Optimal parameterized algorithms for planar facility location problems using Voronoi diagrams and sphere cut decompositions

Dániel Marx

Institute for Computer Science and Control,
Hungarian Academy of Sciences (MTA SZTAKI)
Budapest, Hungary

(Joint work with Michał Pilipczuk)

International Workshop on Graph Decomposition
CIRM, Marseille, France
Januar 19, 2015

The square root phenomenon

Most parameterized problems can be solved faster on planar graphs:

	General graphs	Planar graphs
VERTEX COVER, k-PATH, ...	$2^{O(k)} \cdot n^{O(1)}$	$2^{O(\sqrt{k})} \cdot n^{O(1)}$
INDEPENDENT SET, DOMINATING SET, ...	$n^{O(k)}$	$2^{O(\sqrt{k})} \cdot n^{O(1)}$
STRONGLY CONNECTED STEINER SUBGRAPH, ...	$n^{O(k)}$	$n^{O(\sqrt{k})}$

These running times are optimal under the Exponential Time Hypothesis ($\approx$ n-variable 3SAT cannot be solved in time $2^{o(n)}$).

The square root phenomenon

Most parameterized problems can be solved faster on planar graphs:

	General graphs	Planar graphs
Vertex Cover, k-Path, ...	$2^{O(k)} \cdot n^{O(1)}$	$2^{O(\sqrt{k})} \cdot n^{O(1)}$
Independent Set, Dominating Set, ...	$n^{O(k)}$	$2^{O(\sqrt{k})} \cdot n^{O(1)}$
Strongly Connected Steiner Subgraph, ...	$n^{O(k)}$	$n^{O(\sqrt{k})}$

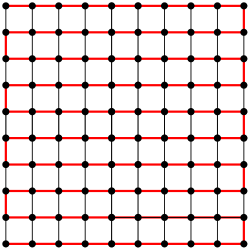
These running times are optimal under the Exponential Time Hypothesis ($\approx$ n-variable 3SAT cannot be solved in time $2^{o(n)}$).

This talk:

A general family of packing/covering problems on planar graphs and on 2D geometric objects that can be solved in time $n^{O(\sqrt{k})}$.

Bidimensionality for k-Path

Observation: If the treewidth of a planar graph G is at least $5\sqrt{k}$
$\Rightarrow$ It has a $\sqrt{k} \times \sqrt{k}$ grid minor (Planar Excluded Grid Theorem)
$\Rightarrow$ The grid has a path of length at least k.
$\Rightarrow$ G has a path of length at least k.

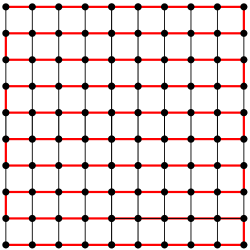


Bidimensionality for k-Path

Observation: If the treewidth of a planar graph G is at least $5\sqrt{k}$
$\Rightarrow$ It has a $\sqrt{k} \times \sqrt{k}$ grid minor (Planar Excluded Grid Theorem)
$\Rightarrow$ The grid has a path of length at least k.
$\Rightarrow$ G has a path of length at least k.

We use this observation to find a path of length at least k on planar graphs:

- Set $w := 5\sqrt{k}$.
- Find an $O(1)$-approximate tree decomposition.
    - If treewidth is at least w: we answer "there is a path of length at least k."
    - If we get a tree decomposition of width $O(w)$, then we can solve the problem in time $2^{O(w \log w)} \cdot n^{O(1)} = 2^{O(\sqrt{k} \log k)} \cdot n^{O(1)}$.

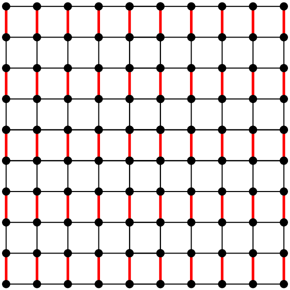


Bidimensionality

Definition

A graph invariant $x(G)$ is **minor-bidimensional** if

- $x(G') \leq x(G)$ for every minor G' of G, and
- If G_k is the $k \times k$ grid, then $x(G_k) \geq ck^2$ (for some constant $c > 0$).



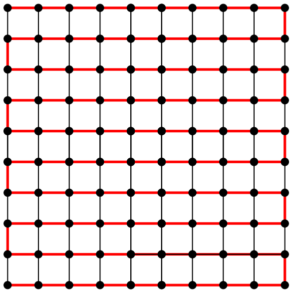
Examples: minimum vertex cover, length of the longest path, feedback vertex set are minor-bidimensional.

Bidimensionality

Definition

A graph invariant $x(G)$ is **minor-bidimensional** if

- $x(G') \leq x(G)$ for every minor G' of G, and
- If G_k is the $k \times k$ grid, then $x(G_k) \geq ck^2$ (for some constant $c > 0$).



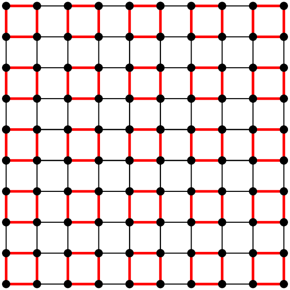
Examples: minimum vertex cover, length of the longest path, feedback vertex set are minor-bidimensional.

Bidimensionality

Definition

A graph invariant $x(G)$ is **minor-bidimensional** if

- $x(G') \leq x(G)$ for every minor G' of G, and
- If G_k is the $k \times k$ grid, then $x(G_k) \geq ck^2$ (for some constant $c > 0$).



Examples: minimum vertex cover, length of the longest path, feedback vertex set are minor-bidimensional.

Bidimensionality

Algorithms based on bidimensionality:

1. If treewidth is $\Omega(\sqrt{k})$, then we can find a $\Omega(\sqrt{k}) \times \Omega(\sqrt{k})$ grid minor.
2. The problem is trivial if there is a $\Omega(\sqrt{k}) \times \Omega(\sqrt{k})$ grid minor.
3. If treewidth is $O(\sqrt{k})$, we can solve the problem in time $2^{O(\sqrt{k})} \cdot n^{O(1)}$.

Variant of theory works for **contraction-bidimensional** problems, e.g., Independent Set, Dominating Set.

Bidimensionality

Algorithms based on bidimensionality:

1. If treewidth is $\Omega(\sqrt{k})$, then we can find a $\Omega(\sqrt{k}) \times \Omega(\sqrt{k})$ grid minor.
2. The problem is trivial if there is a $\Omega(\sqrt{k}) \times \Omega(\sqrt{k})$ grid minor.
3. If treewidth is $O(\sqrt{k})$, we can solve the problem in time $2^{O(\sqrt{k})} \cdot n^{O(1)}$.

Variant of theory works for **contraction-bidimensional** problems, e.g., Independent Set, Dominating Set.

However, for some problems, large treewidth (e.g., Multiway Cut, Subset TSP) is not of any apparent help.

General principle

Exploit the fact that some auxiliary planar graph related to the solution has size $O(k)$ and hence treewidth $O(\sqrt{k})$.

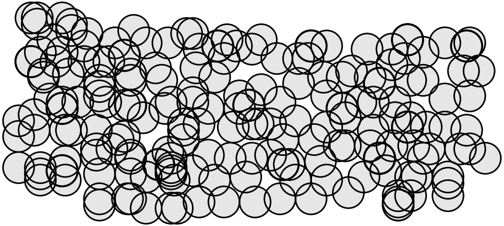
Outline

- A 2D geometric problem: Independent Set problem for unit disks.
    - A simple $n^{O(\sqrt{k})}$ algorithm using shifting.
    - A more complicated algorithm via Voronoi diagrams (idea essentially comes from recent work on QPTASs for geometric problems, e.g., [Har-Peled SOCG 2014]).
- Several generalizations/variants.
- Planar graphs.
- Some lower bounds.

Independent Set for Unit Disks

Theorem [Alber and Fiala 2004]

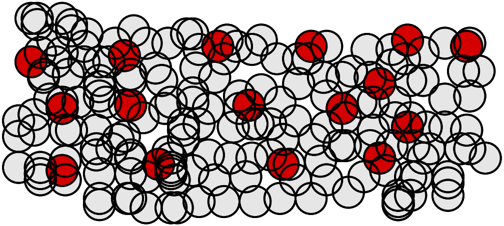
The INDEPENDENT SET problem for unit (diameter) disks can be solved in time $n^{O(\sqrt{k})}$.



Independent Set for Unit Disks

Theorem [Alber and Fiala 2004]

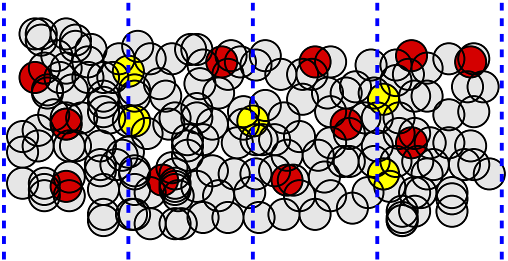
The INDEPENDENT SET problem for unit (diameter) disks can be solved in time $n^{O(\sqrt{k})}$.



Independent Set for Unit Disks

Theorem [Alber and Fiala 2004]

The INDEPENDENT SET problem for unit (diameter) disks can be solved in time $n^{O(\sqrt{k})}$.



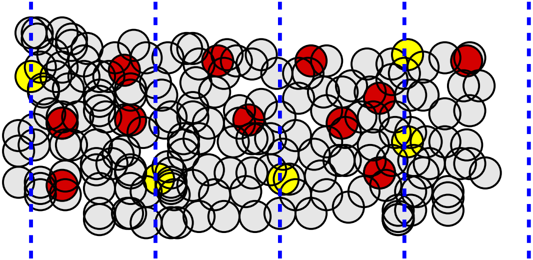
Simple solution by shifting strategy. Consider a family of vertical lines at distance $\lfloor\sqrt{k}\rfloor$ from each other, going through $(i, 0)$ for some integer $0 \leq i < \lfloor\sqrt{k}\rfloor$.

Claim: Exists i such that the lines hit $O(\sqrt{k})$ disks of the solution.

Independent Set for Unit Disks

Theorem [Alber and Fiala 2004]

The INDEPENDENT SET problem for unit (diameter) disks can be solved in time $n^{O(\sqrt{k})}$.



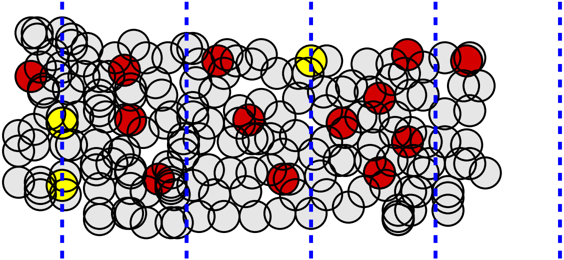
Simple solution by shifting strategy. Consider a family of vertical lines at distance $\lfloor\sqrt{k}\rfloor$ from each other, going through $(i, 0)$ for some integer $0 \leq i < \lfloor\sqrt{k}\rfloor$.

Claim: Exists i such that the lines hit $O(\sqrt{k})$ disks of the solution.

Independent Set for Unit Disks

Theorem [Alber and Fiala 2004]

The INDEPENDENT SET problem for unit (diameter) disks can be solved in time $n^{O(\sqrt{k})}$.



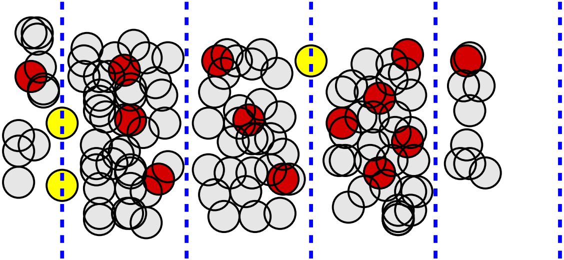
Simple solution by shifting strategy. Consider a family of vertical lines at distance $\lfloor\sqrt{k}\rfloor$ from each other, going through $(i, 0)$ for some integer $0 \le i < \lfloor\sqrt{k}\rfloor$.

Claim: Exists i such that the lines hit $O(\sqrt{k})$ disks of the solution.

Independent Set for Unit Disks

Theorem [Alber and Fiala 2004]

The Independent Set problem for unit (diameter) disks can be solved in time $n^{O(\sqrt{k})}$.



Simple solution by shifting strategy. Consider a family of vertical lines at distance $\lfloor\sqrt{k}\rfloor$ from each other, going through $(i, 0)$ for some integer $0 \le i < \lfloor\sqrt{k}\rfloor$.

Algorithm: Guess i and the $O(\sqrt{k})$ disks hit by the lines $\Rightarrow$ Remove every disk intersected by the lines or disks $\Rightarrow$ Problem falls apart into strips of height $O(\sqrt{k})$; can be solved optimally in time $n^{O(\sqrt{k})}$.

Challenges

Key idea

We were able to find a separator that hits $O(\sqrt{k})$ disks of the solution and breaks the instance in a nice way.

Two natural directions:

1. Can we solve Independent Set for disks with arbitrary radius in time $n^{O(\sqrt{k})}$?
2. Can we solve Scattered Set (find k vertices that are at distance at least d from each other) on planar graphs in time $n^{O(\sqrt{k})}$, if d is part of the input?

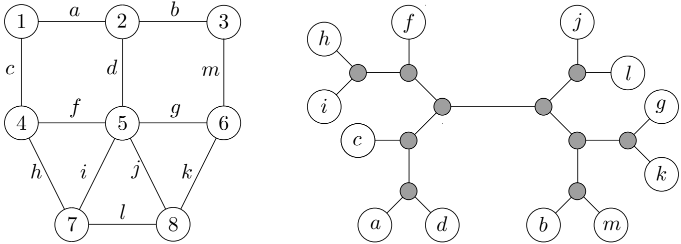
Problem:
The algorithm for unit disks crucially uses the fact that the disks have similar area.

Branch Decompositions

Definition

A **branch decomposition** of a graph $G = (V, E)$ is a tuple (T, μ) where

- T is a tree with degree 3 for all internal nodes.
- μ is a bijection between the leaves of T and $E(G)$.

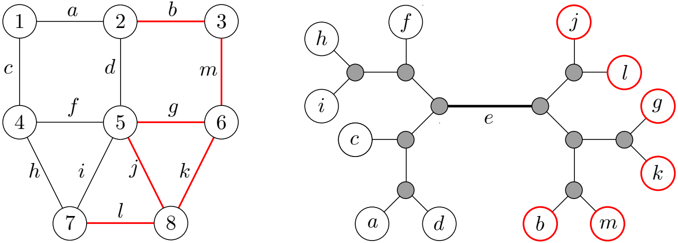


Branch Decompositions

Definition

A **branch decomposition** of a graph $G = (V, E)$ is a tuple (T, μ) where

- T is a tree with degree 3 for all internal nodes.
- μ is a bijection between the leaves of T and $E(G)$.



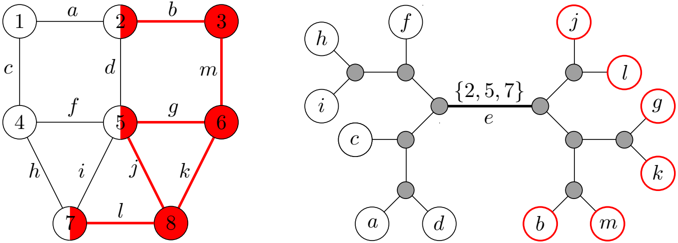
Edge $e \in T$ partitions the edge set of G into A_e and B_e

Branch Decompositions

Definition

A **branch decomposition** of a graph $G = (V, E)$ is a tuple (T, μ) where

- T is a tree with degree 3 for all internal nodes.
- μ is a bijection between the leaves of T and $E(G)$.



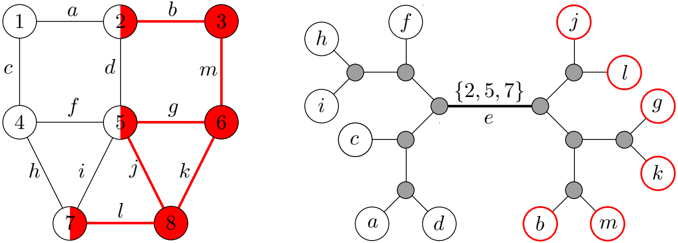
Middle set: $\mathrm{mid}(e) = V(A_e) \cap V(B_e)$

Branch Decompositions

Definition

A **branch decomposition** of a graph $G = (V, E)$ is a tuple (T, μ) where

- T is a tree with degree 3 for all internal nodes.
- μ is a bijection between the leaves of T and $E(G)$.

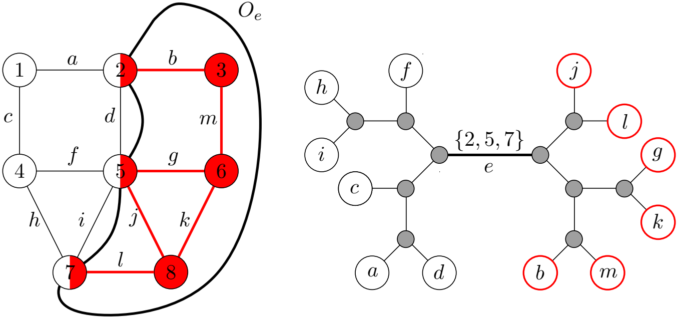


- The **width** of a branch decomposition is $\max_{e \in T} |\text{mid}(e)|$.
- The **branch width** of a graph G is the minimum width over all branch decompositions of G.

Sphere cut decomposition

Let G be a **planar** graph embedded on the sphere (or a plane) $\mathcal{S}_0$

A **sphere cut decomposition** of G is a branch decomposition (T, τ) where for every $e \in E(T)$, the vertices in $\mathrm{mid}(e)$ are the vertices in a Jordan curve of $\mathcal{S}_0$ that meets no edges of G and goes through every face at most once (a noose).



Sphere cut decompositions

Theorem [Seymour-Thomas 1994, Dorn et al. 2005]

Every 2-edge connected planar graph G of branchwidth ℓ has a sphere cut decomposition of width ℓ. This decomposition can be constructed in $O(n^3)$ steps.

Sphere cut decompositions

Theorem [Seymour-Thomas 1994, Dorn et al. 2005]

Every 2-edge connected planar graph G of branchwidth ℓ has a sphere cut decomposition of width ℓ. This decomposition can be constructed in $O(n^3)$ steps.

α-**edge-balanced noose:** at most α fraction of the edges are inside/outside the noose.

Corollary

Every connected 3-regular planar multigraph with n vertices has a $\frac{2}{3}$-edge-balanced noose of length $O(\sqrt{n})$.

Sphere cut decompositions

Theorem [Seymour-Thomas 1994, Dorn et al. 2005]

Every 2-edge connected planar graph G of branchwidth ℓ has a sphere cut decomposition of width ℓ. This decomposition can be constructed in $O(n^3)$ steps.

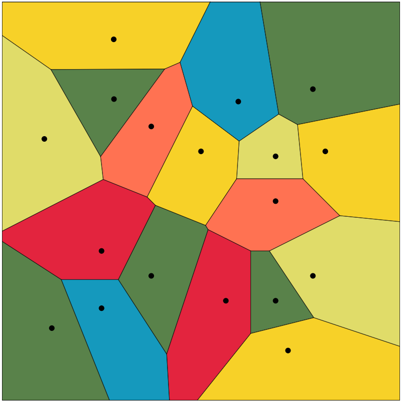
α**-face-balanced noose:** at most α fraction of the faces are strictly inside/outside the noose.

Corollary

Every connected 3-regular planar multigraph with n vertices has a $\frac{2}{3}$-face-balanced noose of length $O(\sqrt{n})$.

Voronoi diagrams

Voronoi diagram: we partition the points of the plane according to the closest center.

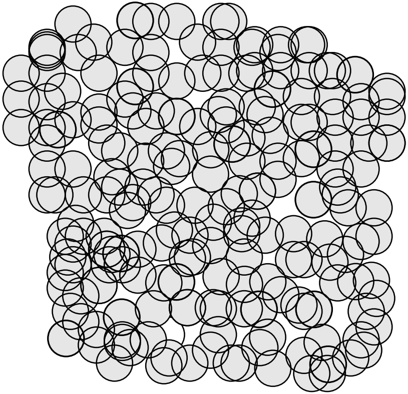


Observation: every cell is convex.

- Assume that the branch points of the diagram have degree 3.
- Ignore what happens at infinity.

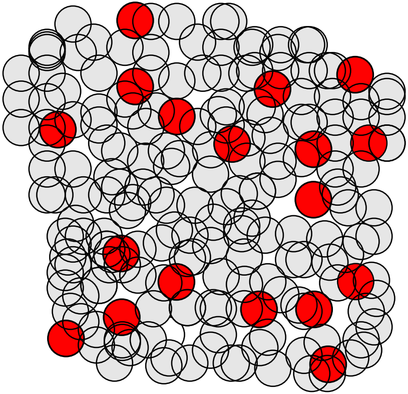
Voronoi separators

Consider the Voronoi diagram of the centers of the solution disks.



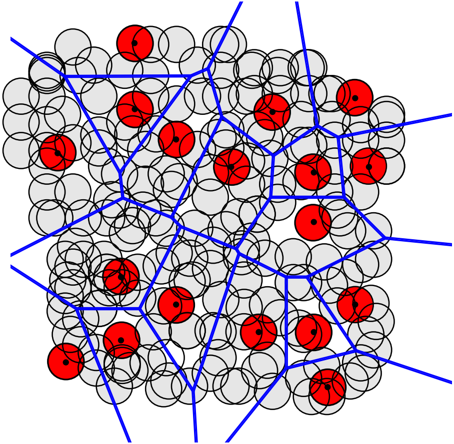
Voronoi separators

Consider the Voronoi diagram of the centers of the solution disks.



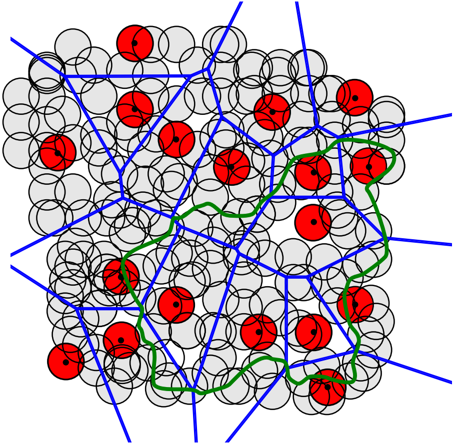
Voronoi separators

Consider the Voronoi diagram of the centers of the solution disks.



Voronoi separators

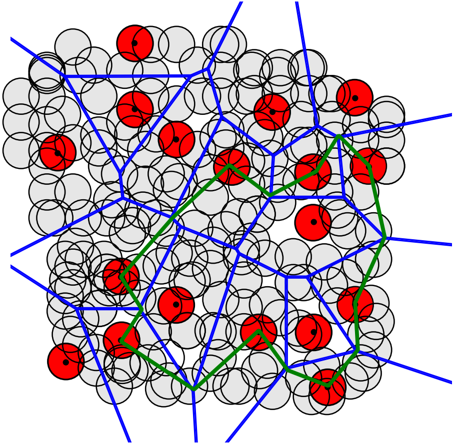
Consider the Voronoi diagram of the centers of the solution disks.



There is a $\frac{2}{3}$-face-balanced noose of length $O(\sqrt{k})$.

Voronoi separators

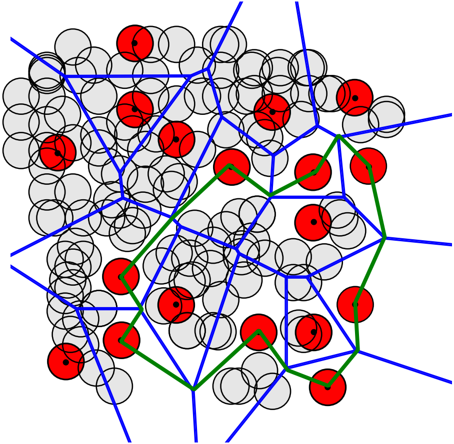
Consider the Voronoi diagram of the centers of the solution disks.



There is a $\frac{2}{3}$-face-balanced noose of length $O(\sqrt{k})$.
$\Rightarrow$ There is a corresponding polygon of length $O(\sqrt{k})$.

Voronoi separators

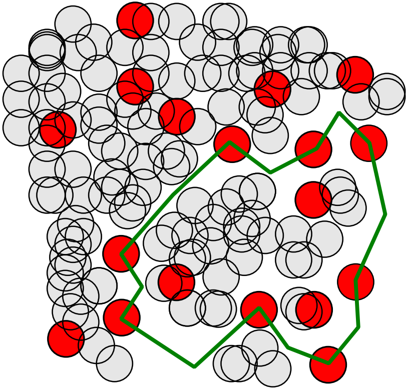
Consider the Voronoi diagram of the centers of the solution disks.



Algorithm: guess $O(\sqrt{k})$ disks and a polygon going through them, remove any disks intersecting the polygon or the guessed disks, recursion on the inside and the outside.

Voronoi separators

Consider the Voronoi diagram of the centers of the solution disks.



Algorithm: guess $O(\sqrt{k})$ disks and a polygon going through them, remove any disks intersecting the polygon or the guessed disks, recursion on the inside and the outside.

Running time

Number of candidate polygons

Number of centers: n.

Potential locations of Voronoi branch points: n^3.

$\Rightarrow$ Number of polygons of length $O(\sqrt{k})$: $n^{O(\sqrt{k})}$.

Recursion

$T(n,k)$: running time with n centers and solution size at most k.

$$T(n,k) = n^{O(\sqrt{k})} T(n, \frac{2}{3}k)$$

Running time

Number of candidate polygons

Number of centers: n.

Potential locations of Voronoi branch points: n^3.

$\Rightarrow$ Number of polygons of length $O(\sqrt{k})$: $n^{O(\sqrt{k})}$.

Recursion

$T(n,k)$: running time with n centers and solution size at most k.

$$
\begin{aligned}
T(n,k) &= n^{O(\sqrt{k})}\, T(n, \tfrac{2}{3}k) \\
&= n^{O(\sqrt{k})} \cdot n^{O(\sqrt{\frac{2}{3}k})} \cdot n^{O(\sqrt{(\frac{2}{3})^2 k})} \cdot n^{O(\sqrt{(\frac{2}{3})^3 k})} \cdots \\
&= n^{O((1+(\frac{2}{3})^{\frac{1}{2}}+(\frac{2}{3})^{\frac{2}{2}}+(\frac{2}{3})^{\frac{3}{2}}+\dots)\sqrt{k})} = n^{O(\sqrt{k})}.
\end{aligned}
$$

Running time

Number of candidate polygons

Number of centers: n.

Potential locations of Voronoi branch points: n^3.

$\Rightarrow$ Number of polygons of length $O(\sqrt{k})$: $n^{O(\sqrt{k})}$.

Recursion

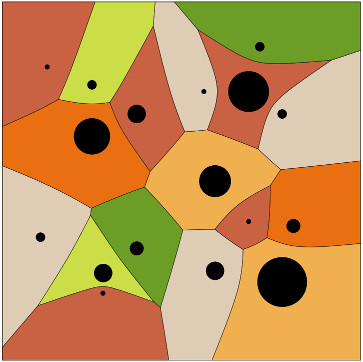
$T(n,k)$: running time with n centers and solution size at most k.

$$\begin{aligned} T(n,k) &= n^{O(\sqrt{k})} T(n, \frac{2}{3}k) \\ &= n^{O(\sqrt{k})} \cdot n^{O(\sqrt{\frac{2}{3}k})} \cdot n^{O(\sqrt{(\frac{2}{3})^2 k})} \cdot n^{O(\sqrt{(\frac{2}{3})^3 k})} \dots \\ &= n^{O((1+(\frac{2}{3})^{\frac{1}{2}}+(\frac{2}{3})^{\frac{2}{2}}+(\frac{2}{3})^{\frac{3}{2}}+\dots)\sqrt{k})} = n^{O(\sqrt{k})}. \end{aligned}$$

This gives another $n^{O(\sqrt{k})}$ time algorithm for INDEPENDENT SET for unit disks. But what about general disks?

Additively weighted Voronoi diagrams

Each center c_i has a weight w_i and we classify each point v according to $\mathrm{dist}(v, c_i) - w_i$.

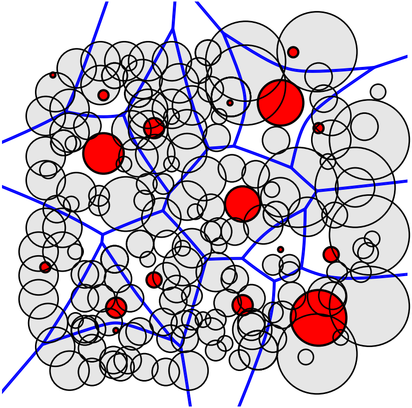


Alternative view: distance is measured from a disk of radius w_i centered at c_i.
Observation: The cells are star convex, that is, the segment between the center and a point of the cell is in the cell.

INDEPENDENT SET for general disks

Theorem

The INDEPENDENT SET problem for disks (of arbitrary radius) can be solved in time $n^{O(\sqrt{k})}$.

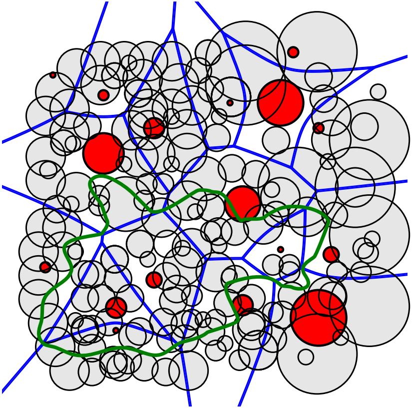


Use the additively weighted Voronoi diagram. Algorithm for unit disks go through.

Independent Set for general disks

Theorem

The Independent Set problem for disks (of arbitrary radius) can be solved in time $n^{O(\sqrt{k})}$.

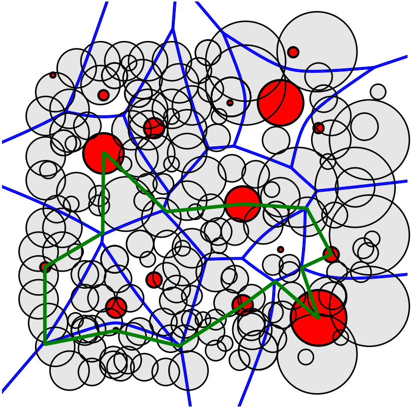


Use the additively weighted Voronoi diagram. Algorithm for unit disks go through.

Independent Set for general disks

Theorem

The Independent Set problem for disks (of arbitrary radius) can be solved in time $n^{O(\sqrt{k})}$.

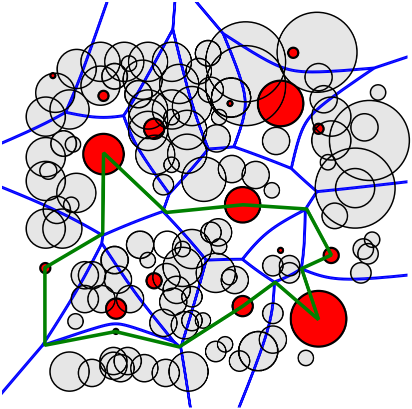


Use the additively weighted Voronoi diagram. Algorithm for unit disks go through.

Independent Set for general disks

Theorem

The Independent Set problem for disks (of arbitrary radius) can be solved in time $n^{O(\sqrt{k})}$.

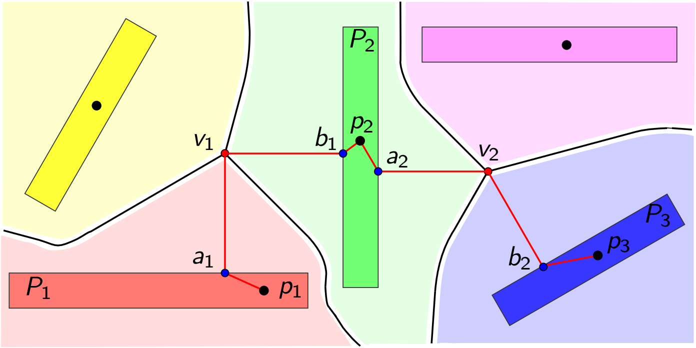


Use the additively weighted Voronoi diagram. Algorithm for unit disks go through.

Independent Set for convex polygons

We can define Voronoi diagrams for arbitrary objects.

Problem: The cells are not necessarily star convex.



Fix: the polygon is now of the form

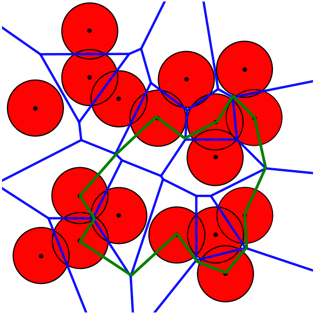
$\cdots$ — *branch point — polygon boundary — polygon center — polygon boundary — branch point* — $\cdots$.

Can be further generalized to nonconvex polygons.

Covering points with unit disks

Task: Given n unit disks and m points, select k disks covering the maximum number of points in total.

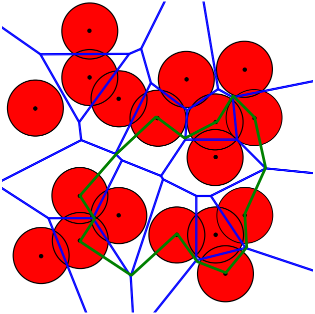
Consider again the Voronoi diagram of the solution centers.



Covering points with unit disks

Task: Given n unit disks and m points, select k disks covering the maximum number of points in total.

Consider again the Voronoi diagram of the solution centers.

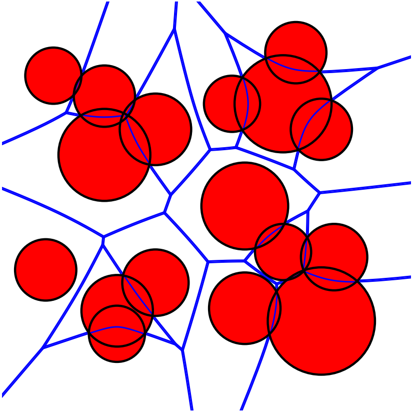


We guess $O(\sqrt{k})$ centers and a polygon.

- Remove the points covered by the selected centers.
- Remove any center that is closer to a point of the polygon than the selected centers.
- We solve recursively the inside/outside (why?)

Covering points with arbitrary disks

Task: Given n arbitrary disks and m points, select k disks covering the maximum number of points in total.



Can be solved in time $n^{O(\sqrt{k})}m^{O(1)}$ using the additively weighted Voronoi diagram.

Planar graphs

So far we have considered

- selecting k disjoint connected objects in the plane,
- selecting k disks covering the maximum number of points.

Similar algorithms can be worked out for planar graphs.

Packing problems:

Theorem

Given a set $\mathcal{D}$ of connected subgraphs in a planar graph, we can find k vertex-disjoint subgraphs in time $|\mathcal{D}|^{O(\sqrt{k})} \cdot n^{O(1)}$.

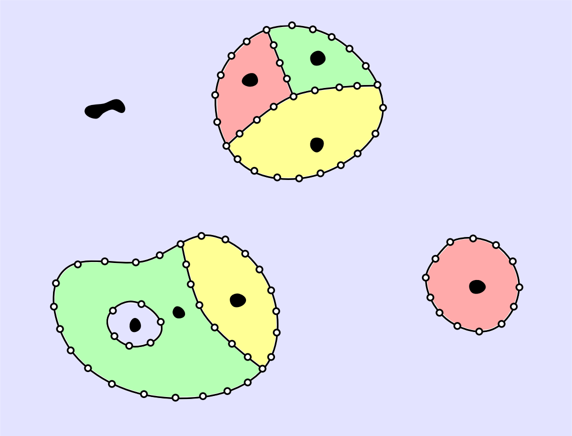
Covering problems:

Theorem

Given a set $\mathcal{D}$ of center points equipped with a radius and a set $\mathcal{C}$ of client points, we can find k center points satisfying the maximum number of client points in time $|\mathcal{D}|^{O(\sqrt{k})} \cdot n^{O(1)}$.

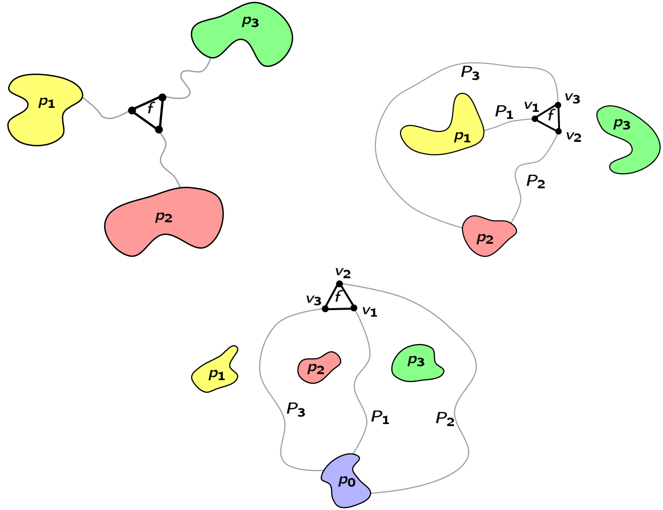
Planar graphs — challenges

The (analog of) Voronoi diagram is not necessarily connected.



Planar graphs — challenges

We need to define the analog of branch points and find candidate branch points efficiently (to have $|\mathcal{D}|^{O(\sqrt{k})}$ instead of $n^{O(\sqrt{k})}$).



The general planar graph problem

We have the following setting:

- A planar graph with weighted edges.
- Set $\mathcal{D}$ of connected subgraphs, equipped with cost and radius.
- Set $\mathcal{C}$ of client points, equipped with sensitivity and prize.
- Task: find a set of exactly k vertex-disjoint objects from $\mathcal{D}$.
- An object satisfies a client point if

$$\text{distance} \leq \text{radius} + \text{sensitivity}.$$

- We want to maximize the total prize minus the total cost.

Theorem

The general problem on planar graphs can be solved in time $|\mathcal{D}|^{O(\sqrt{k})} \cdot n^{O(1)}$ (where n is the length of the input).

Consequences

Packing problems:

Theorem

Scattered Set on planar graphs can be solved in time $n^{O(\sqrt{k})}$.

Theorem

Given a set $\mathcal{D}$ of connected subgraphs in a planar graph, we can find k vertex-disjoint subgraphs in time $|\mathcal{D}|^{O(\sqrt{k})} \cdot n^{O(1)}$.

Consequences

Packing problems:

Theorem

SCATTERED SET on planar graphs can be solved in time $n^{O(\sqrt{k})}$.

Theorem

Given a set $\mathcal{D}$ of connected subgraphs in a planar graph, we can find k vertex-disjoint subgraphs in time $|\mathcal{D}|^{O(\sqrt{k})} \cdot n^{O(1)}$.

Covering problems:

Theorem

d-DOMINATING SET on planar graphs can be solved in $n^{O(\sqrt{k})}$ (d is part of the input).

Theorem

Given a set $\mathcal{D}$ of metric balls in a planar graph, we can find k vertex-disjoint balls in time $|\mathcal{D}|^{O(\sqrt{k})} \cdot n^{O(1)}$.

Consequences

Packing problems:

Theorem

Scattered Set on planar graphs can be solved in time $n^{O(\sqrt{k})}$.

Theorem

Given a set $\mathcal{D}$ of connected subgraphs in a planar graph, we can find k vertex-disjoint subgraphs in time $|\mathcal{D}|^{O(\sqrt{k})} \cdot n^{O(1)}$.

Covering problems:

What about covering points by arbitrary objects?

Lower bounds

Strong Exponential Time Hypothesis (SETH):
No $(2-\epsilon)^n$ time algorithm for $\mathrm{CNF\text{-}SAT}$.

Theorem

Given a set $\mathcal{D}$ of convex polygons and a set $\mathcal{C}$ of points, there is no $f(k)(|\mathcal{D}|+|\mathcal{C}|)^{k-\epsilon}$ time algorithm for the problem of covering $\mathcal{C}$ with k polygons from $\mathcal{D}$ (unless SETH fails).

Lower bounds

Strong Exponential Time Hypothesis (SETH):
No $(2-\epsilon)^n$ time algorithm for CNF-SAT.

Theorem

Given a set $\mathcal{D}$ of convex polygons and a set $\mathcal{C}$ of points, there is no $f(k)(|\mathcal{D}|+|\mathcal{C}|)^{k-\epsilon}$ time algorithm for the problem of covering $\mathcal{C}$ with k polygons from $\mathcal{D}$ (unless SETH fails).

We use the following result:

Theorem [Pătrașcu and Williams 2010]

There is no $f(k)n^{k-\epsilon}$ time algorithm for Dominating Set (unless SETH fails).

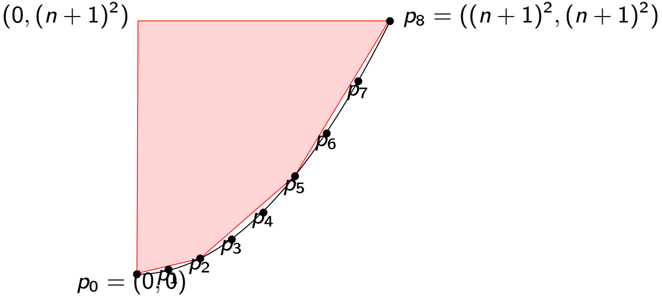
Lower bounds

Strong Exponential Time Hypothesis (SETH):
No $(2-\epsilon)^n$ time algorithm for CNF-SAT.

Theorem

Given a set $\mathcal{D}$ of convex polygons and a set $\mathcal{C}$ of points, there is no $f(k)(|\mathcal{D}|+|\mathcal{C}|)^{k-\epsilon}$ time algorithm for the problem of covering $\mathcal{C}$ with k polygons from $\mathcal{D}$ (unless SETH fails).

Transparent reduction from DOMINATING SET:



Lower bounds

Exponential Time Hypothesis (ETH):
No $2^{o(n)}$ time algorithm for n-variable $\mathrm{3SAT}$.

Theorem

Given a set $\mathcal{D}$ of axis-parallel rectangles and a set $\mathcal{C}$ of points, there is no $f(k)(|\mathcal{D}| + |\mathcal{C}|)^{o(k)}$ time algorithm for the problem of covering $\mathcal{C}$ with k rectangles from $\mathcal{D}$ (unless ETH fails).

Lower bounds

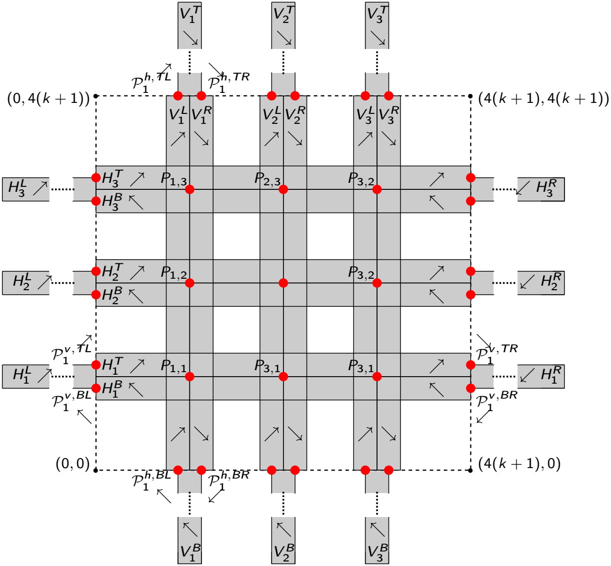
Exponential Time Hypothesis (ETH):
No $2^{o(n)}$ time algorithm for n-variable $\mathrm{3SAT}$.

Theorem

Given a set $\mathcal{D}$ of axis-parallel rectangles and a set $\mathcal{C}$ of points, there is no $f(k)(|\mathcal{D}| + |\mathcal{C}|)^{o(k)}$ time algorithm for the problem of covering $\mathcal{C}$ with k rectangles from $\mathcal{D}$ (unless ETH fails).

- Remains true if every rectangle is either $1 \times k$ or $k \times 1$.
- Slightly weaker bound $f(k)(|\mathcal{D}| + |\mathcal{C}|)^{o(k/\log k)}$ remains true if every rectangle has width and height in the range $[1, 1+\epsilon]$.

Lower bounds



Summary

- Main result: $n^{O(\sqrt{k})}$ time algorithms for natural packing/covering problems.
- Main technical tool: find balanced separators in the Voronoi diagram of the solution (comes from recent results on QPTASs for geometric problems, e.g., [Har-Peled SOCG 2014]).
- Lower bounds show that the square root phenomenon does not hold for certain natural covering problems.
- It seems essential that the covering problems are defined in a metric way.