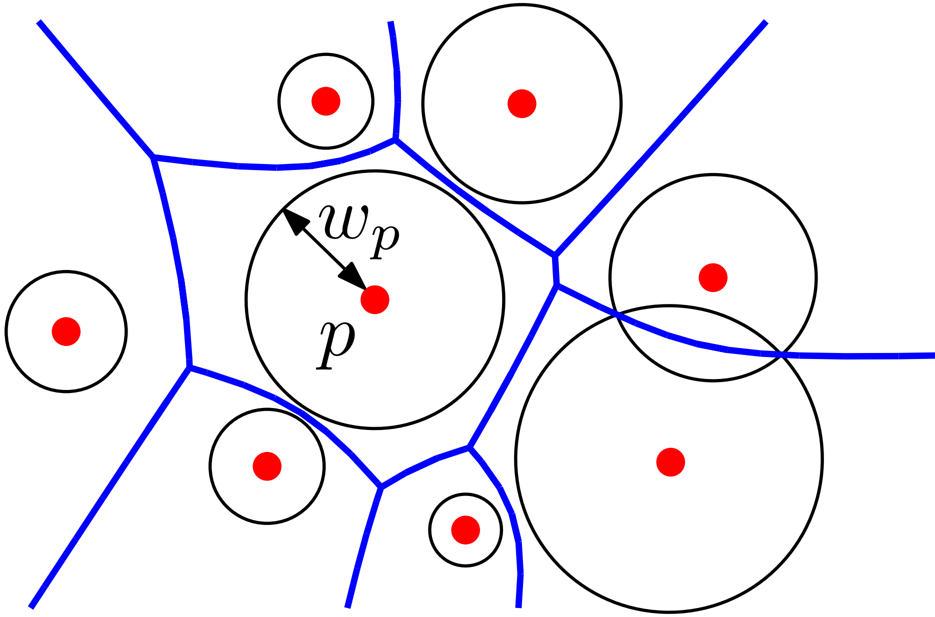
	updates	queries
[AES99]	n^ε	$\log n$

Dynamic Nearest Neighbor Search

sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ for all $p \in P$



Euclidean



gen. metric
(e.g., add. weighted or any ℓ_p-norm)

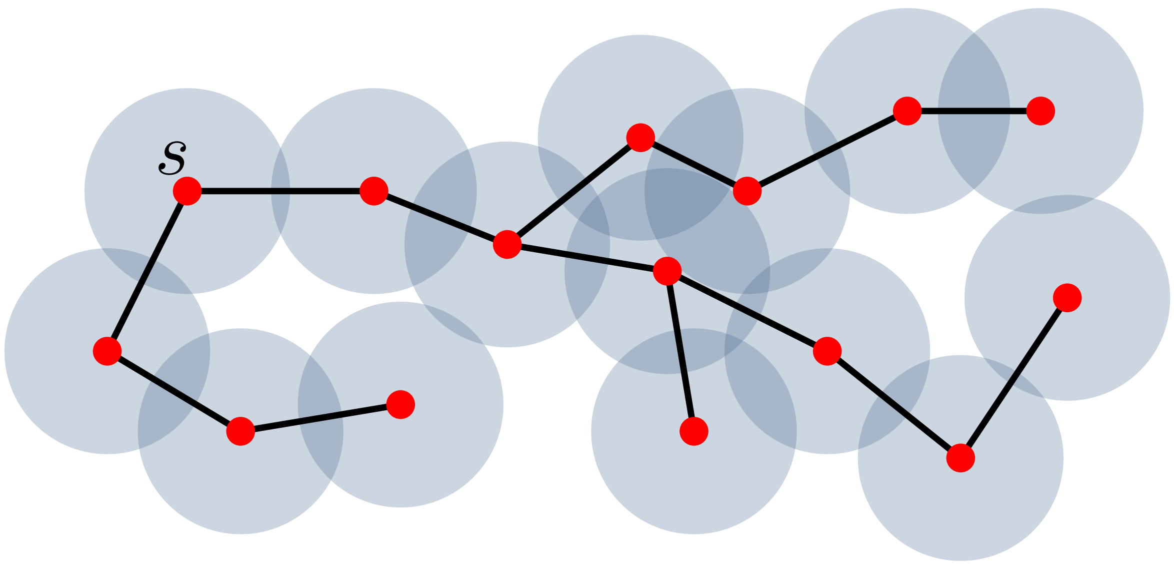
	updates	queries
[AM95]	n^ε	$\log n$
[Cha06]	$\log^6 n$	$\log^2 n$

	updates	queries
[AES99]	n^ε	$\log n$
Now:	polylog(n)	$\log^2 n$

Applications

Single source shortest path in unit disk graphs

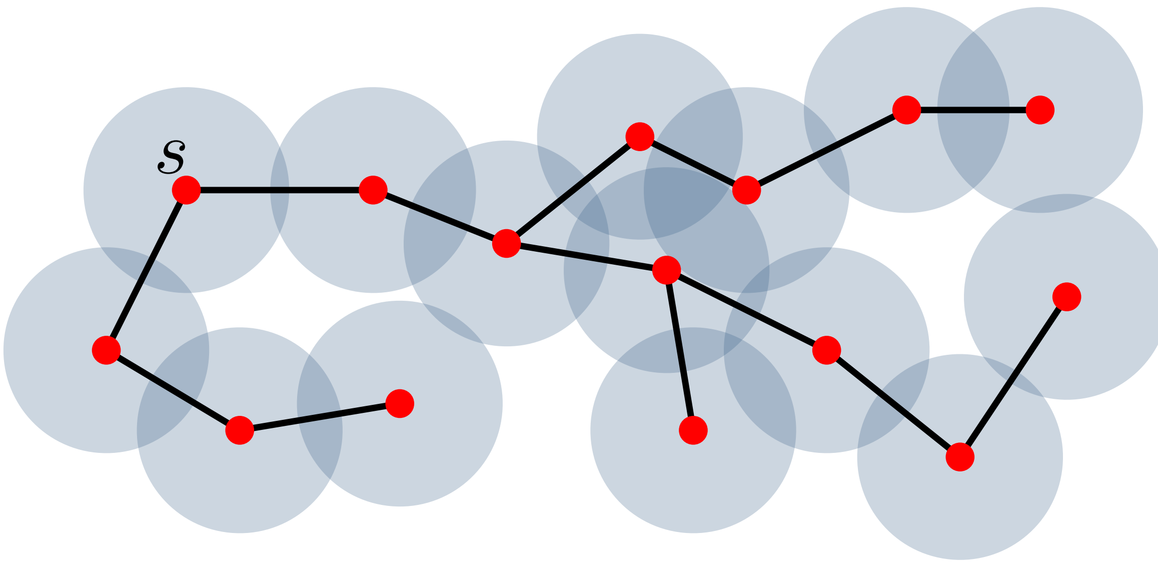
Old Bound: $n^{1+\varepsilon}$ [CJ15]   New Bound: $n\text{polylog}(n)$ exp.



Applications

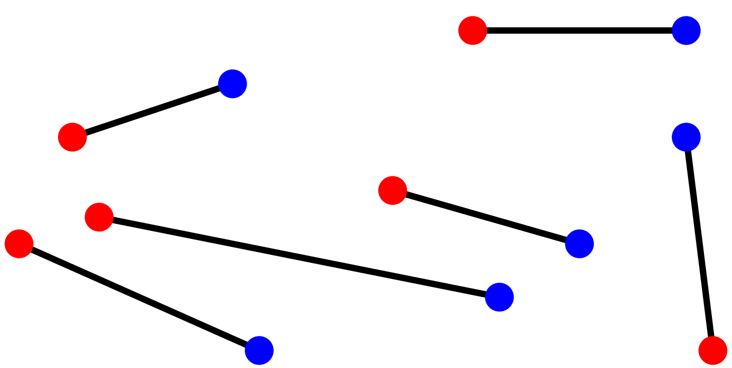
Single source shortest path in unit disk graphs

Old Bound: $n^{1+\varepsilon}$ [CJ15] New Bound: $n\text{polylog}(n)$ exp.



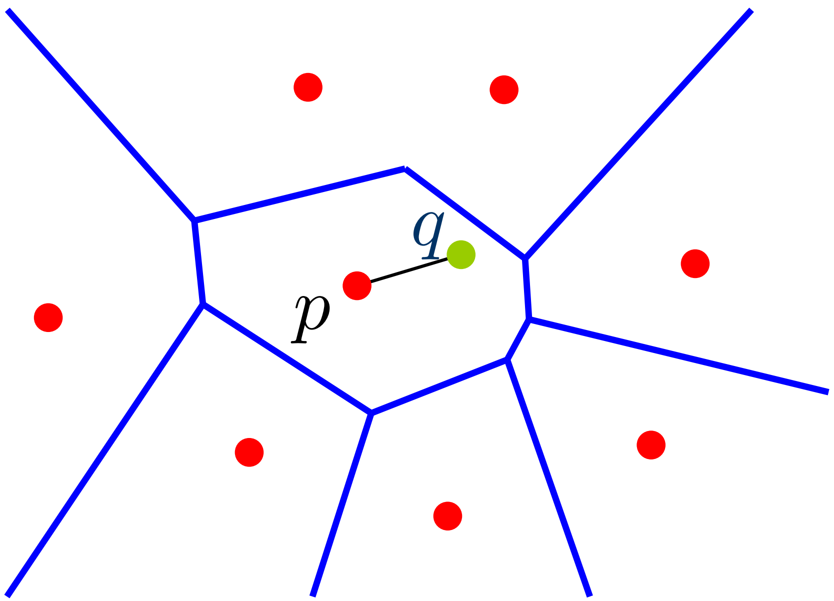
Minimum Euclidean planar bichromatic matching

Old Bound: $n^{2+\varepsilon}$ [AES99] New Bound: $n^2\text{polylog}(n)$ exp.

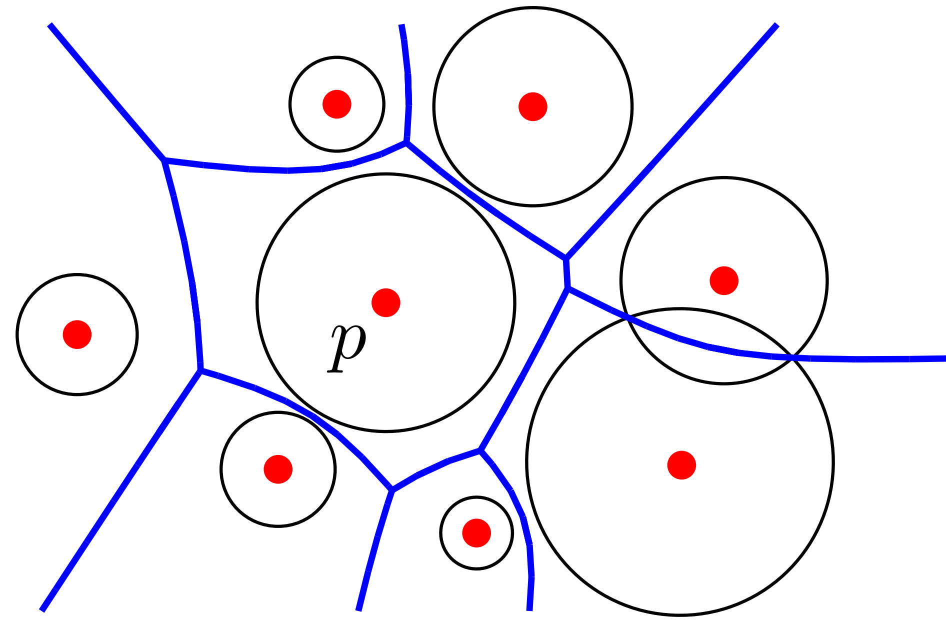


Dynamic Nearest Neighbor Search

sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \rightarrow \mathbb{R}$



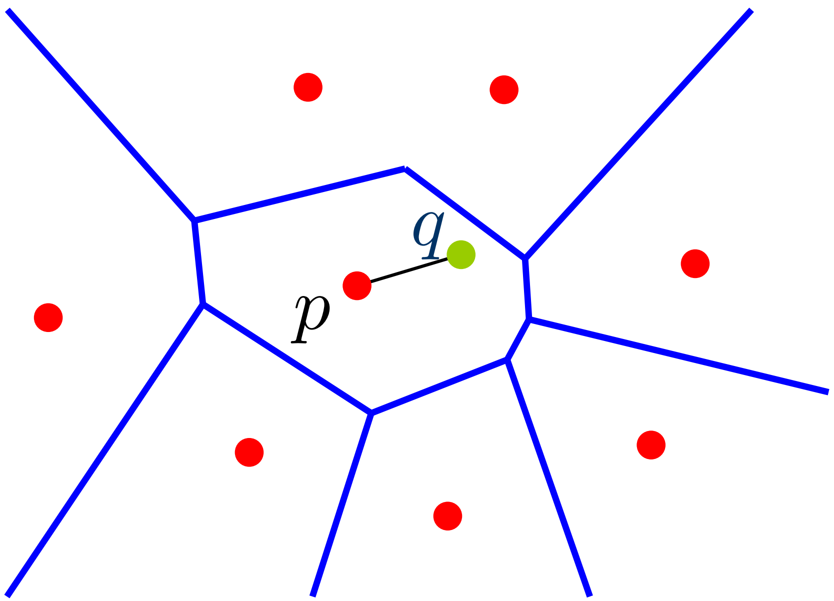
Euclidean



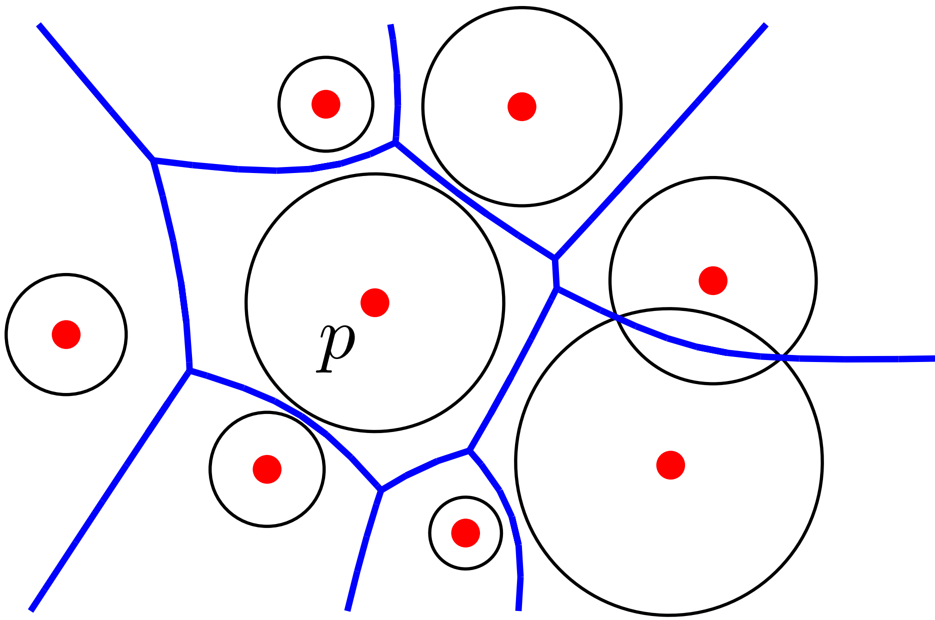
gen. metric

Dynamic Nearest Neighbor Search

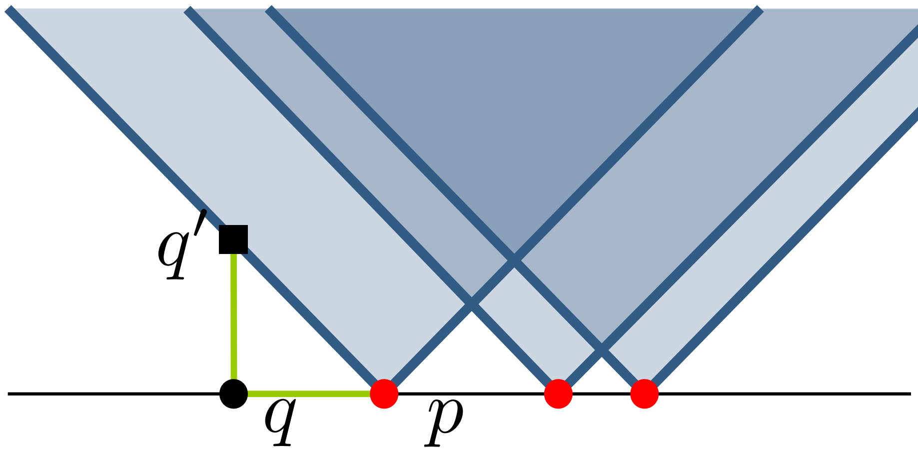
sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$



Euclidean

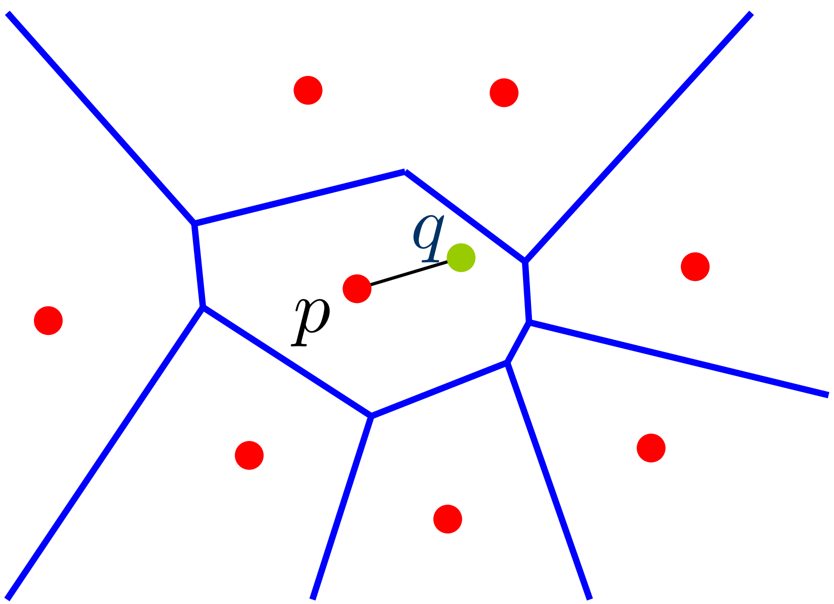


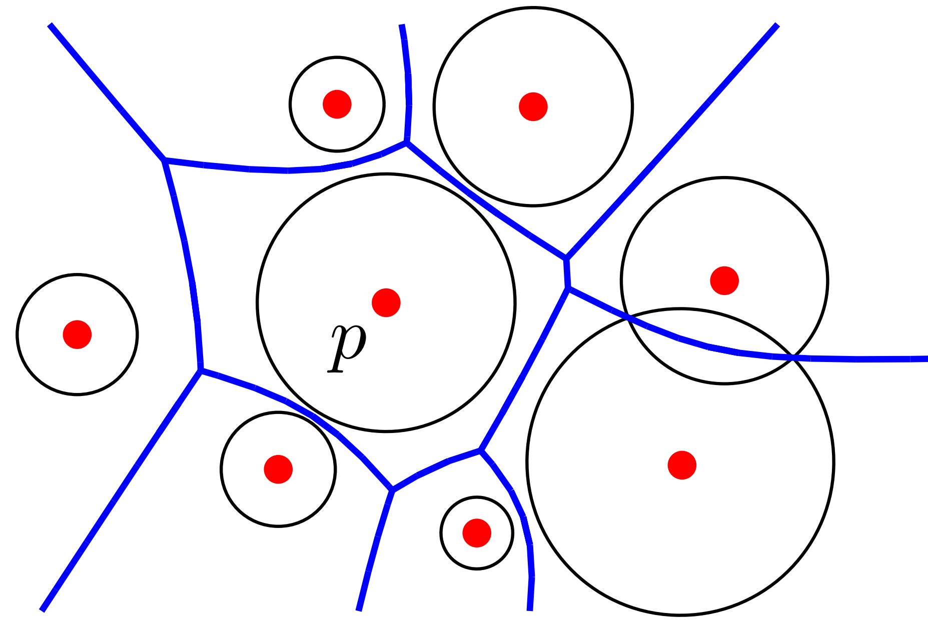
gen. metric



Dynamic Nearest Neighbor Search

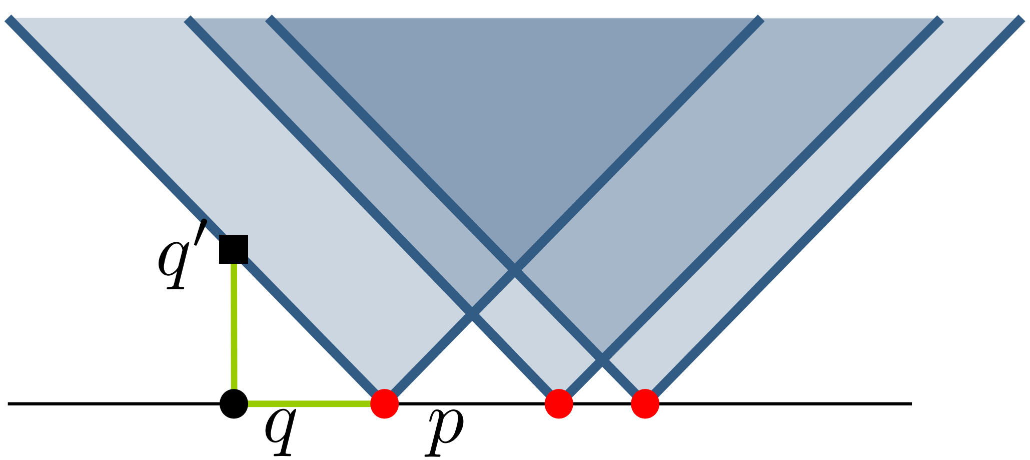
sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$

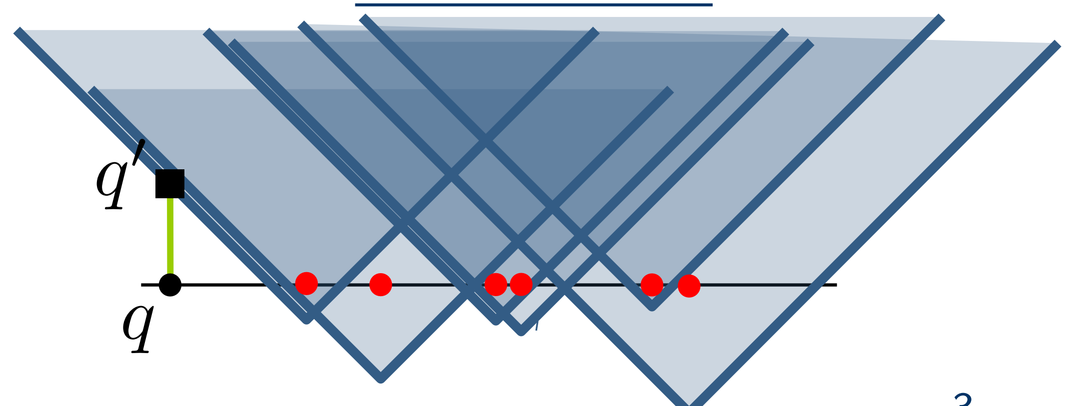




Euclidean

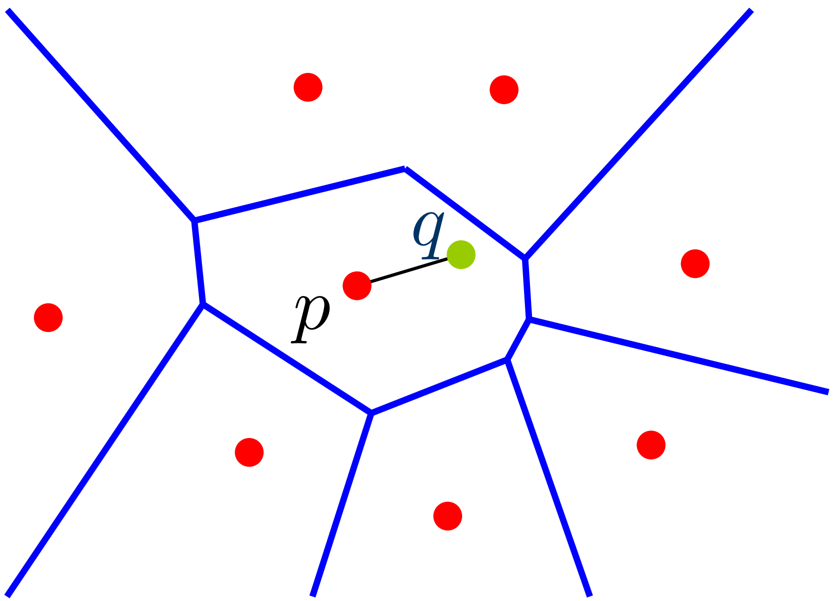
gen. metric

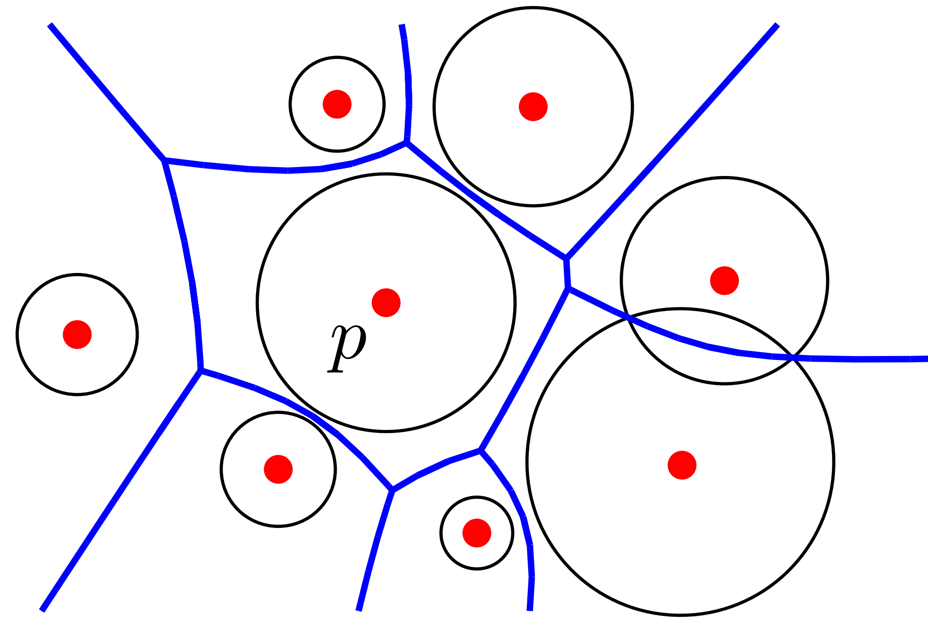




Dynamic Nearest Neighbor Search

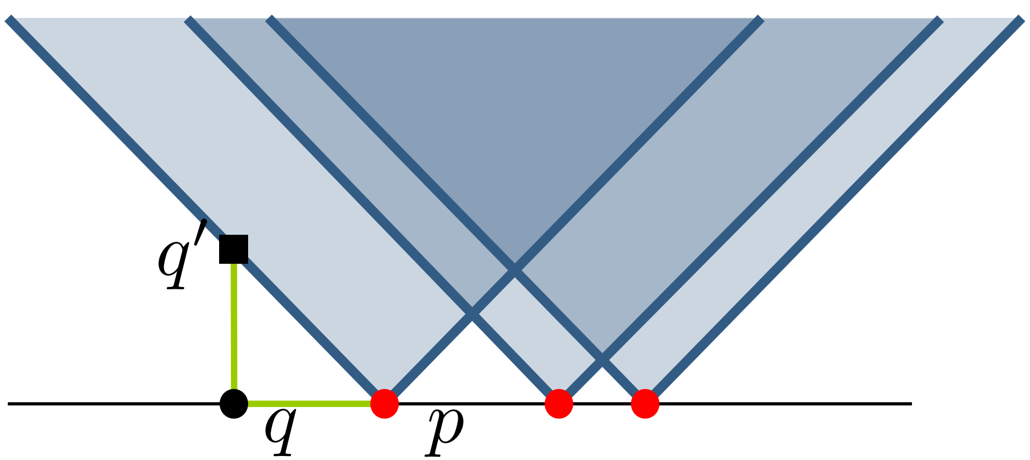
sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ } surfaces S in $\mathbb{R}^3$

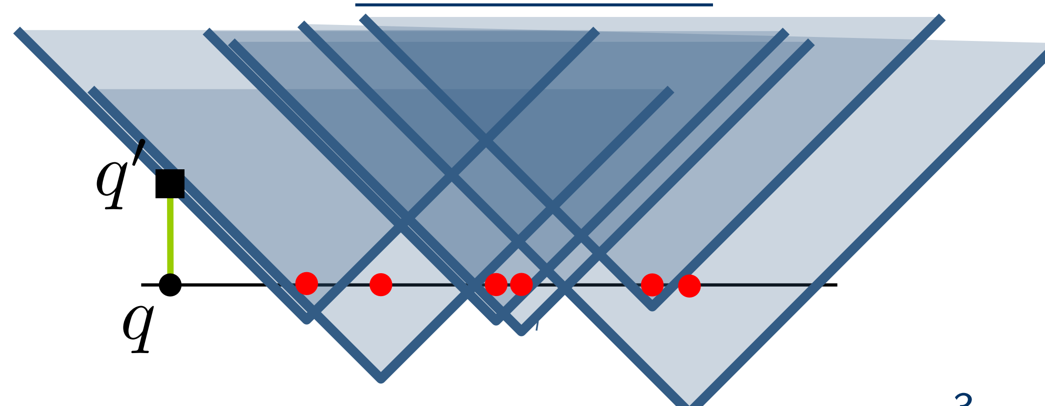




Euclidean

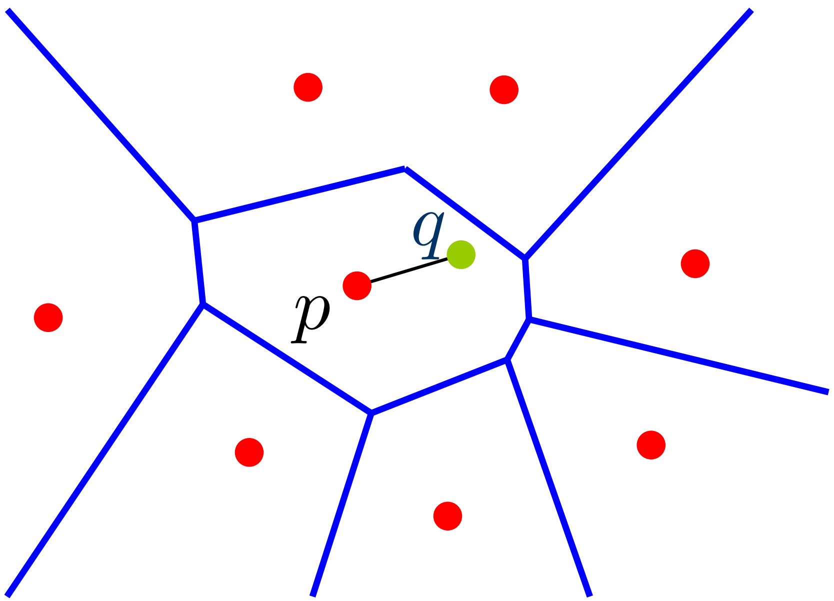
gen. metric



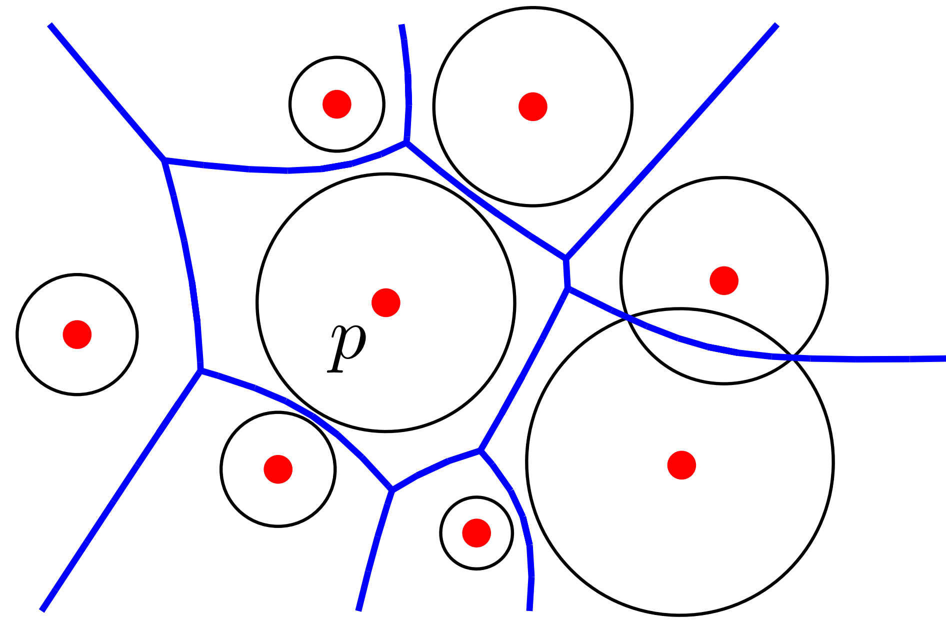


Dynamic Nearest Neighbor Search

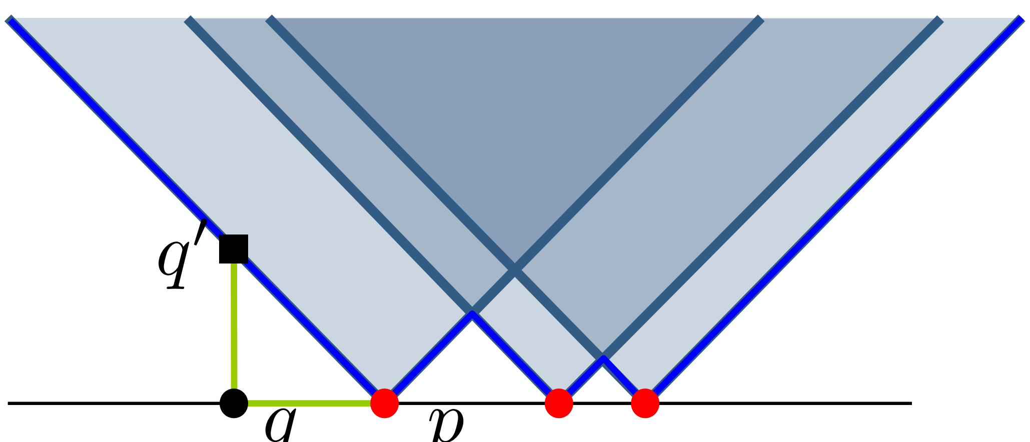
sites P in $\mathbb{R}^2$ +
distance functions $\delta_p : \mathbb{R}^2 \to \mathbb{R}$ } surfaces S in $\mathbb{R}^3$

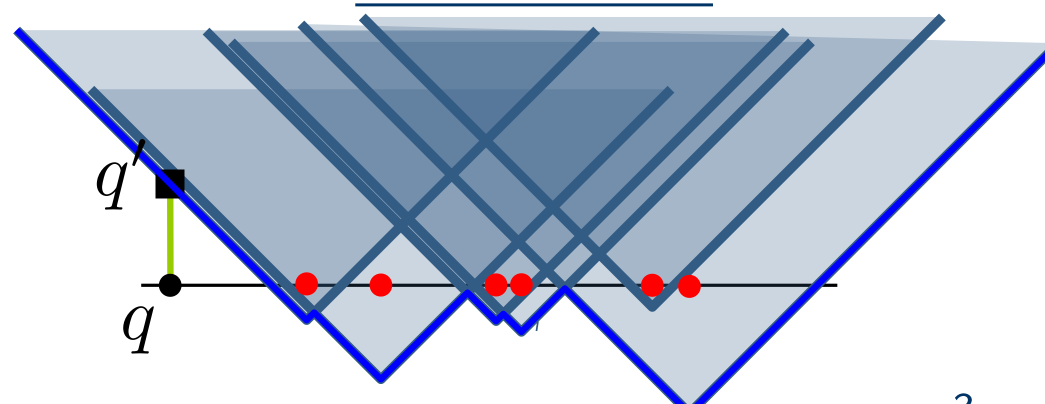


Euclidean

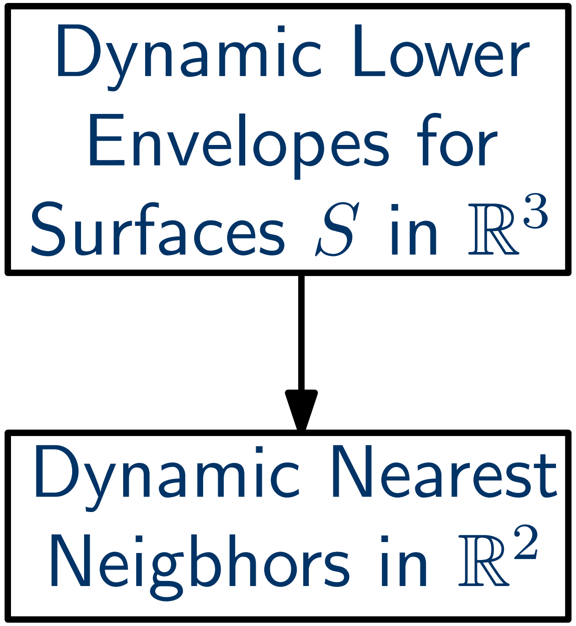


gen. metric

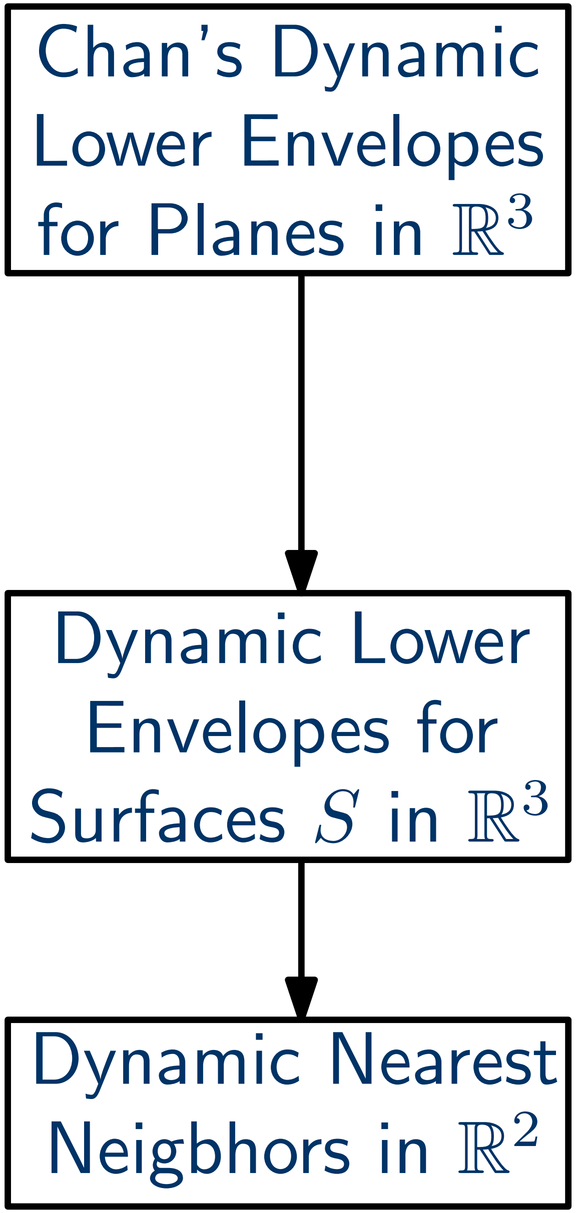




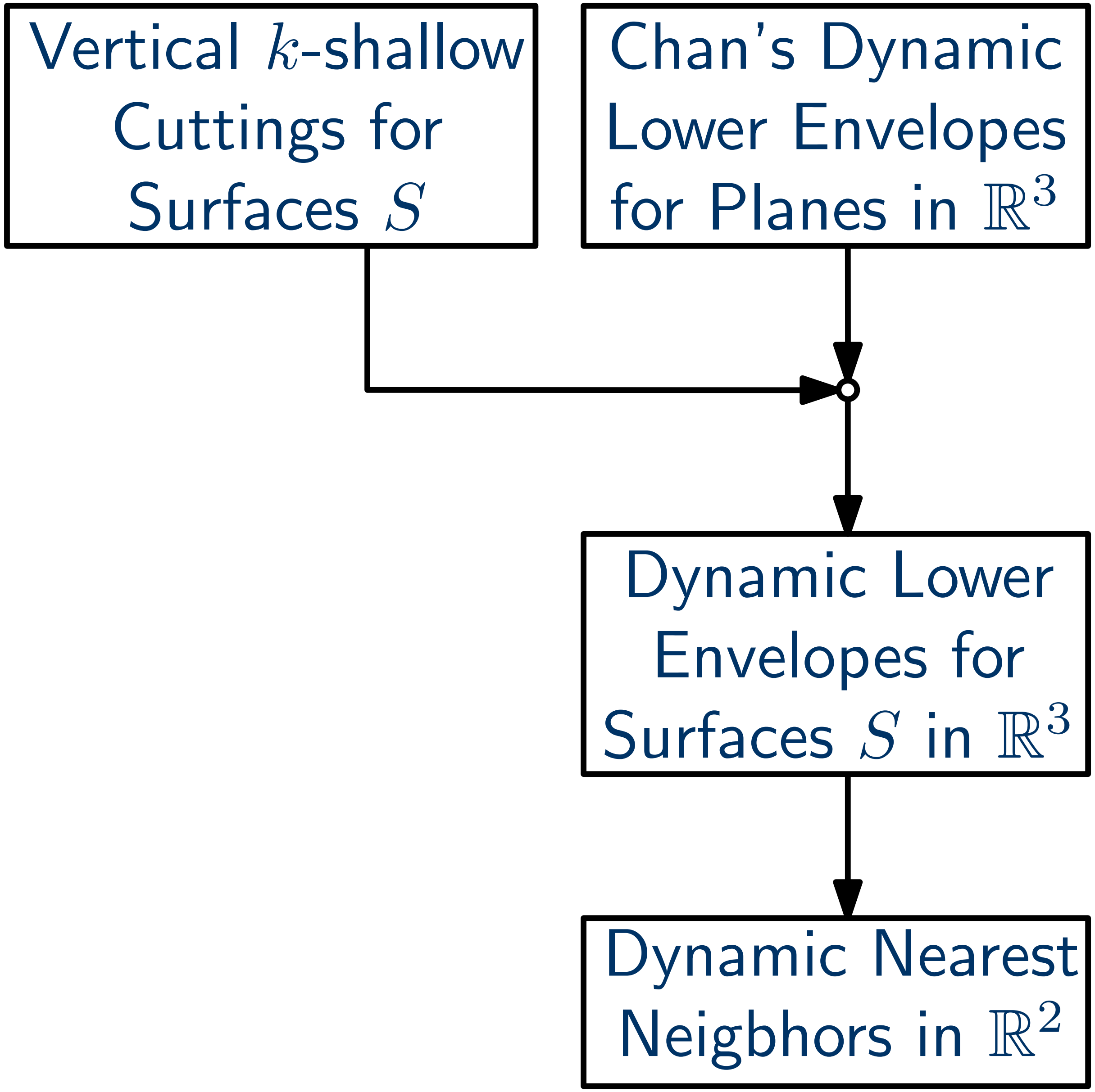
Overview


Dynamic Lower Envelopes for Surfaces S in $\mathbb{R}^3$
Dynamic Nearest Neighbors in $\mathbb{R}^2$

Overview


Chan's Dynamic Lower Envelopes for Planes in $\mathbb{R}^3$
Dynamic Lower Envelopes for Surfaces S in $\mathbb{R}^3$
Dynamic Nearest Neighbors in $\mathbb{R}^2$

Overview



Vertical k-shallow Cuttings for Surfaces S
Chan's Dynamic Lower Envelopes for Planes in $\mathbb{R}^3$
Dynamic Lower Envelopes for Surfaces S in $\mathbb{R}^3$
Dynamic Nearest Neigbhors in $\mathbb{R}^2$

Overview

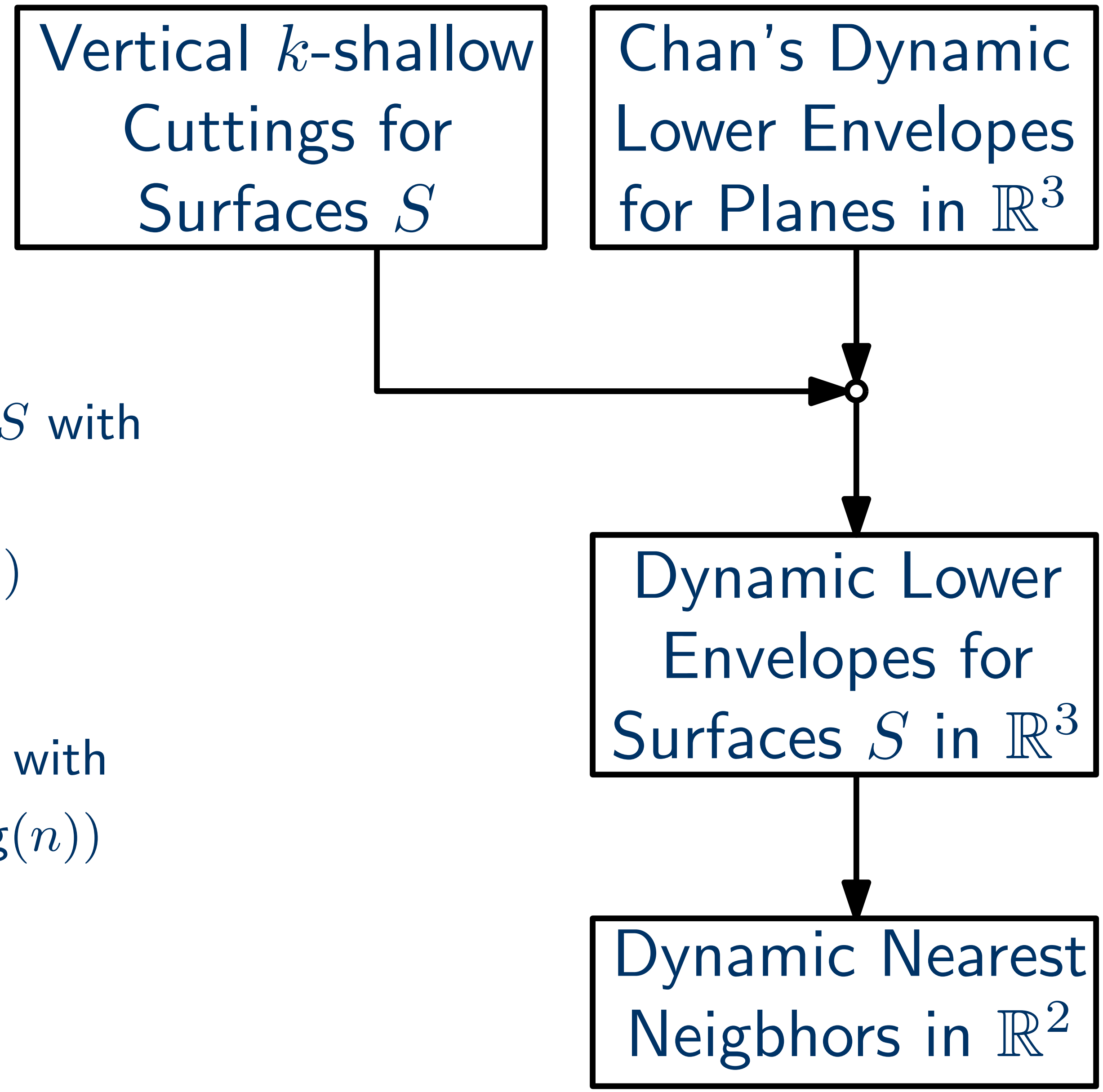
vertical k-shallow cuttings for S with

- size $O((n/k)\text{polylog}n)$
- in time $O(n\text{polylog}(n))$

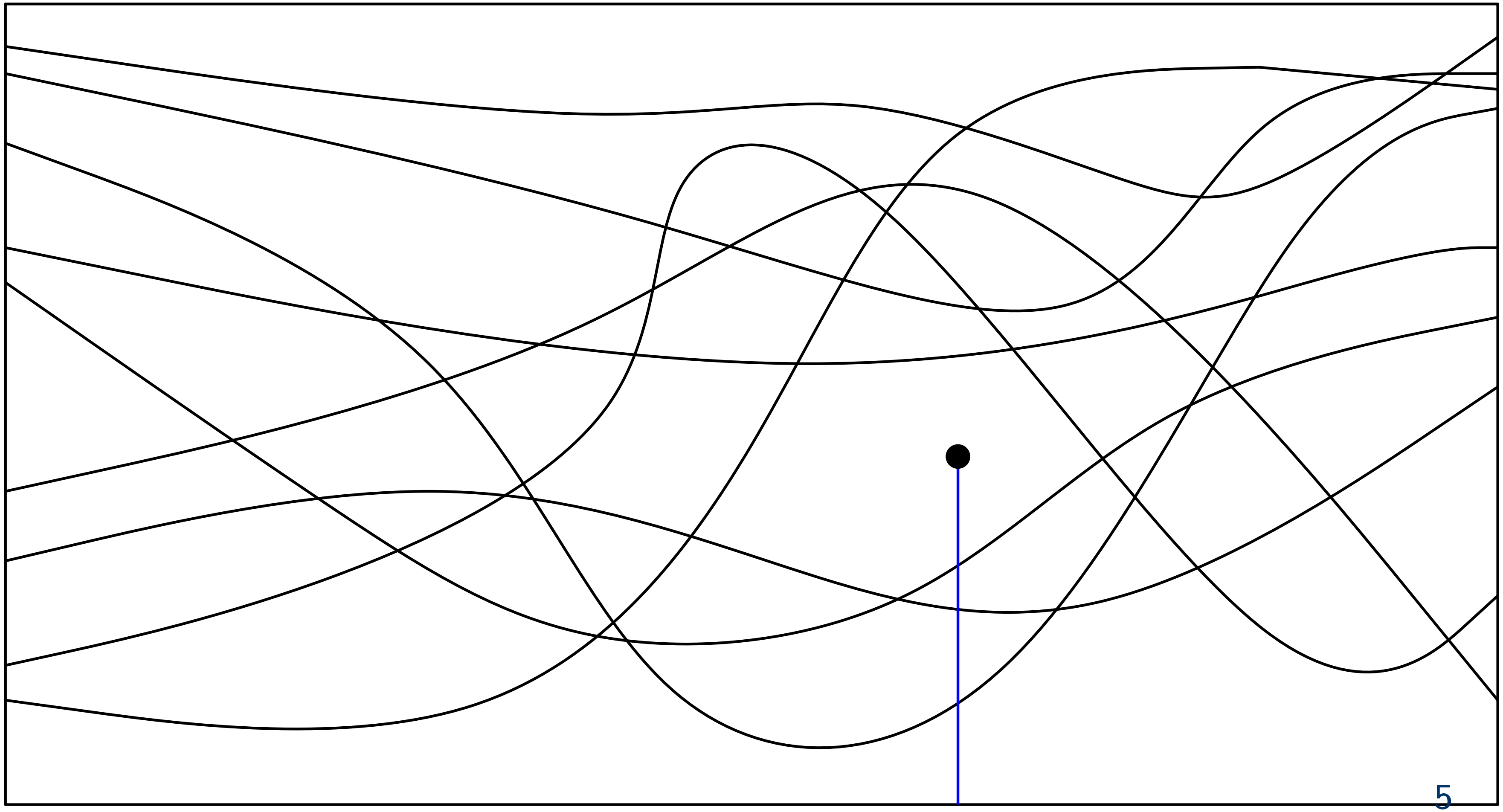
$\Downarrow$

dynamic lower envelopes for S with

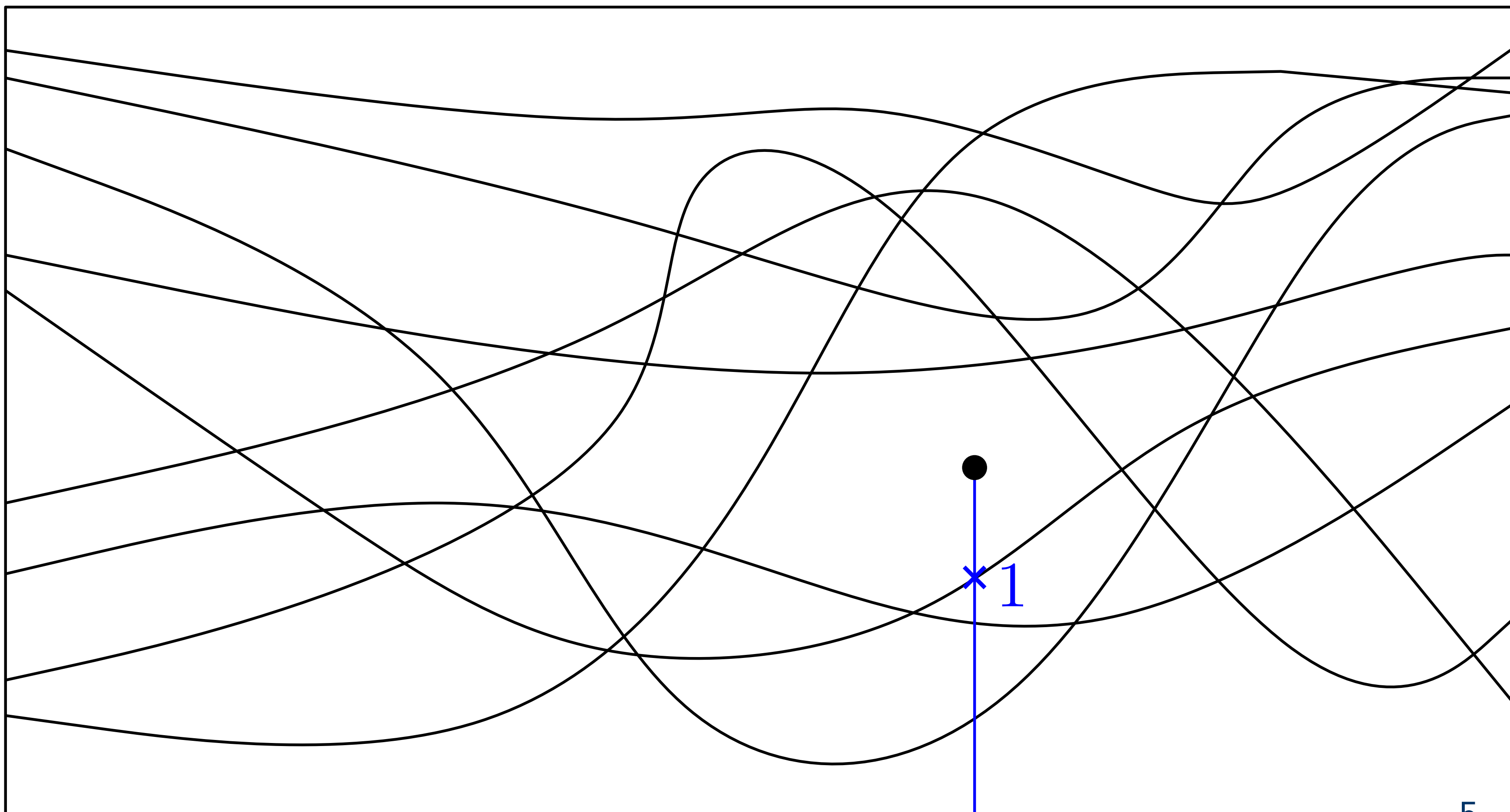
- update time $O(\text{polylog}(n))$
- query time $O(\log^2 n)$



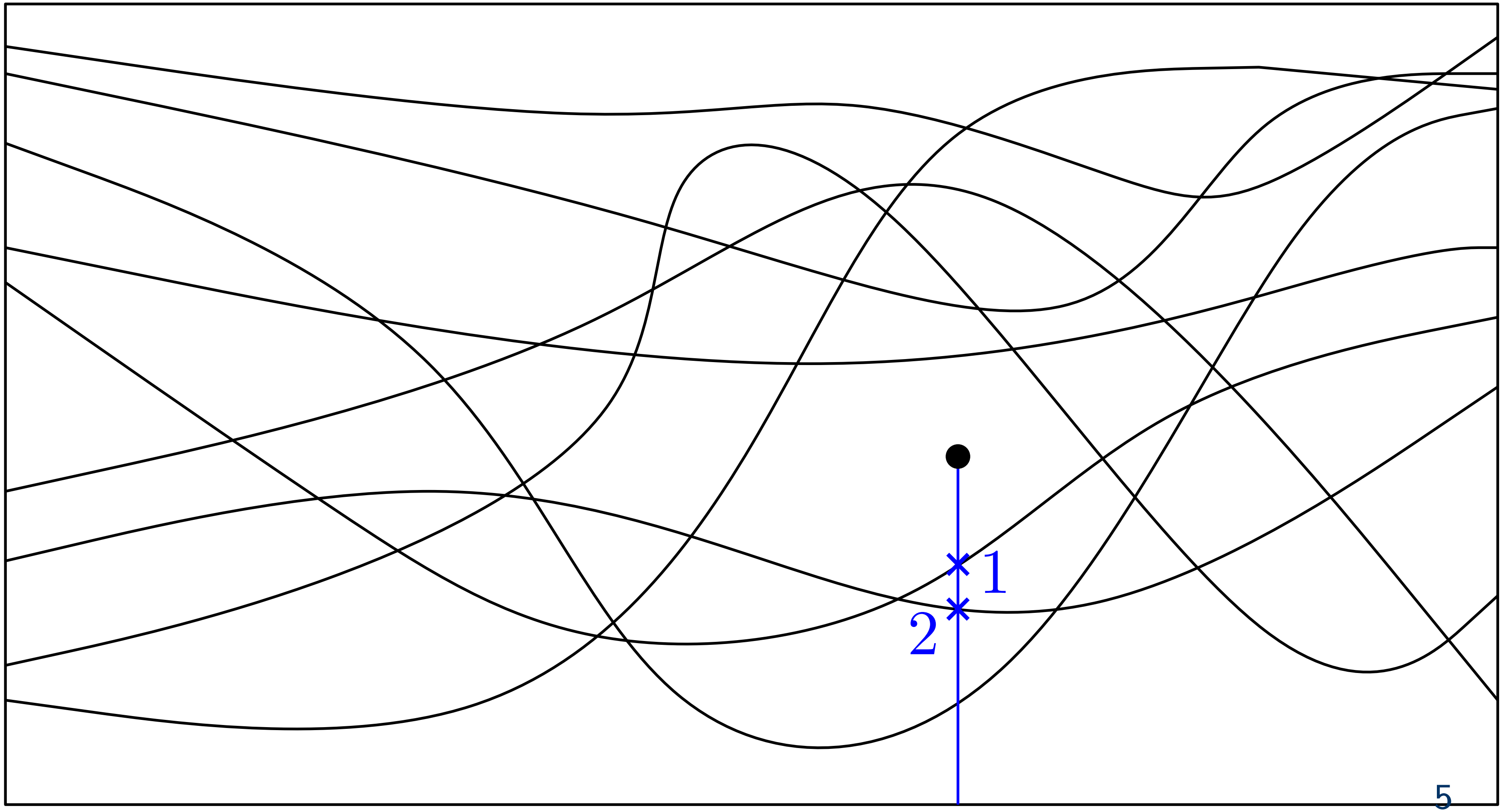
Vertical k-shallow cuttings



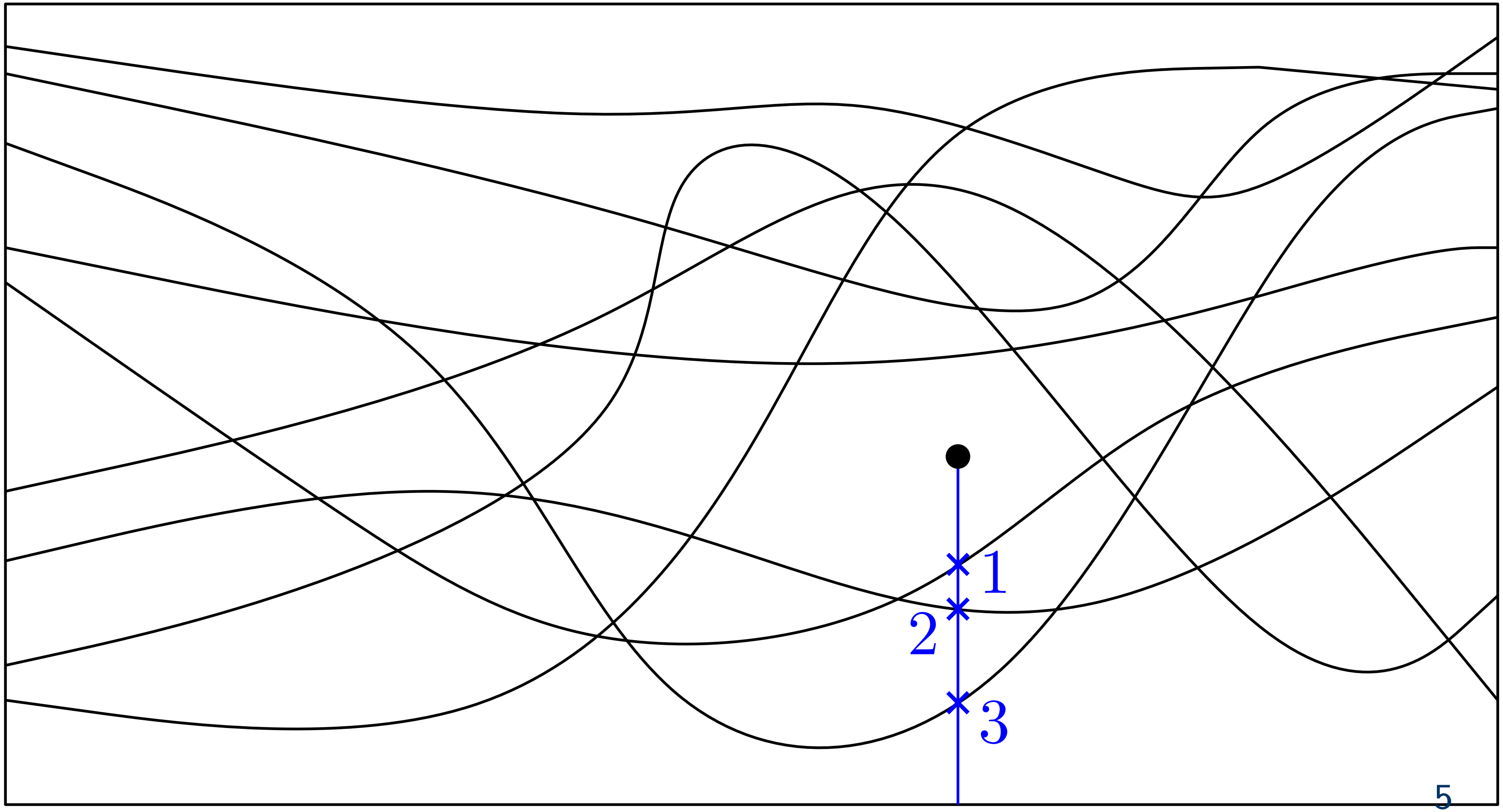
Vertical k-shallow cuttings



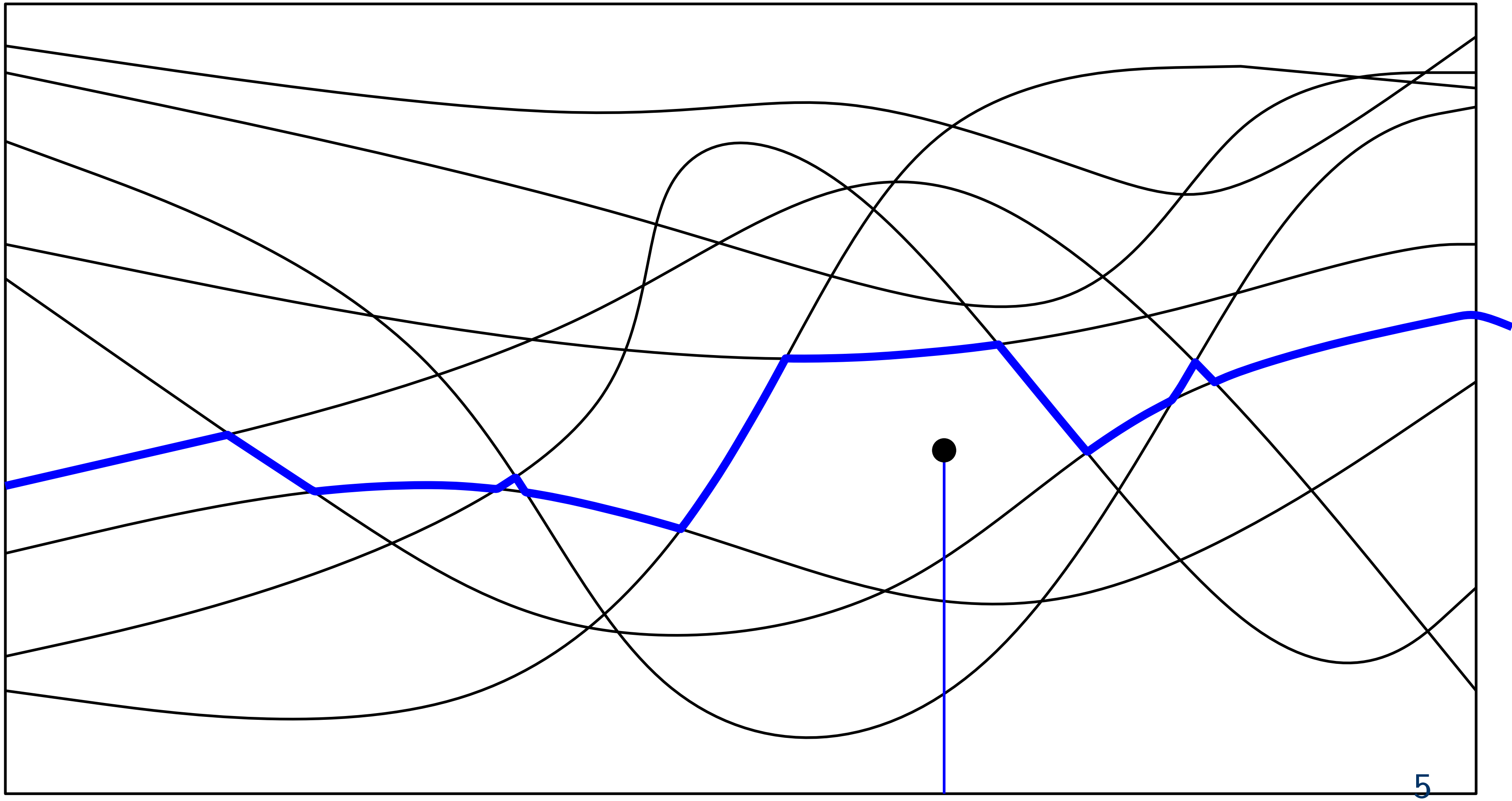
Vertical k-shallow cuttings



Vertical k-shallow cuttings

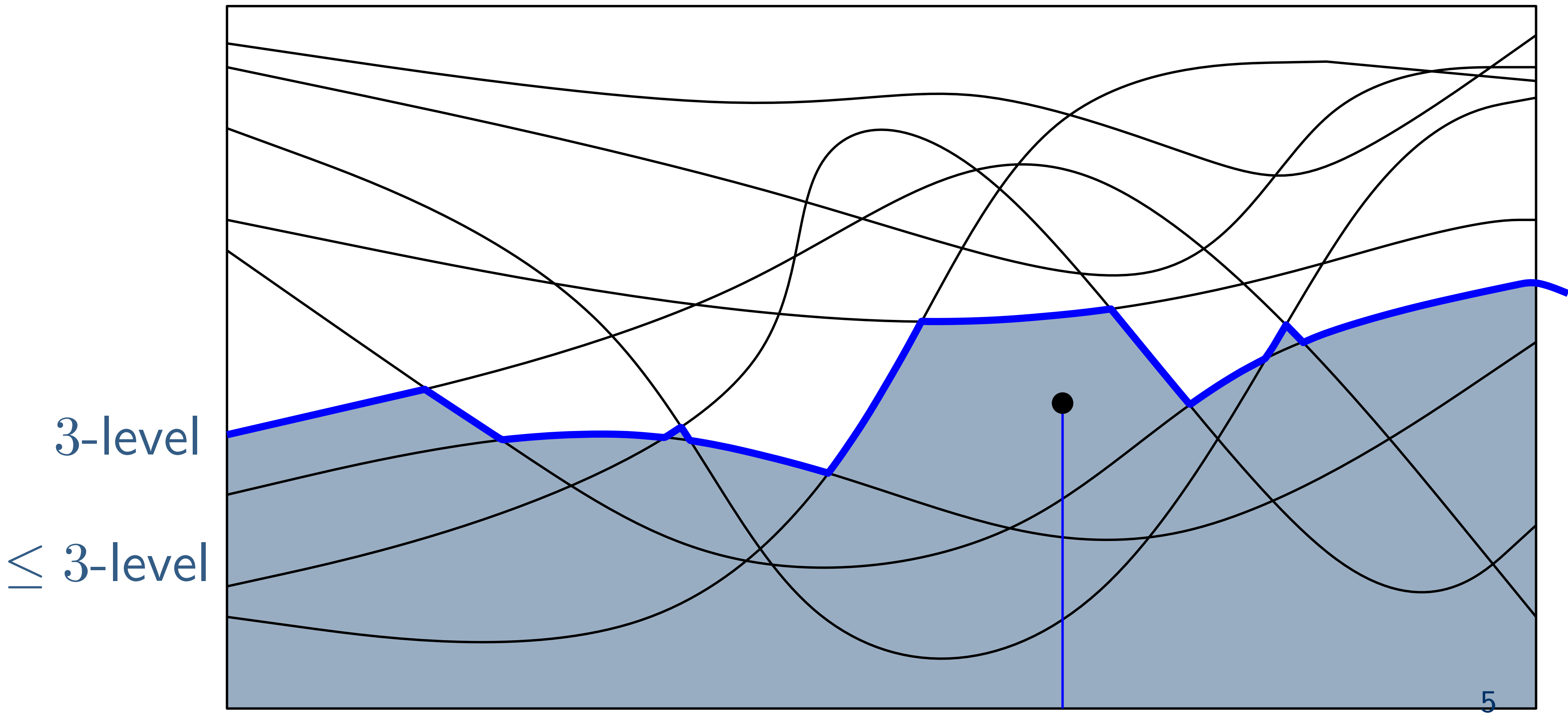


Vertical k-shallow cuttings



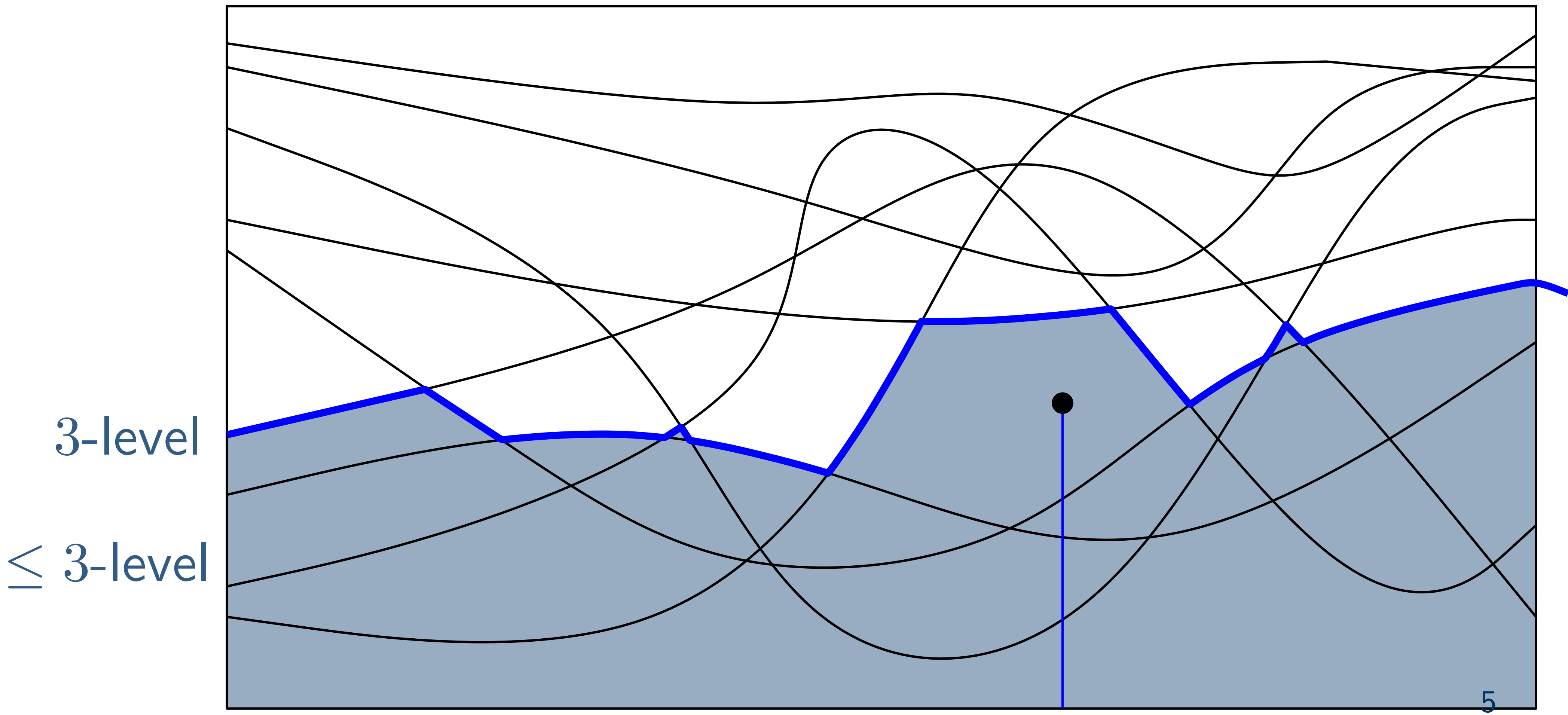
3-level

Vertical k-shallow cuttings



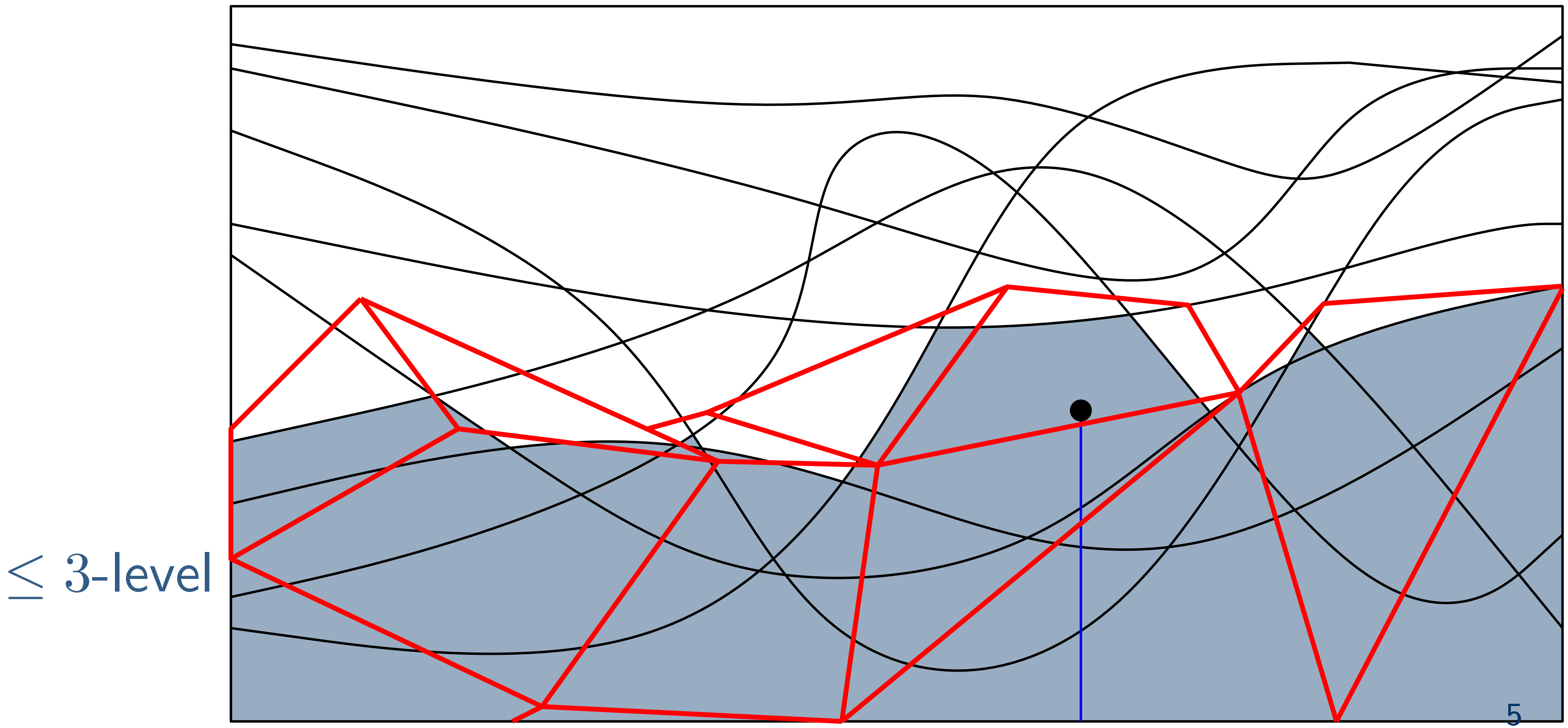
Vertical k-shallow cuttings

k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces



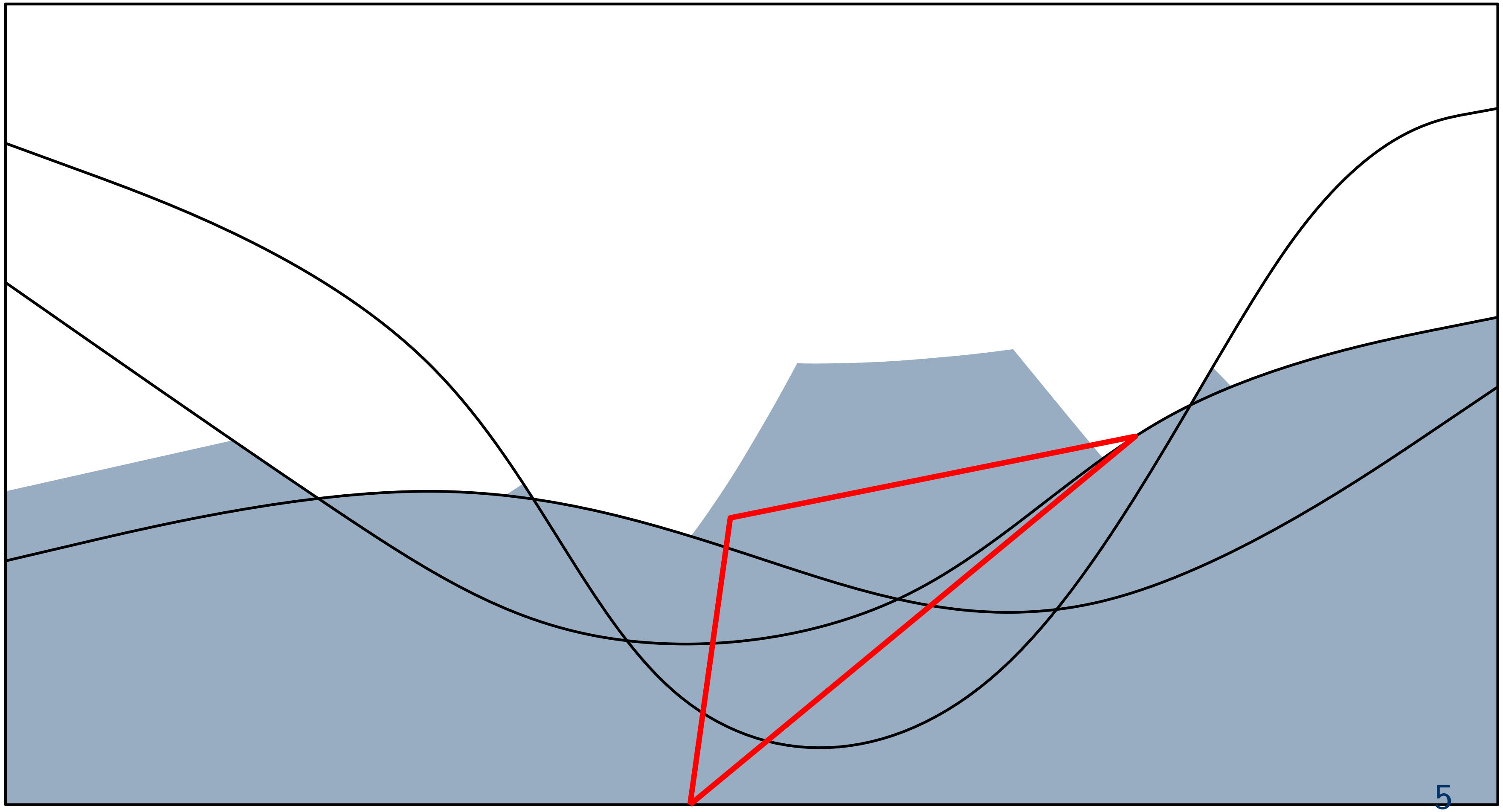
Vertical k-shallow cuttings

k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces



Vertical k-shallow cuttings

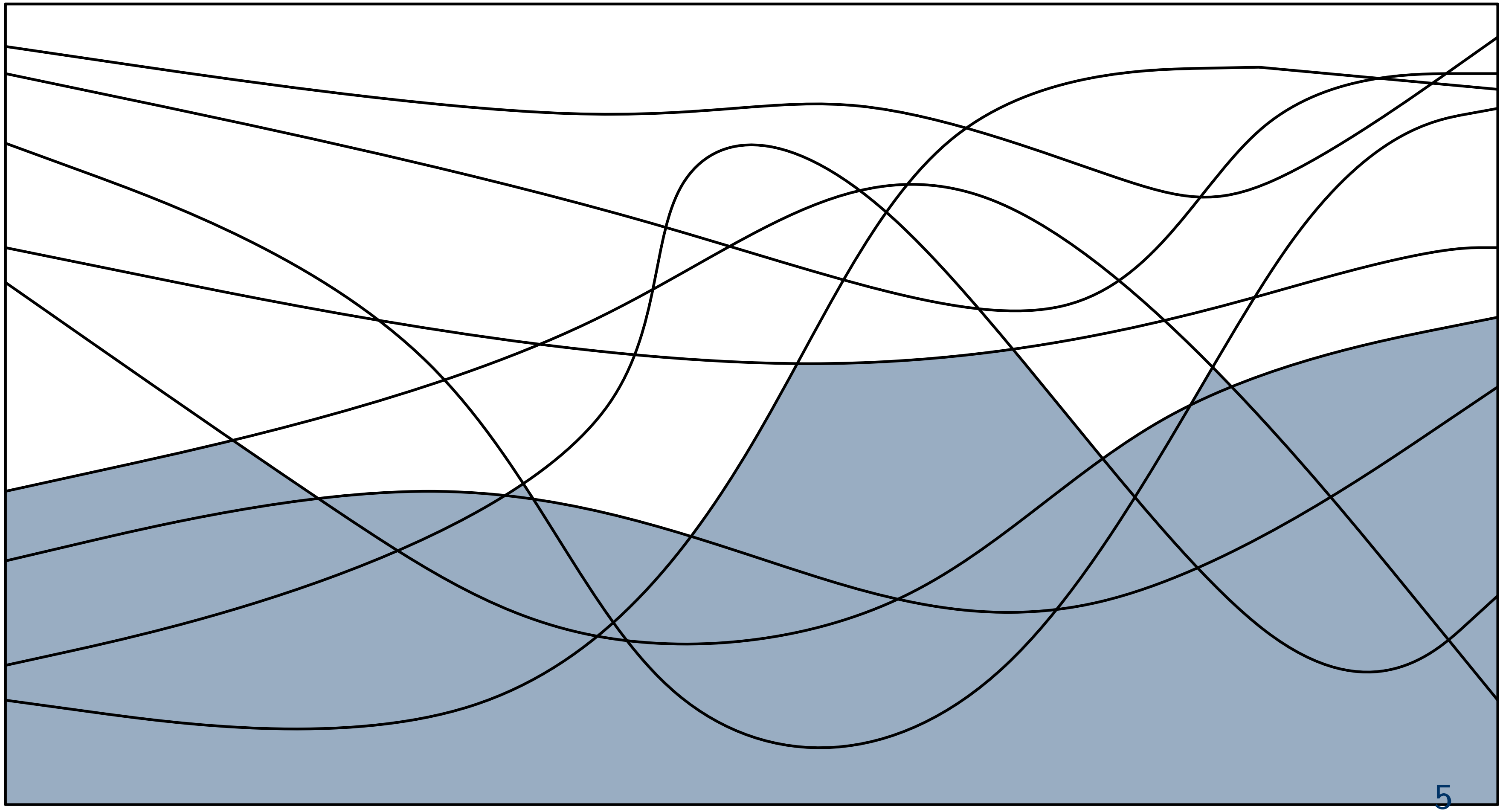
k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces



Vertical k-shallow cuttings

vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

obtained from an x-y-monotone terrain

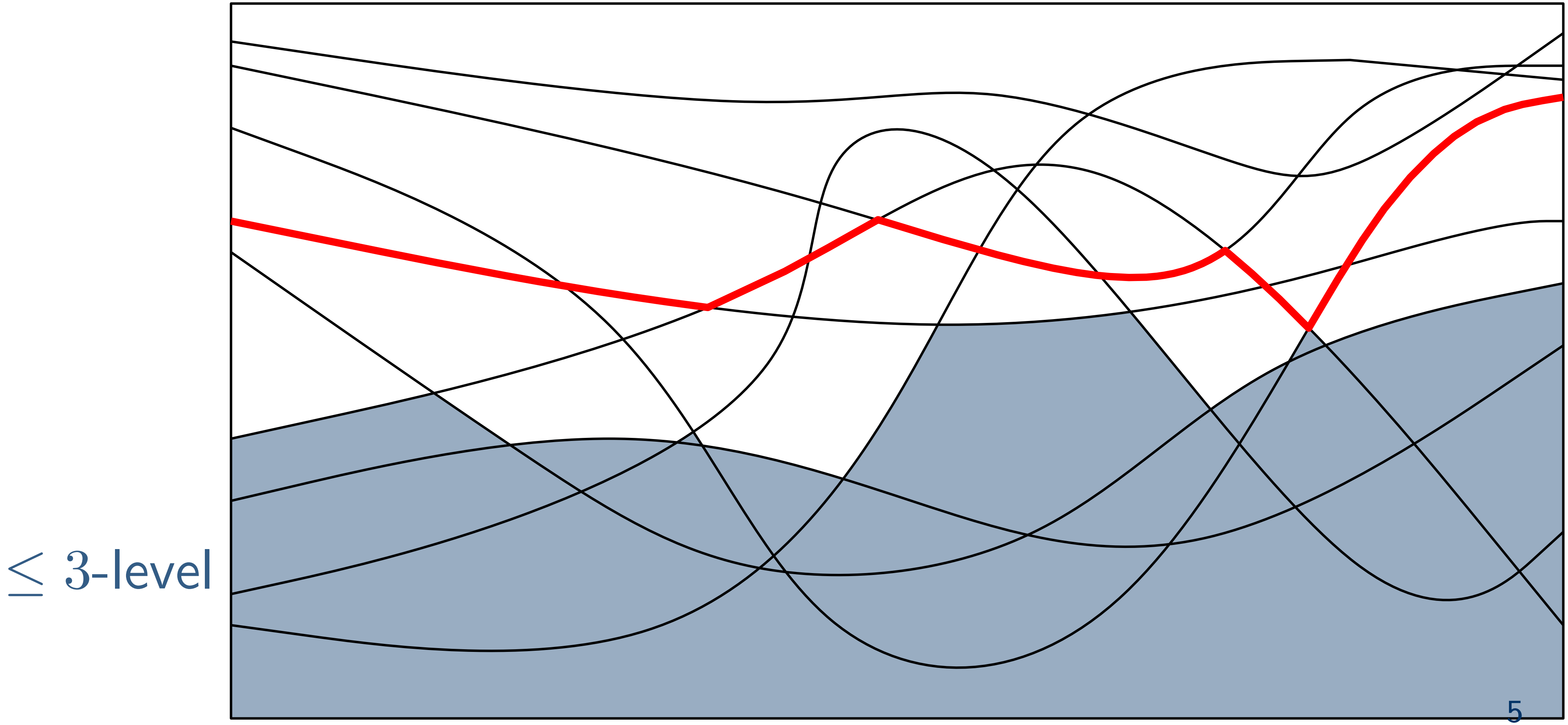


≤ 3-level

Vertical k-shallow cuttings

vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

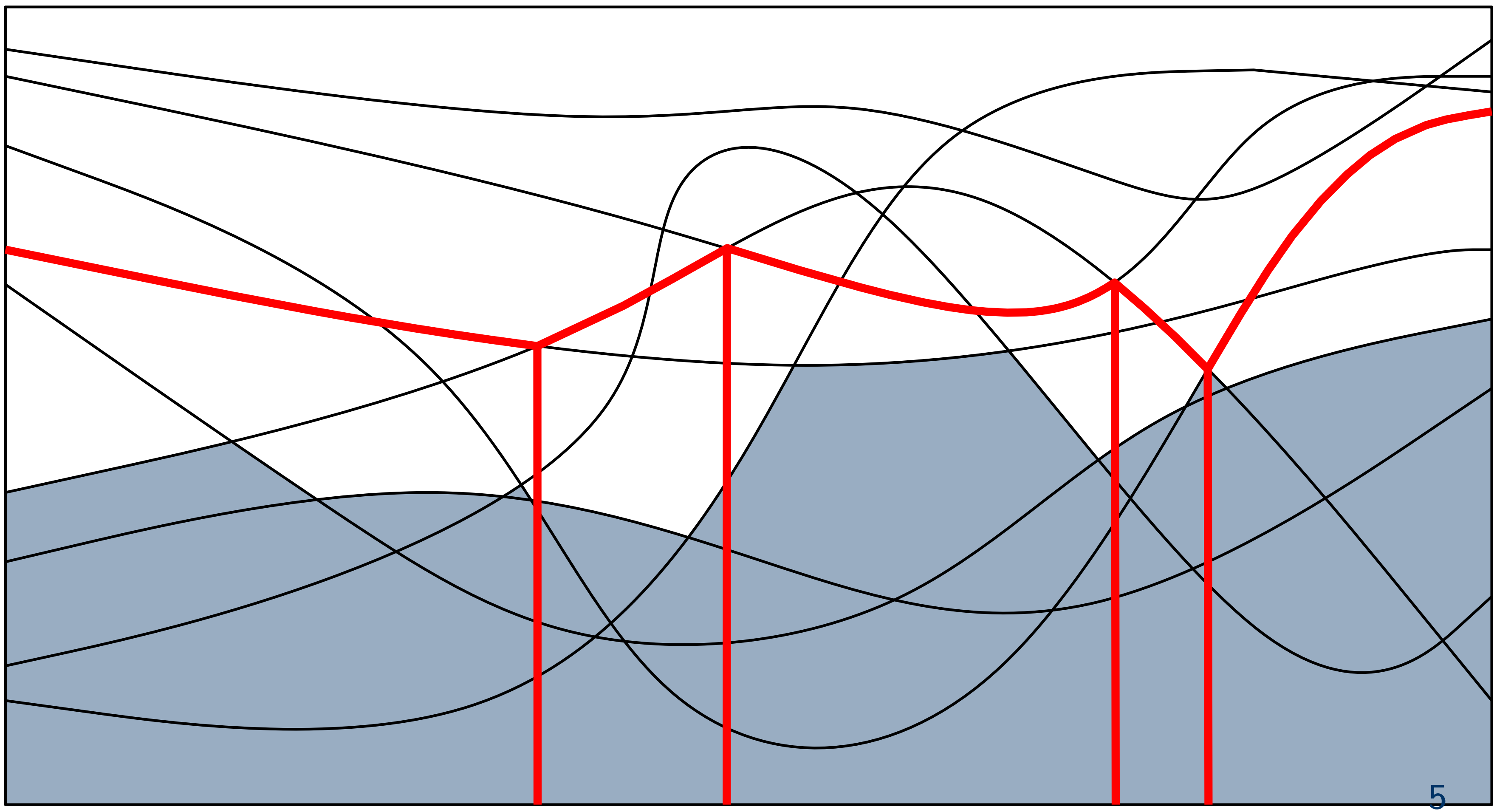
obtained from an x-y-monotone terrain



Vertical k-shallow cuttings

vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

obtained from an x-y-monotone terrain



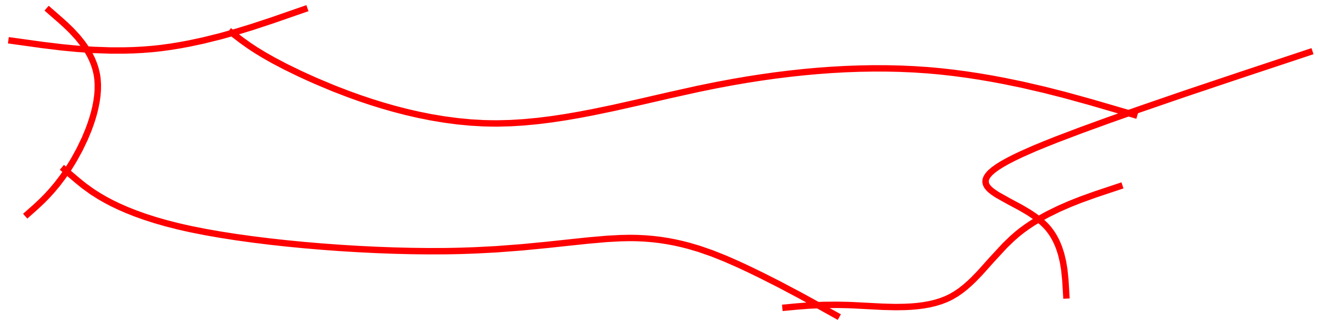
≤ 3-level

Vertical k-shallow cuttings

vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

obtained from an x-y-monotone terrain

$3D$

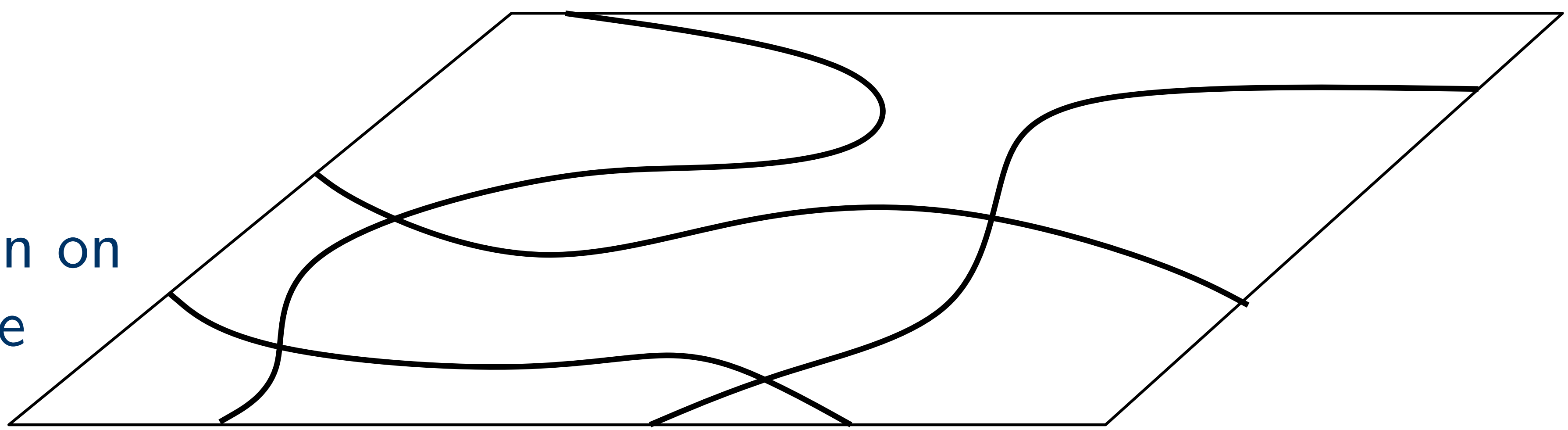


Vertical k-shallow cuttings

vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

obtained from an x-y-monotone terrain

projection on x-y-plane

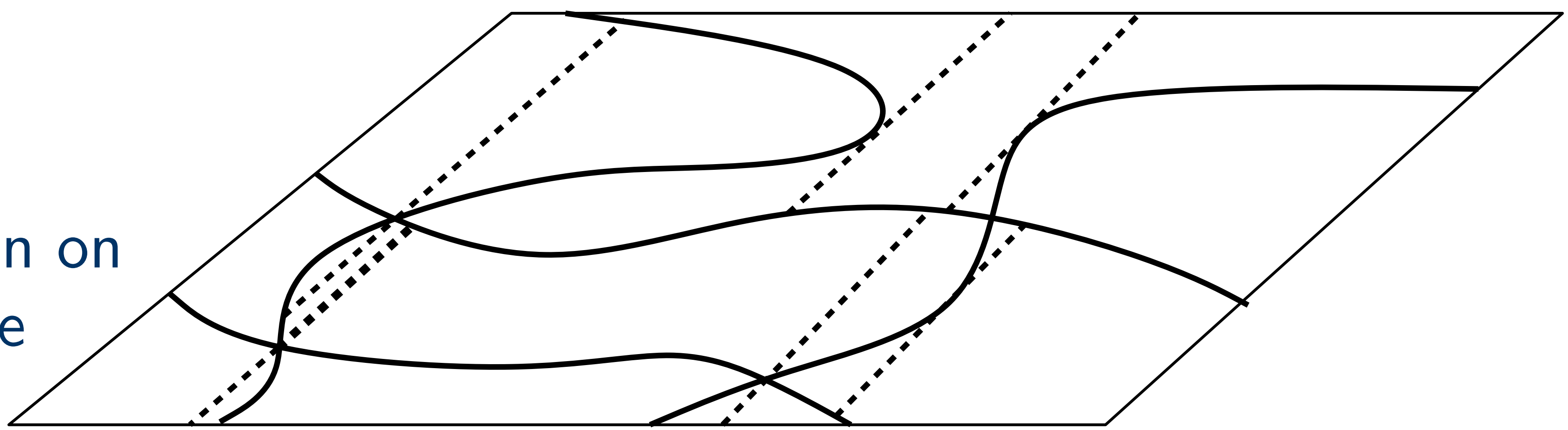


Vertical k-shallow cuttings

vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

obtained from an x-y-monotone terrain

projection on x-y-plane

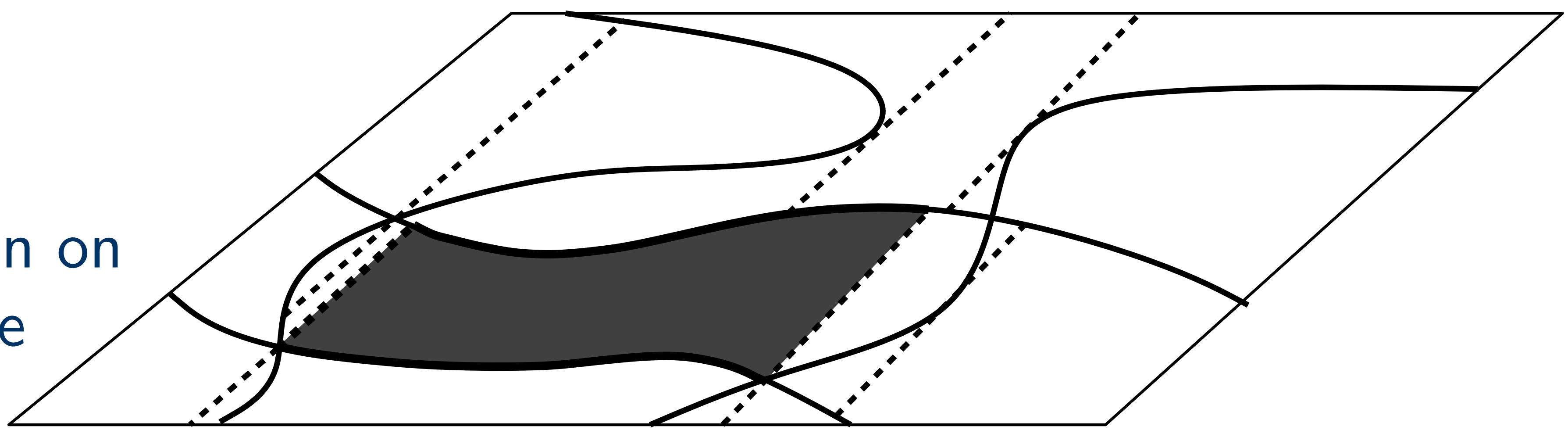


Vertical k-shallow cuttings

vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

obtained from an x-y-monotone terrain

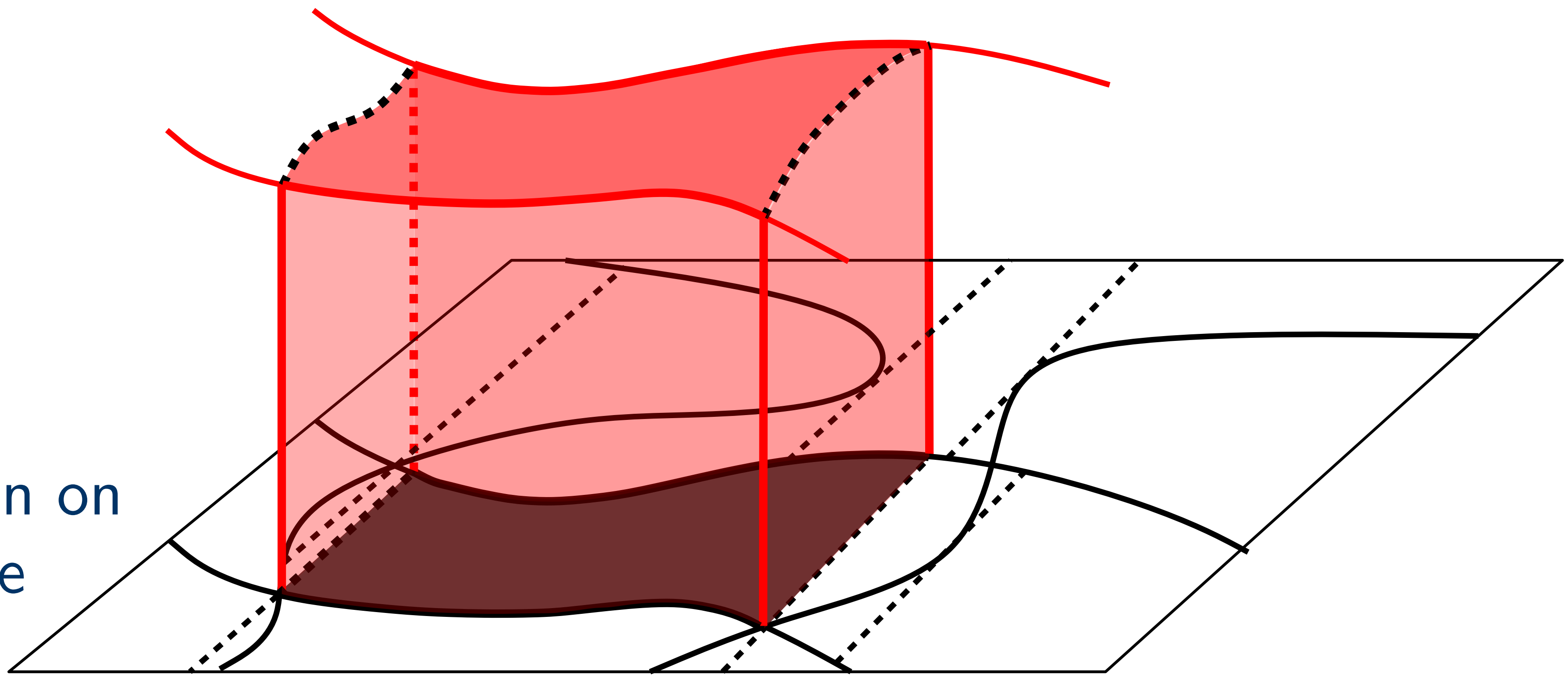
projection on x-y-plane



Vertical k-shallow cuttings

vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

obtained from an x-y-monotone terrain

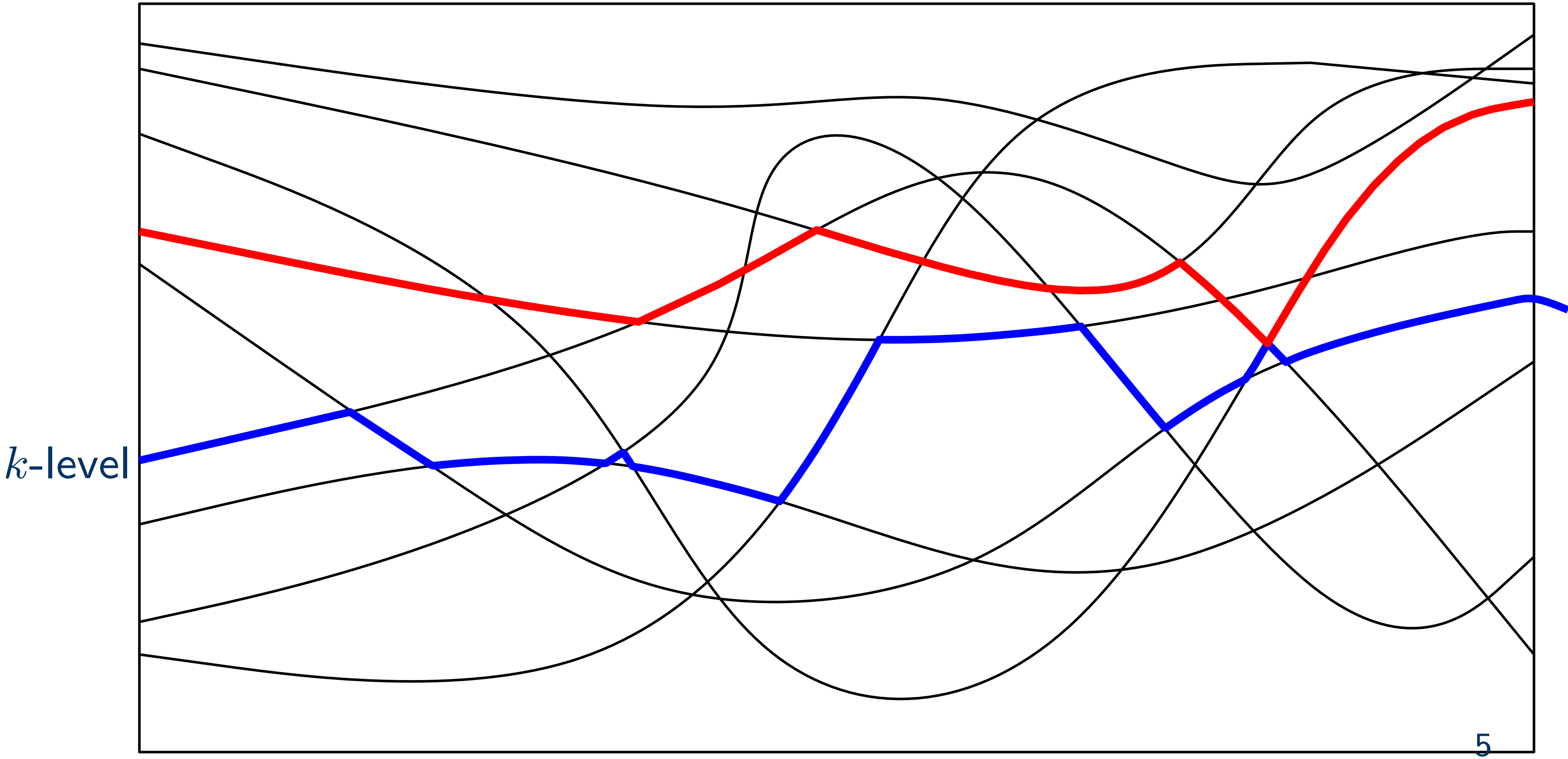


projection on x-y-plane

Vertical k-shallow cuttings

vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

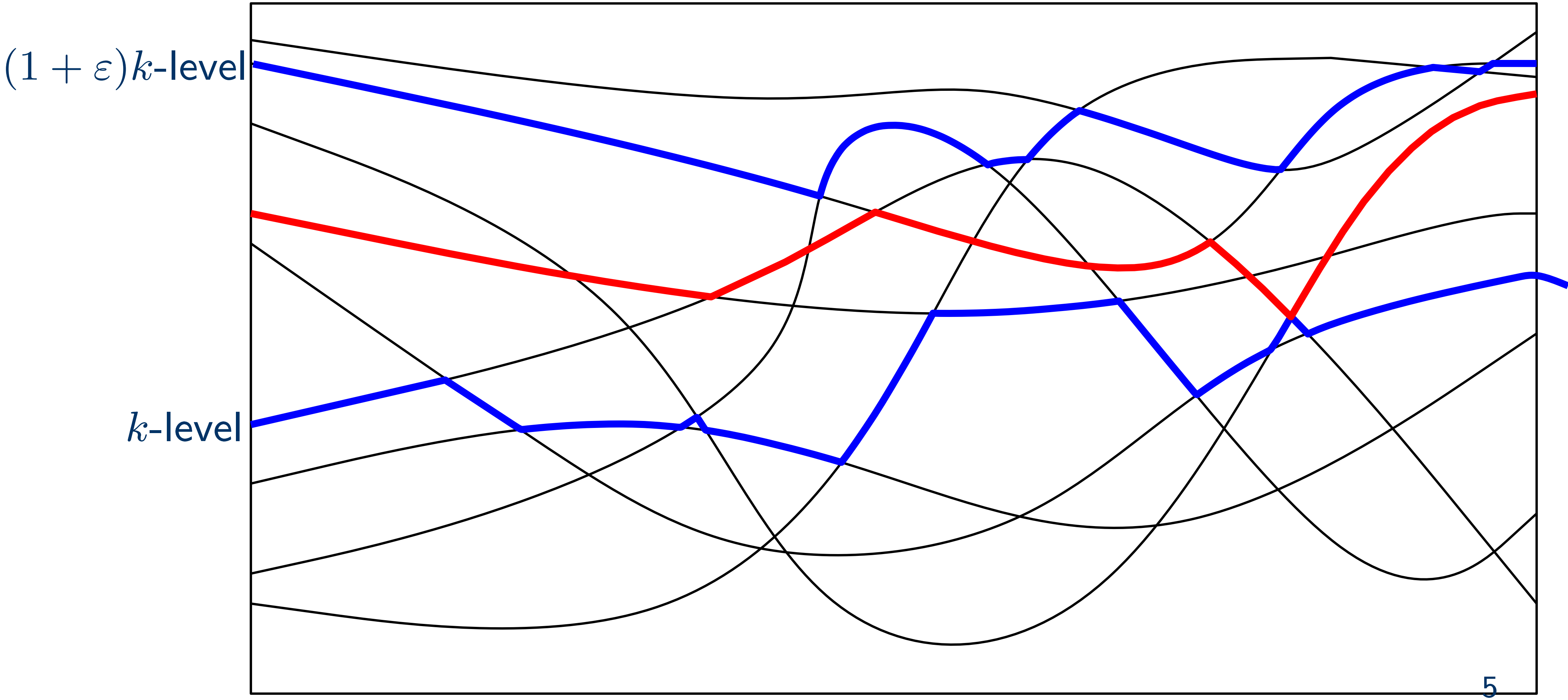
obtained from an x-y-monotone terrain



Vertical k-shallow cuttings

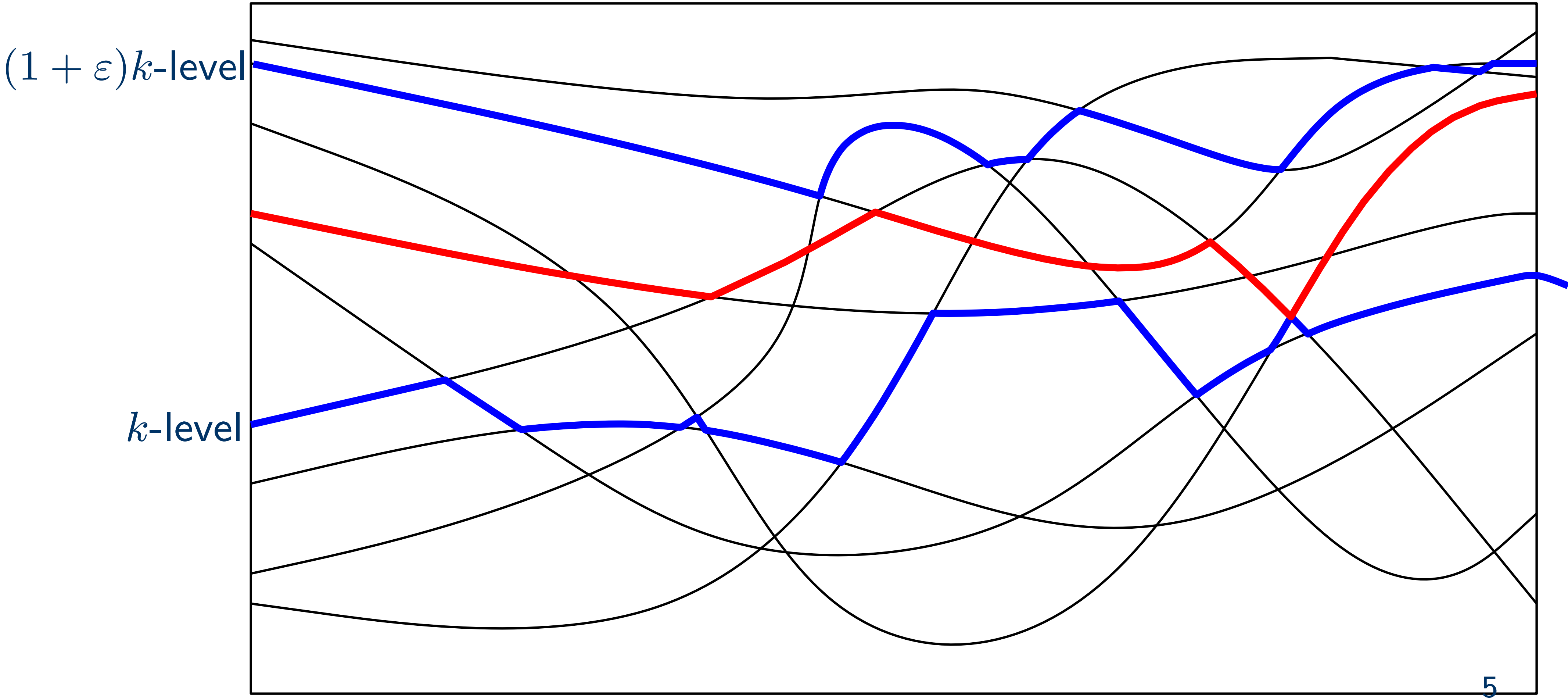
vertical k-shallow cutting: covering of $\leq k$-level with "few" cells s.t. each cell intersects $O(k)$ surfaces

obtained from an x-y-monotone terrain

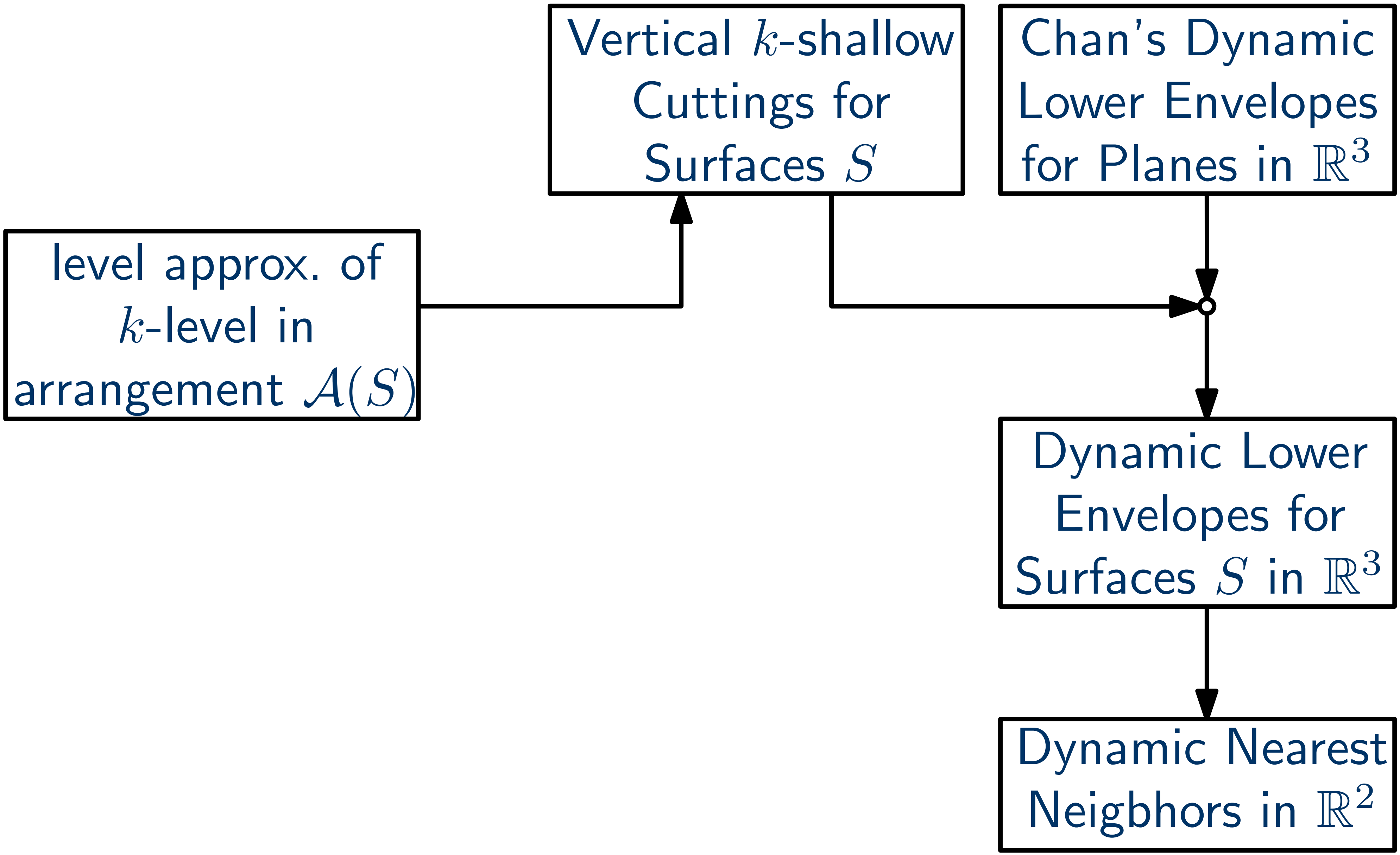


Vertical k-shallow cuttings

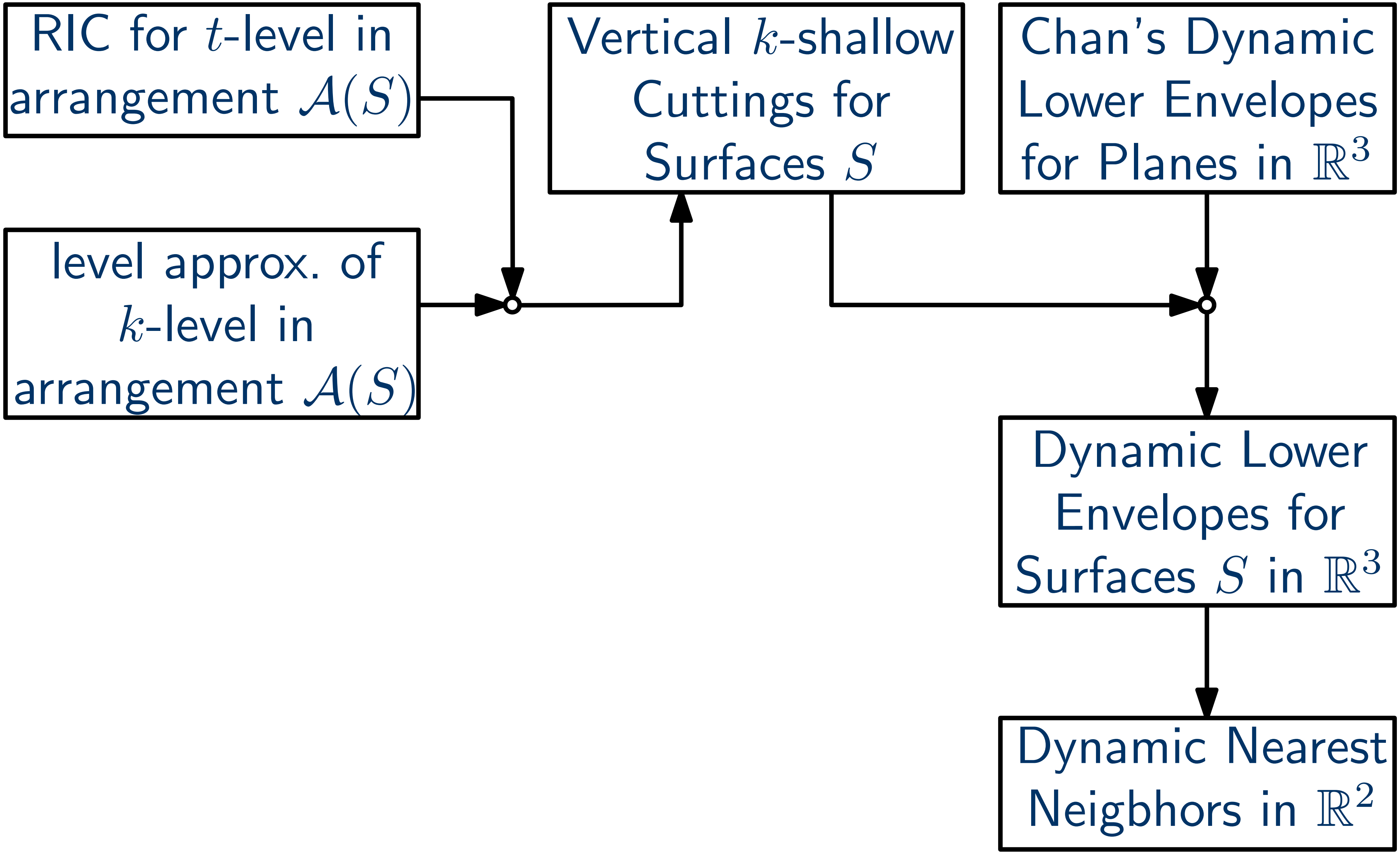
Theorem: The $t = O(\log n)$-level of a sample of size $O((n/k)\log n)$ yields a level approximation with **expected complextity** $O((n/k)\log^2 n)$



Overview

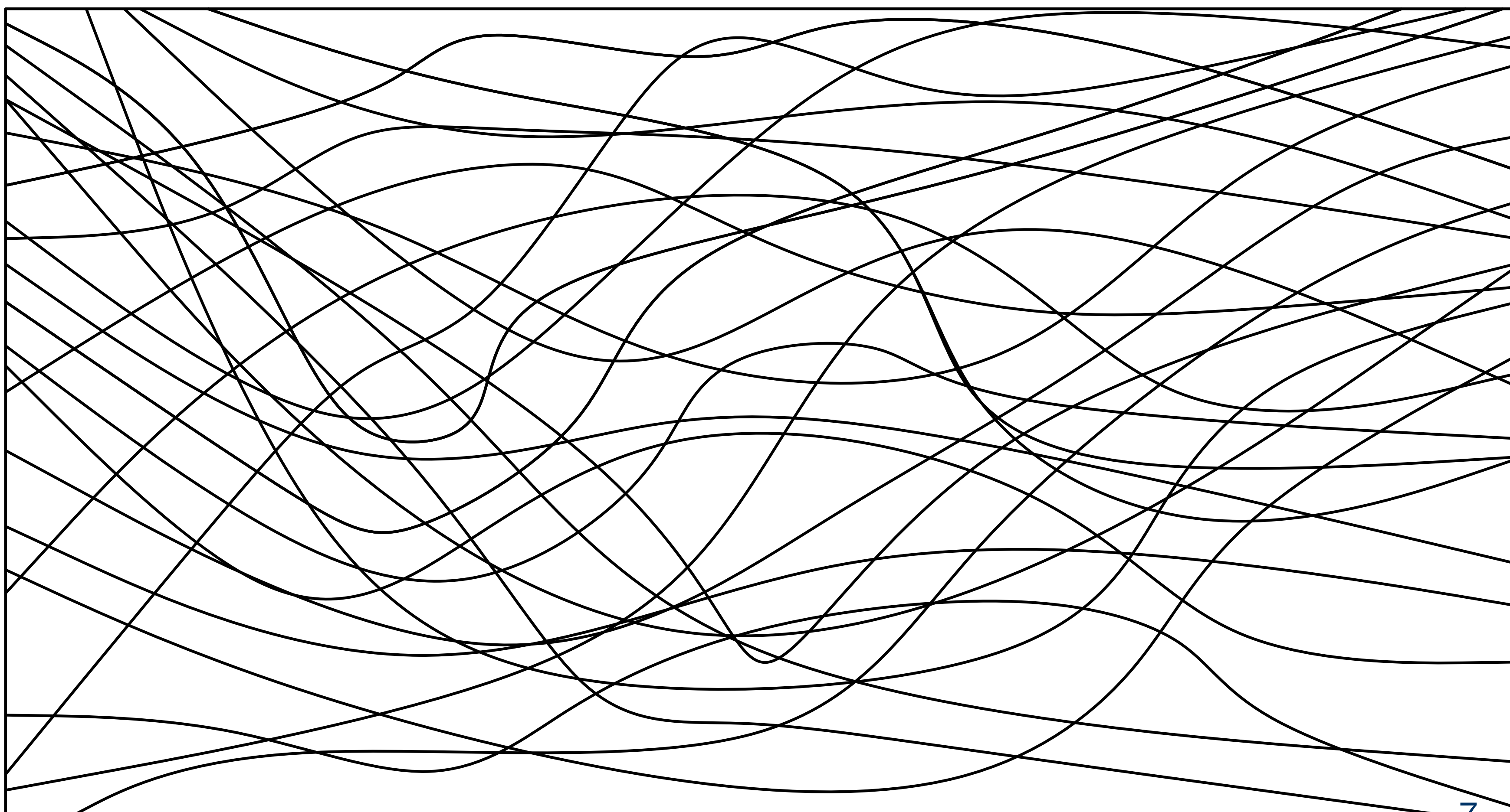

Vertical k-shallow Cuttings for Surfaces S
Chan's Dynamic Lower Envelopes for Planes in $\mathbb{R}^3$
level approx. of k-level in arrangement $\mathcal{A}(S)$
Dynamic Lower Envelopes for Surfaces S in $\mathbb{R}^3$
Dynamic Nearest Neigbhors in $\mathbb{R}^2$

Overview


RIC for t-level in arrangement $\mathcal{A}(S)$
level approx. of k-level in arrangement $\mathcal{A}(S)$
Vertical k-shallow Cuttings for Surfaces S
Chan's Dynamic Lower Envelopes for Planes in $\mathbb{R}^3$
Dynamic Lower Envelopes for Surfaces S in $\mathbb{R}^3$
Dynamic Nearest Neigbhors in $\mathbb{R}^2$

Randomized Incremental Construction

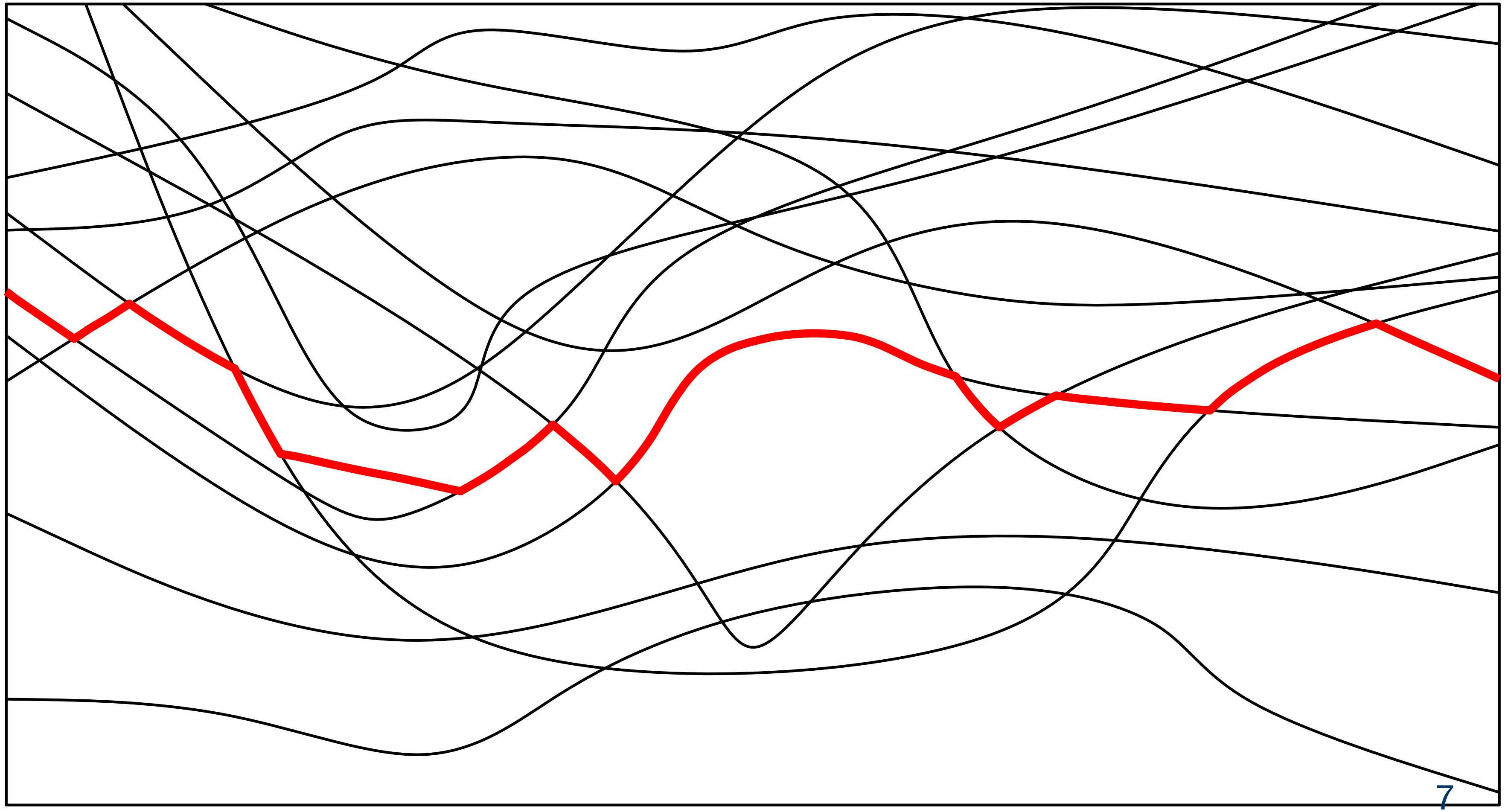
Strategy:
- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S



Randomized Incremental Construction

Strategy:
- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k) \log n)$ steps

t-level

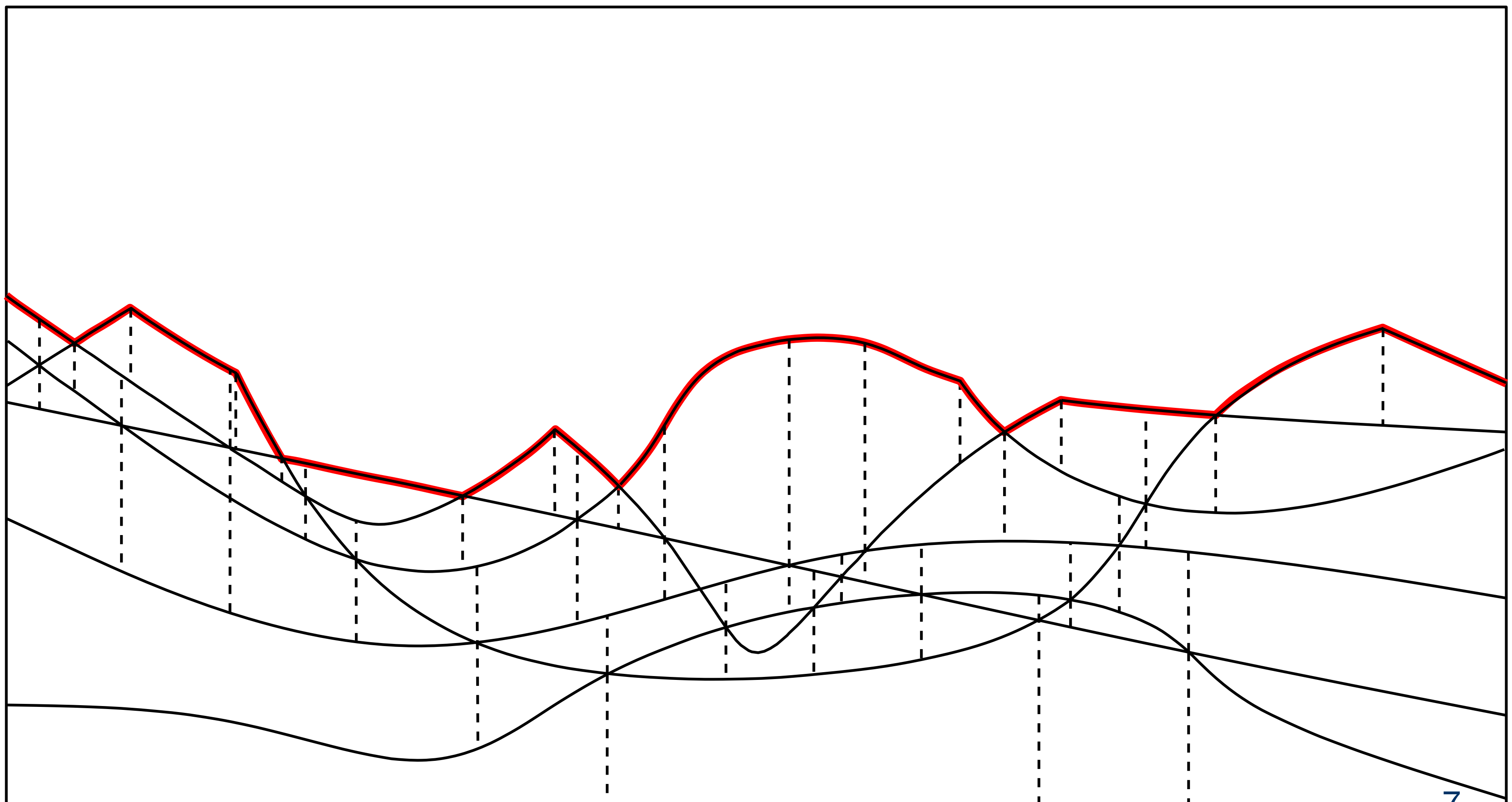


Randomized Incremental Construction

Strategy:

- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k)\log n)$ steps

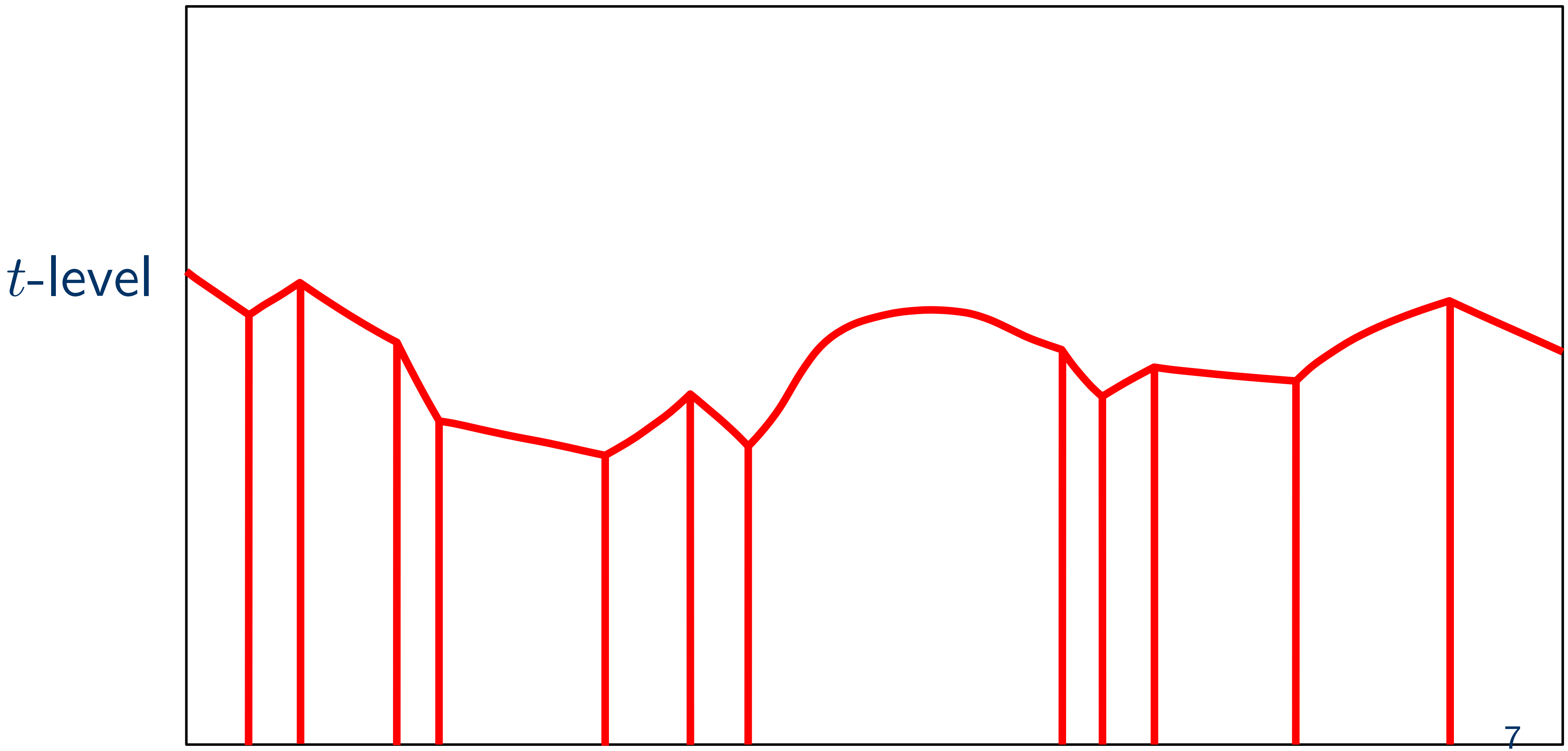
t-level



Randomized Incremental Construction

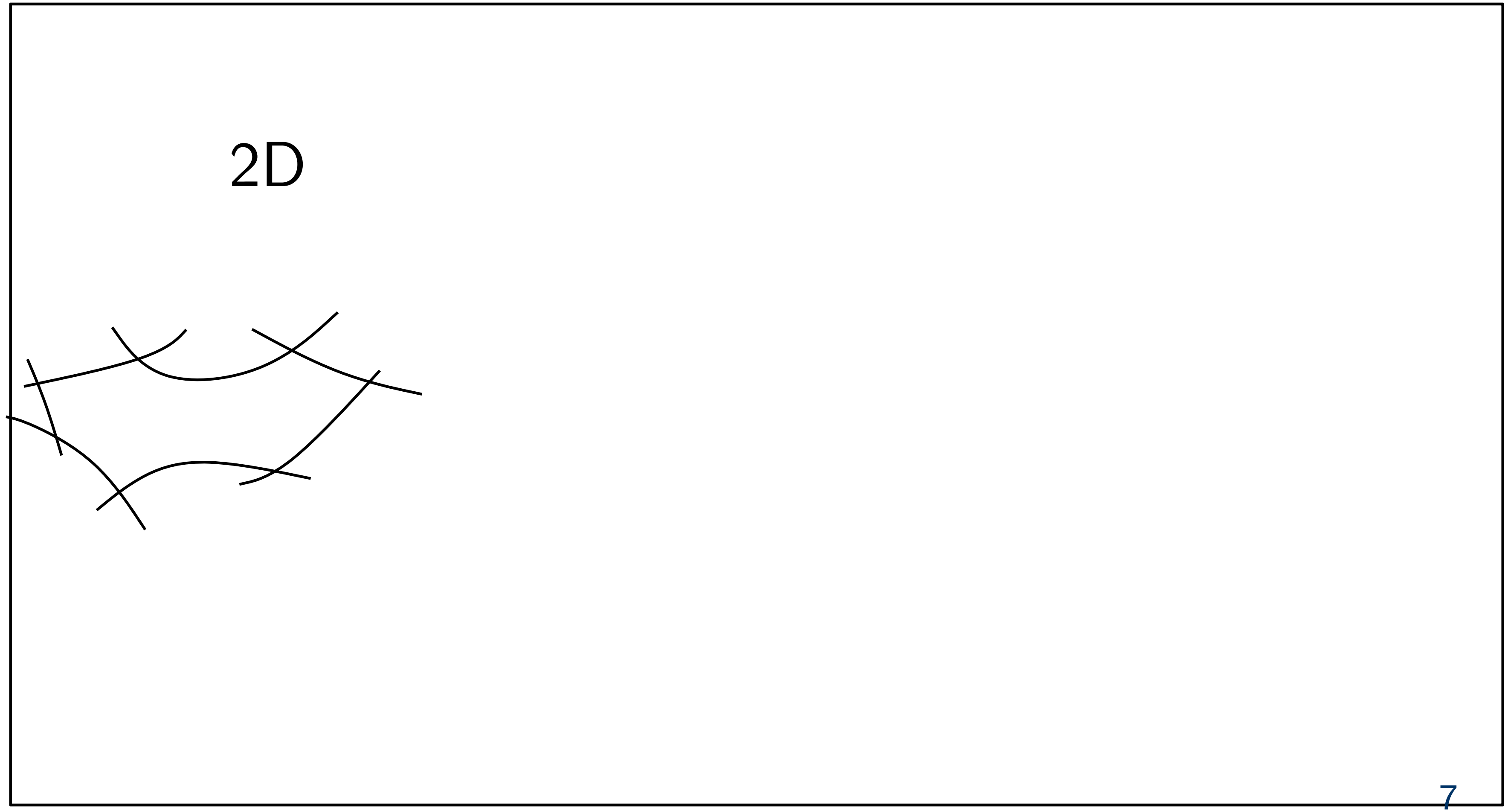
Strategy:

- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k)\log n)$ steps



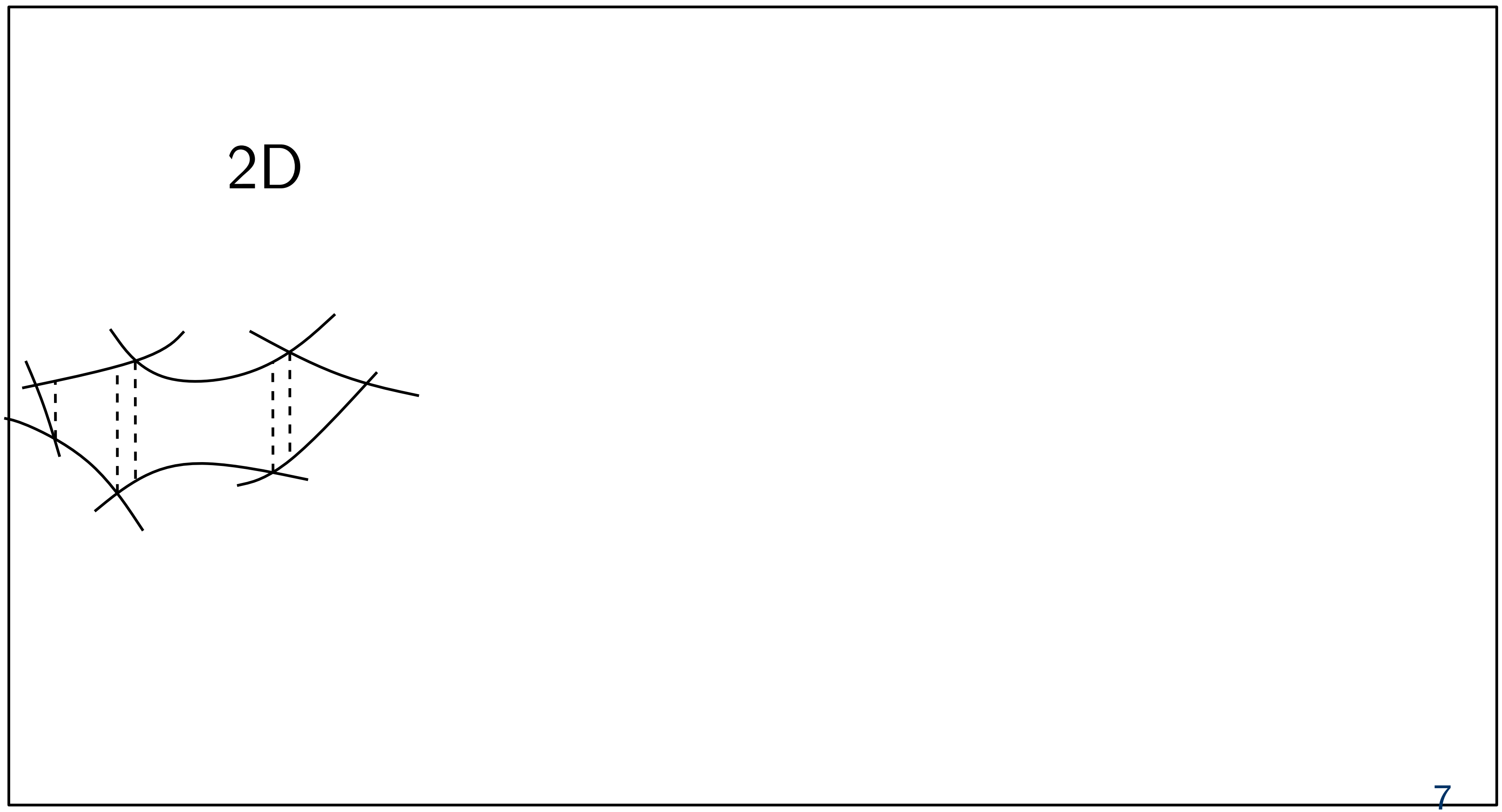
Randomized Incremental Construction

Strategy:
- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k) \log n)$ steps



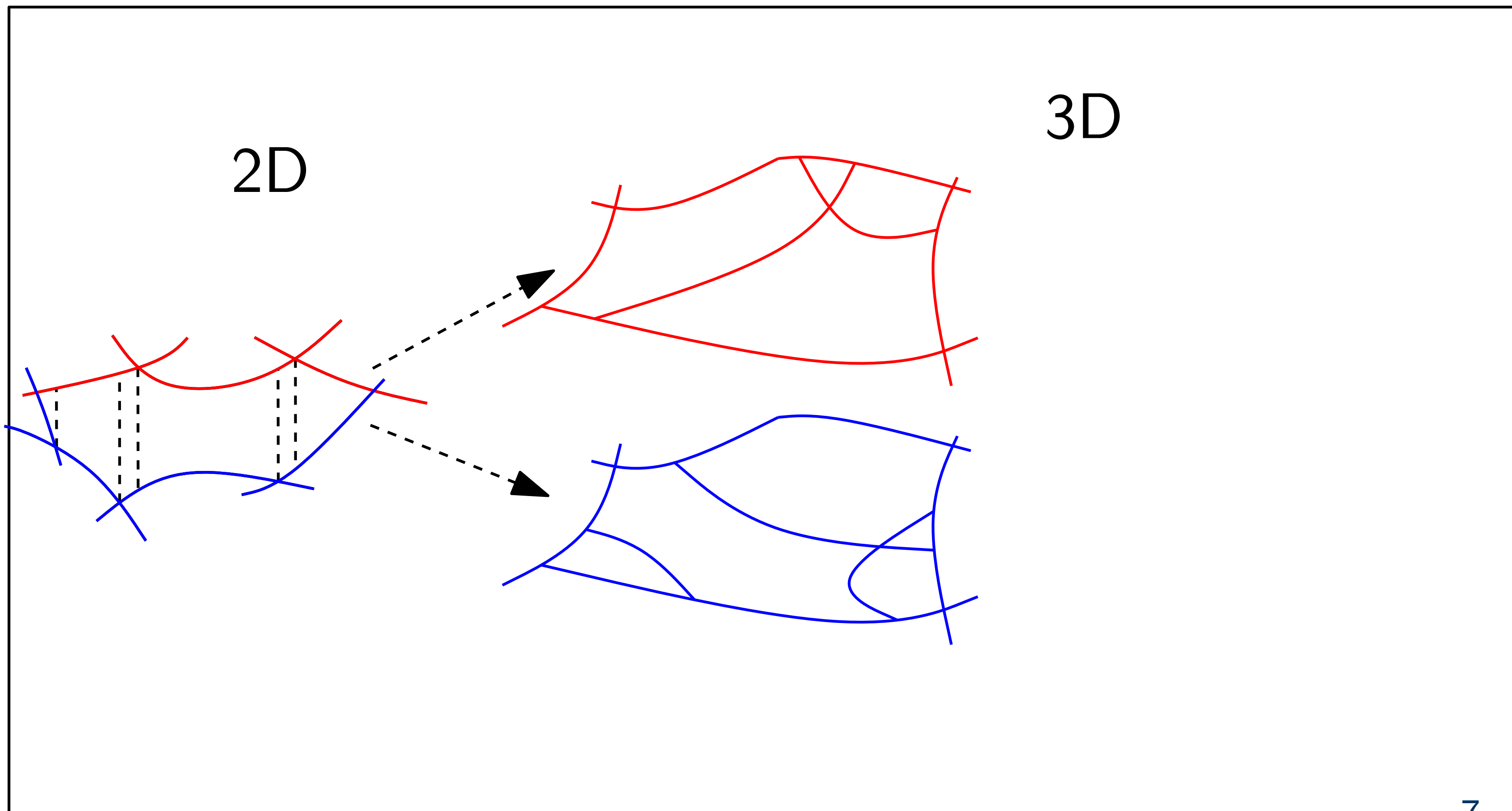
Randomized Incremental Construction

Strategy:
- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k)\log n)$ steps



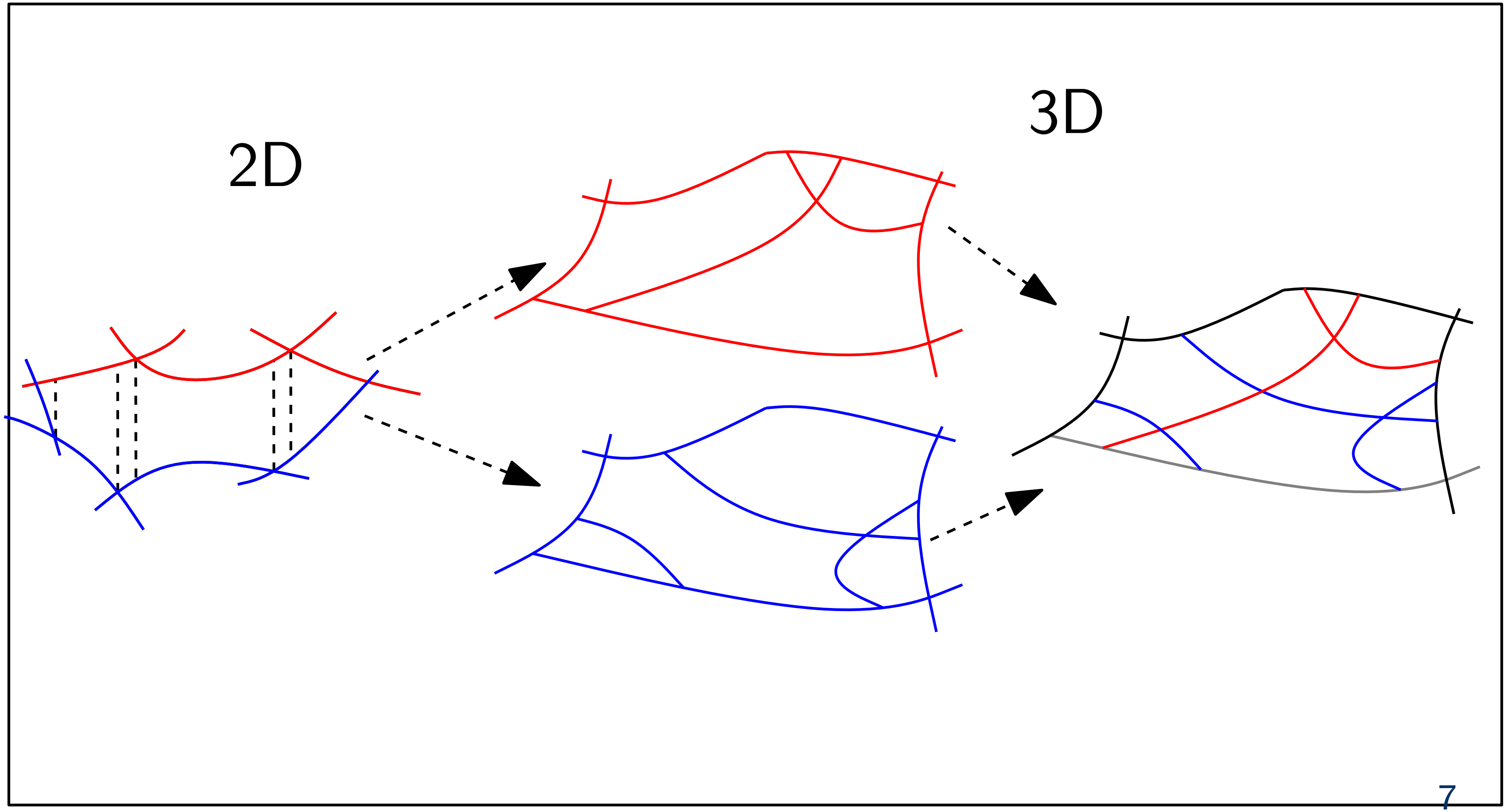
Randomized Incremental Construction

Strategy:
- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k)\log n)$ steps



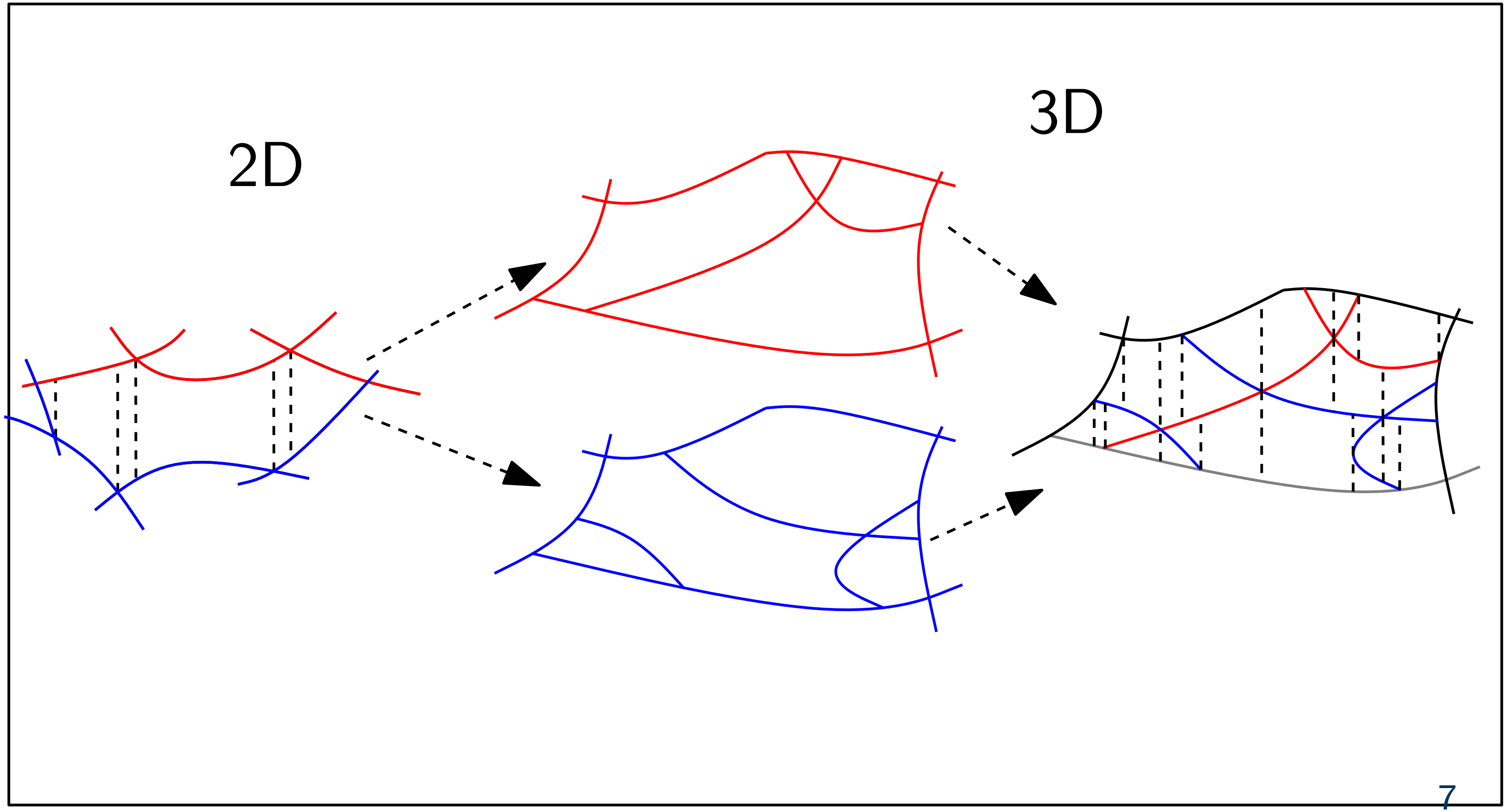
Randomized Incremental Construction

Strategy:
- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k)\log n)$ steps



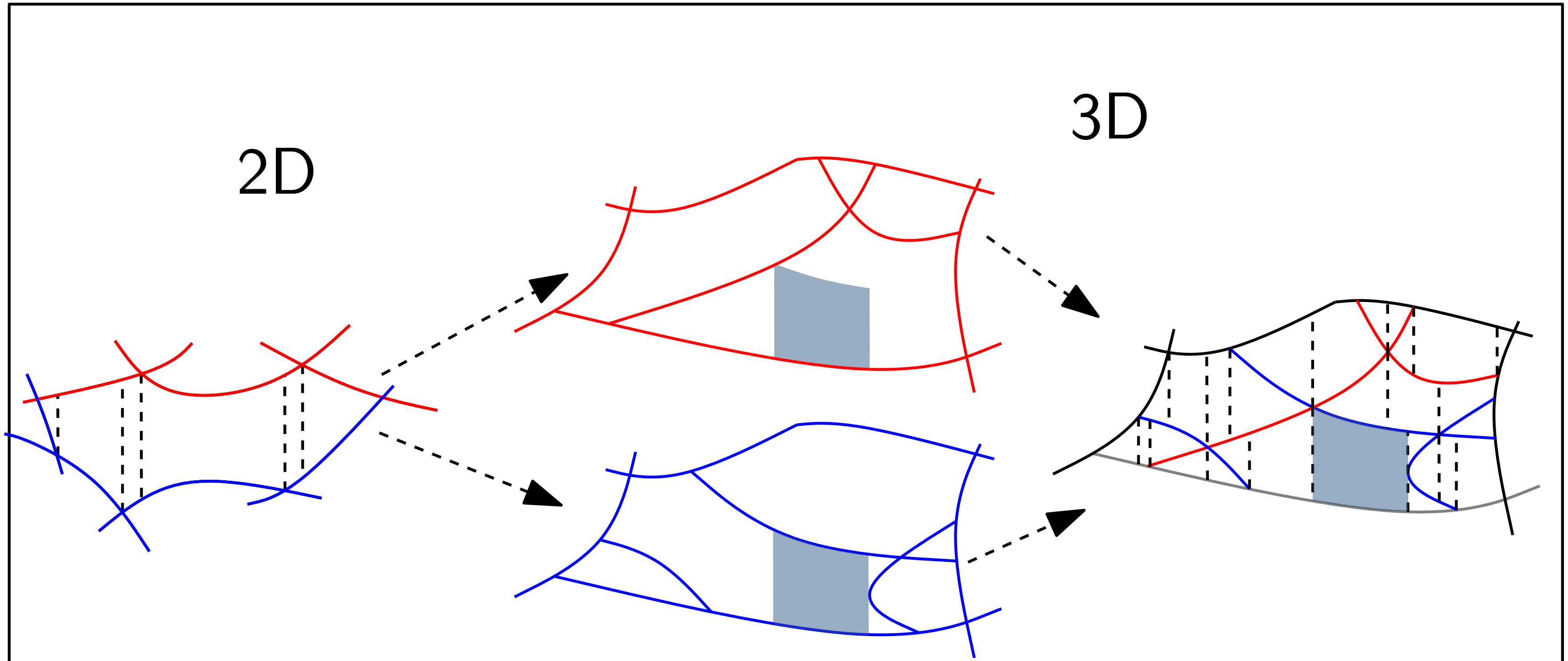
Randomized Incremental Construction

Strategy:
- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k)\log n)$ steps



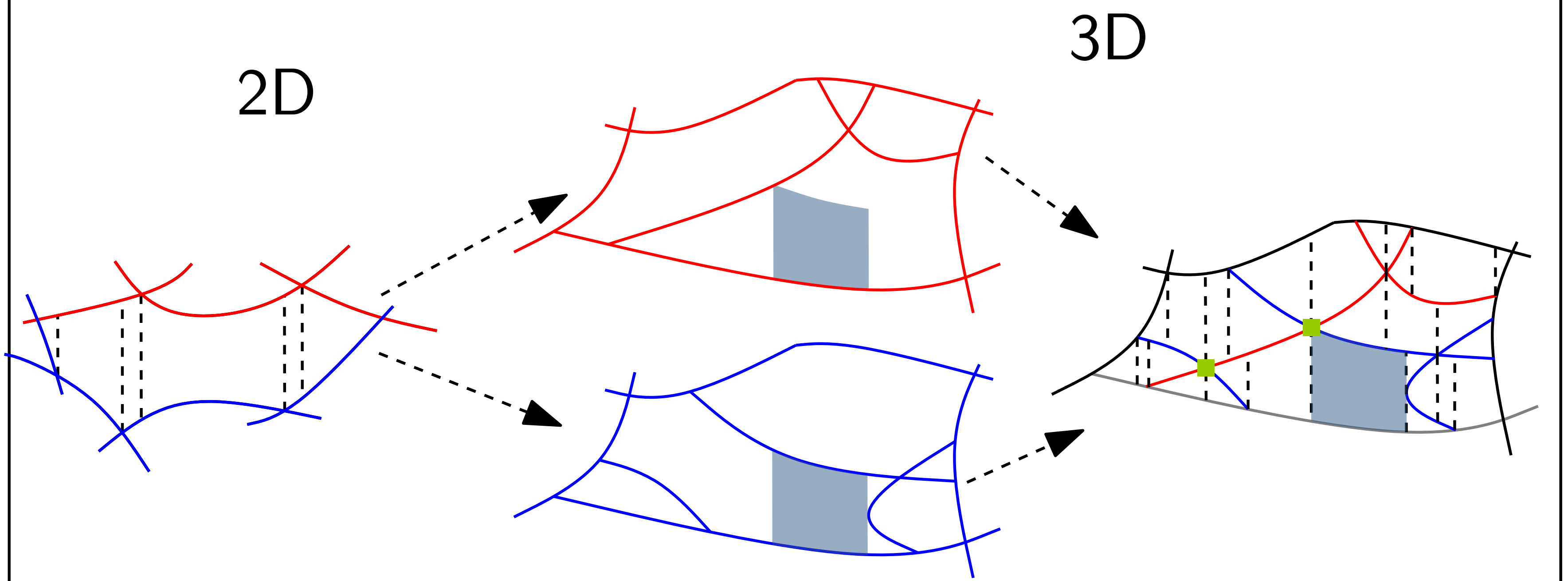
Randomized Incremental Construction

Strategy:
- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k)\log n)$ steps

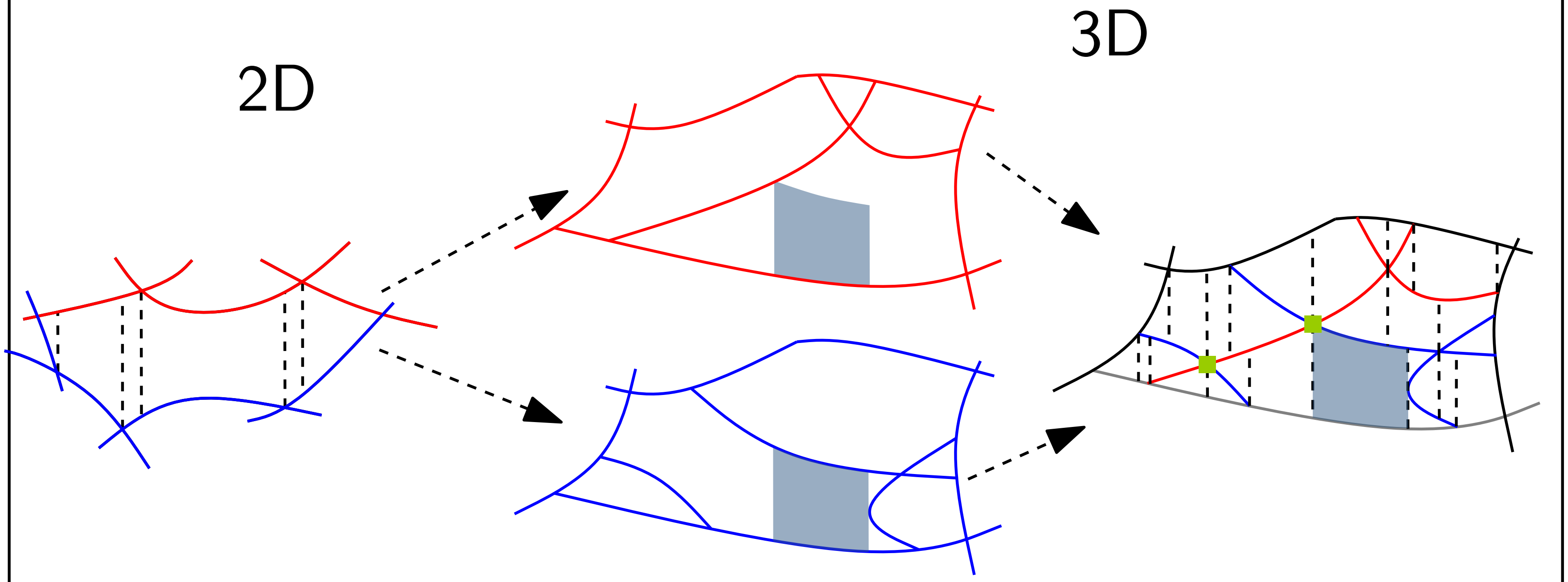


Randomized Incremental Construction

Strategy:
- take rand. perm. $s_1, s_2, s_3, s_4, \ldots, s_n$
- perform RIC for $t = O(\log n)$ level of surfaces S
- stop after $O((n/k)\log n)$ steps

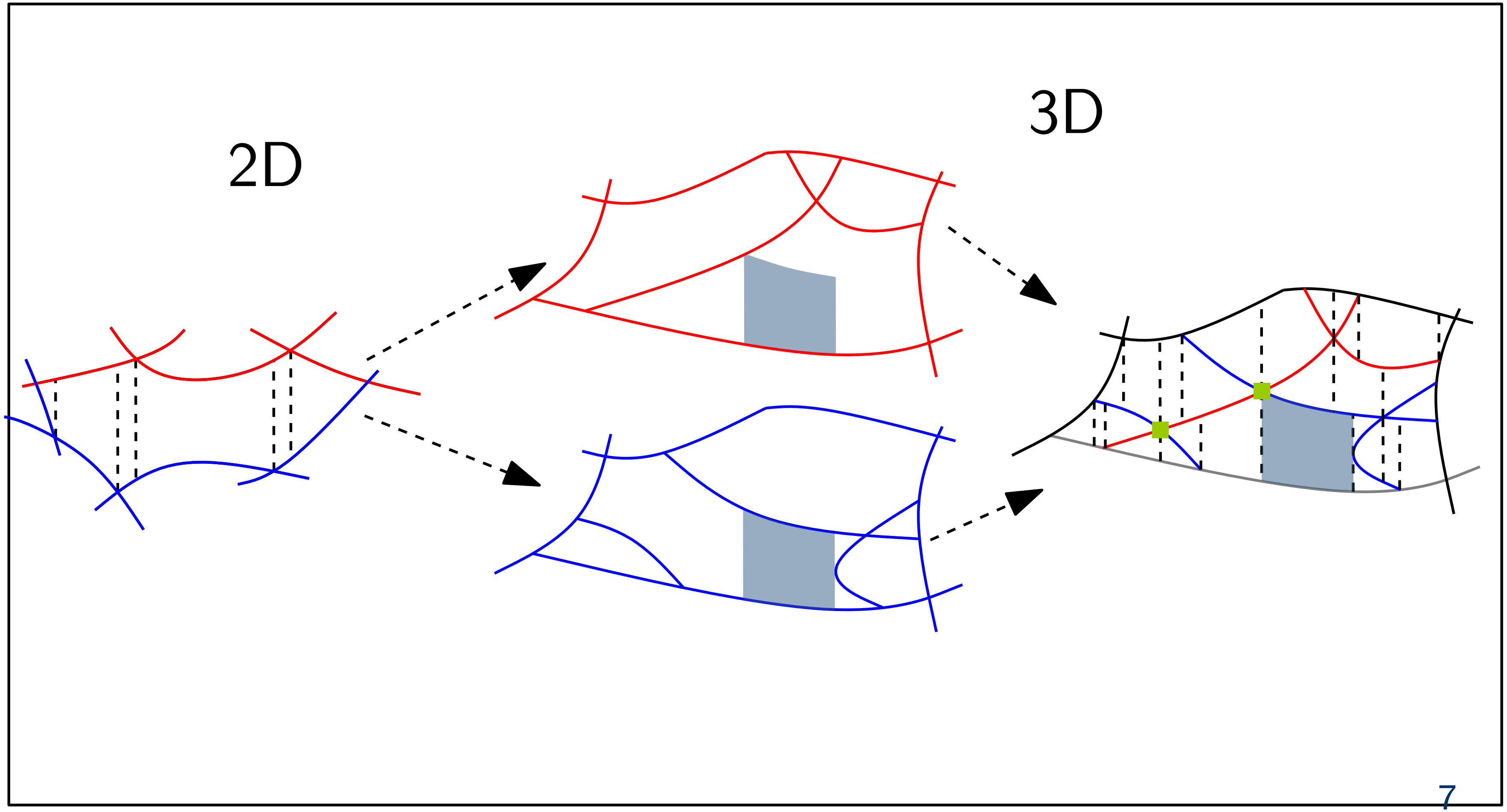


Theorem: The vertical decomposition of the $\leq t$ level of $\mathcal{A}(S)$ has complexity $O(nt\lambda_{s+2}(t))$.

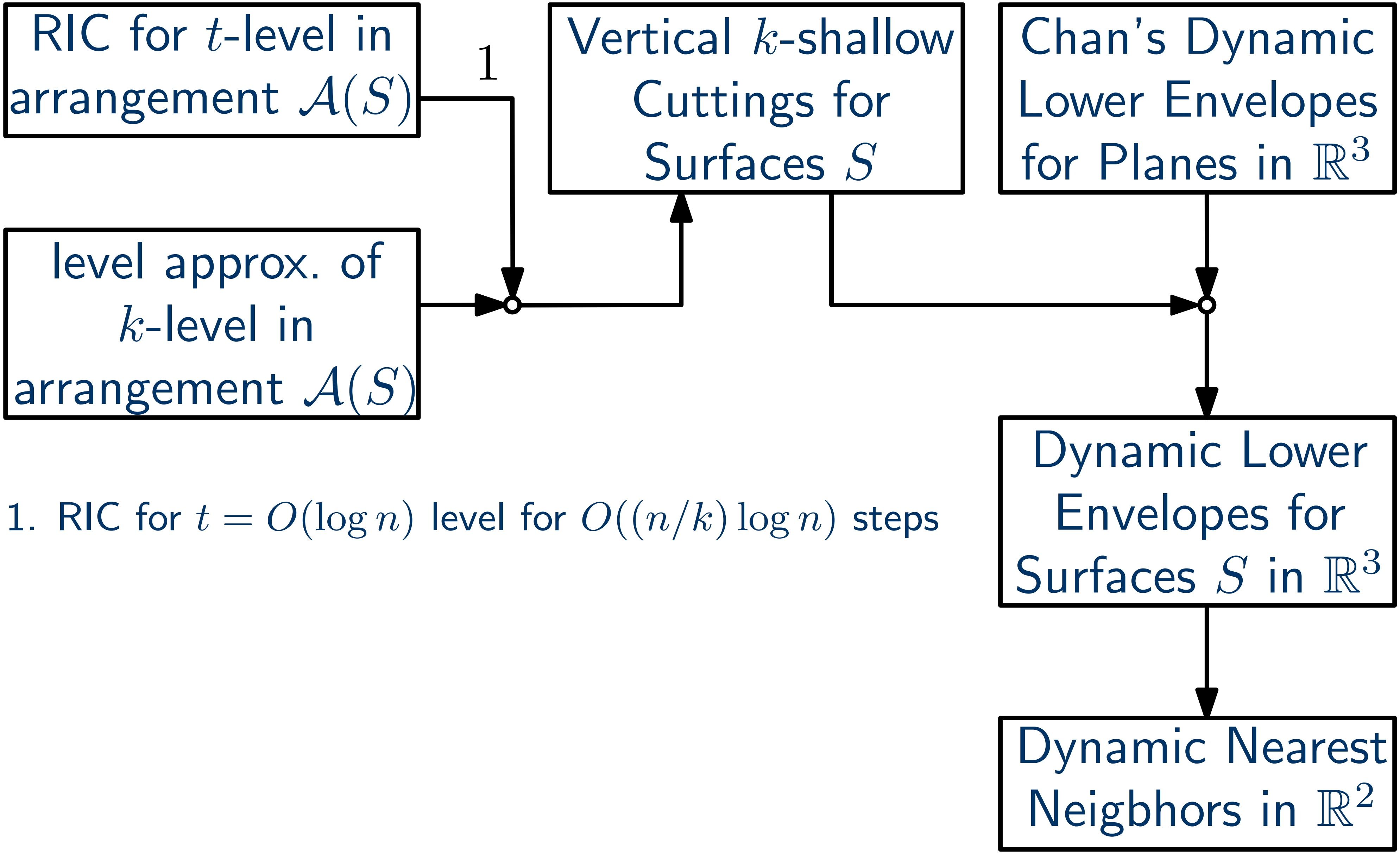


Randomized Incremental Construction

Theorem: Constructing the $\leq t$-level of $\mathcal{A}(S)$ takes $O(nt^2\text{polylog}(n))$ expected time.

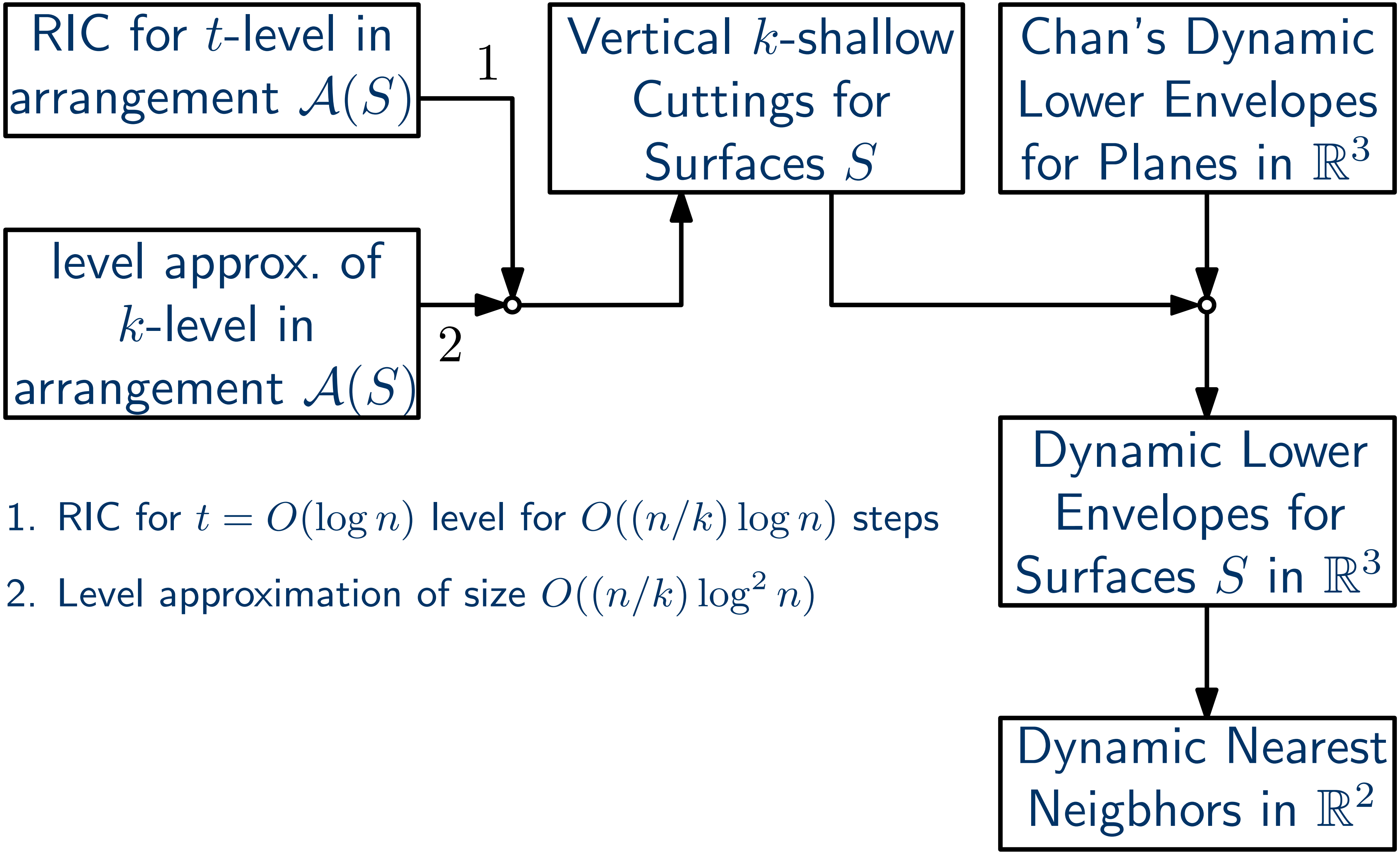


Overview



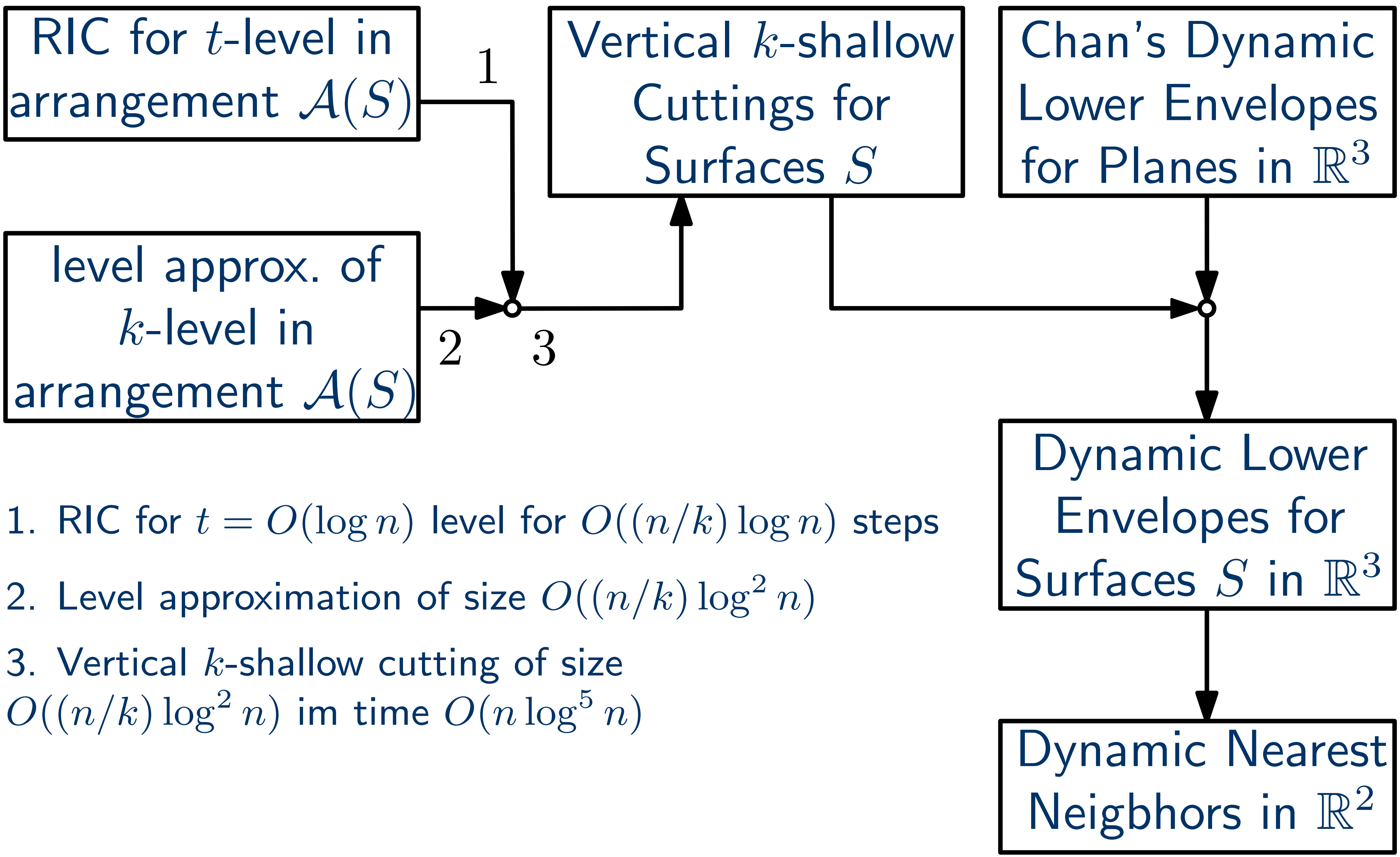
1. RIC for $t = O(\log n)$ level for $O((n/k)\log n)$ steps

Overview



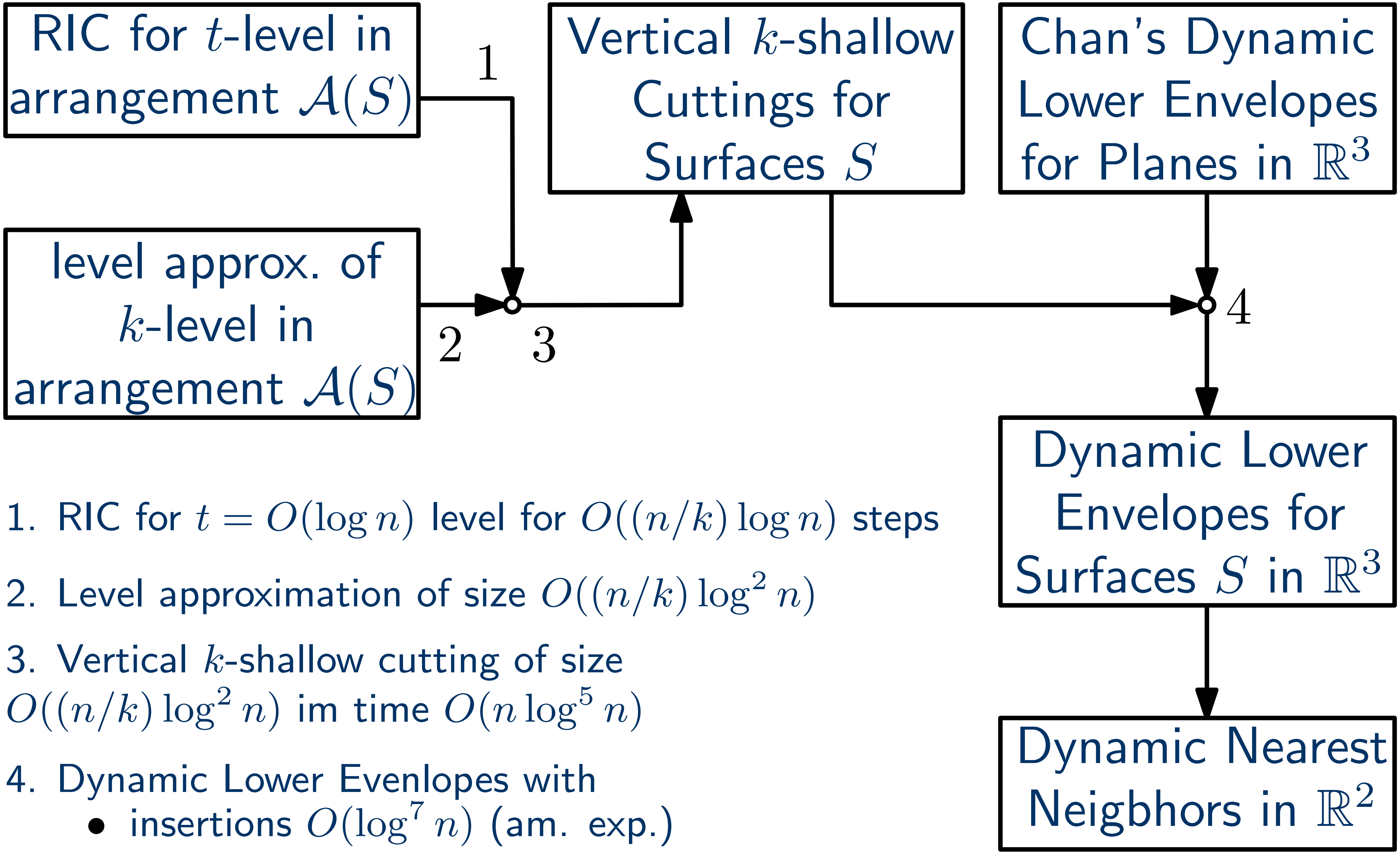
1. RIC for $t = O(\log n)$ level for $O((n/k)\log n)$ steps
2. Level approximation of size $O((n/k)\log^2 n)$

Overview



1. RIC for $t = O(\log n)$ level for $O((n/k) \log n)$ steps
2. Level approximation of size $O((n/k) \log^2 n)$
3. Vertical k-shallow cutting of size $O((n/k) \log^2 n)$ im time $O(n \log^5 n)$

Overview



1. RIC for $t = O(\log n)$ level for $O((n/k)\log n)$ steps
2. Level approximation of size $O((n/k)\log^2 n)$
3. Vertical k-shallow cutting of size $O((n/k)\log^2 n)$ im time $O(n\log^5 n)$
4. Dynamic Lower Evenlopes with
   - insertions $O(\log^7 n)$ (am. exp.)
   - deletions: $O(\log^{11} n)$ (am. exp.)
   - queries: $O(\log^2 n)$ (w.c.)