Geometric Independet Set Problem

Abraham Hinteregger

Vienna University of Technology

13.4.2016

Chapter

Intersection graph

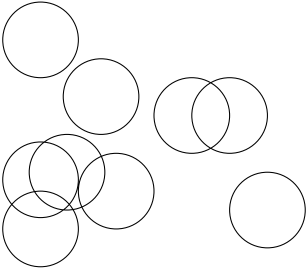
- Introduction
- Independent set of IG
- PTAS

M(W)IS in Unit Disk Graphs

MIS for Rectangles

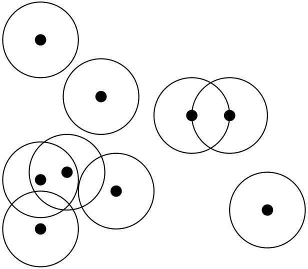
Intersection graph

- Given an arrangement A of geometric objects



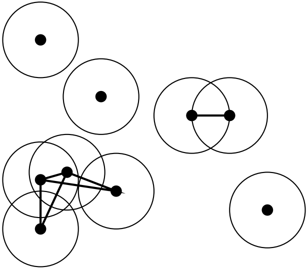
Intersection graph

- Given an arrangement A of geometric objects
- The intersection graph G has a vertice v_i for every object $O_i \in A$



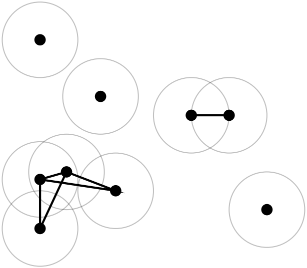
Intersection graph

- Given an arrangement A of geometric objects
- The intersection graph G has a vertice v_i for every object $O_i \in A$
- If two objects intersect there is an edge between them in G



Intersection graph

- Given an arrangement A of geometric objects
- The intersection graph G has a vertice v_i for every object $O_i \in A$
- If two objects intersect there is an edge between them in G



Intersection graph

- Can be defined for arbitrary geometric objects (example on last slide was a unit disk graph - UDG)
- Given the arrangement of objects can be calculated in polynomial time $O(n^2)$ if complexity of objects is bounded

Intersection graph

- Can be defined for arbitrary geometric objects (example on last slide was a unit disk graph - UDG)
- Given the arrangement of objects can be calculated in polynomial time $O(n^2)$ if complexity of objects is bounded
- Some subsets of IG and possible applications
  - Clique (K_n subgraphs of G) of interval graph (IG for intervals $i \in \mathbb{R}^2$) can be used for scheduling (similar to doodle)

Intersection graph

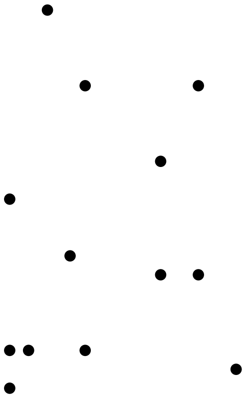
- Can be defined for arbitrary geometric objects (example on last slide was a unit disk graph - UDG)
- Given the arrangement of objects can be calculated in polynomial time $O(n^2)$ if complexity of objects is bounded
- Some subsets of IG and possible applications
  - Clique (K_n subgraphs of G) of interval graph (IG for intervals $i \in \mathbb{R}^2$) can be used for scheduling (similar to doodle)
  - Path between two vertices on IG of geometric shapes corresponds to a path between two points without leaving perimeter of shapes (e.g. cheapest way from A to B with fee on borders)

Intersection graph

- Can be defined for arbitrary geometric objects (example on last slide was a unit disk graph - UDG)
- Given the arrangement of objects can be calculated in polynomial time $O(n^2)$ if complexity of objects is bounded
- Some subsets of IG and possible applications
  - Clique (K_n subgraphs of G) of interval graph (IG for intervals $i \in \mathbb{R}^2$) can be used for scheduling (similar to doodle)
  - Path between two vertices on IG of geometric shapes corresponds to a path between two points without leaving perimeter of shapes (e.g. cheapest way from A to B with fee on borders)
  - Independent set (non-adjacent vertices) of IG of geometric shapes corresponds to non-intersecting subset of the shapes in the arrangement

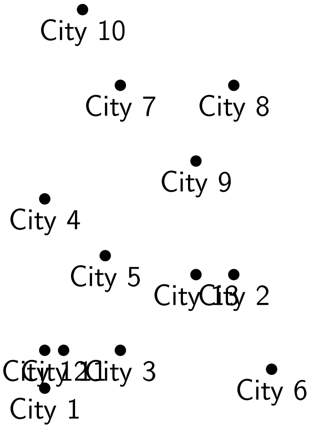
Map labelling

- Given a 2D map with various points of interest – which labels should be drawn?



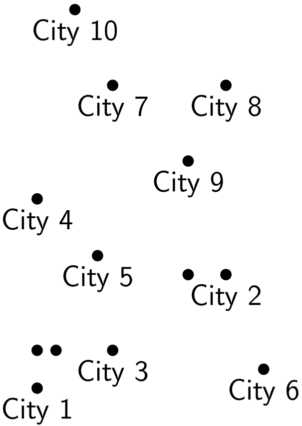
Map labelling

- Given a 2D map with various points of interest – which labels should be drawn?



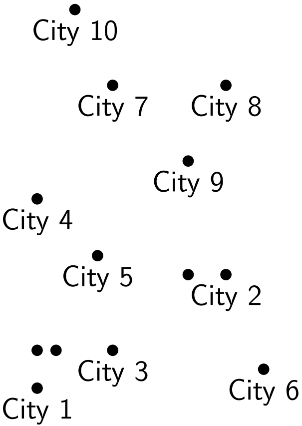
Map labelling

- Given a 2D map with various points of interest – which labels should be drawn?
- Labels must not intersect each other



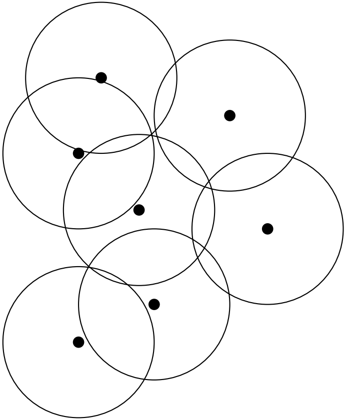
Map labelling

- Given a 2D map with various points of interest – which labels should be drawn?
- Labels must not intersect each other
- Variants of the problem:
  - Labels with size-constraints (e.g.: uniform height)
  - Labels are allowed in a radius around corresponding datapoint



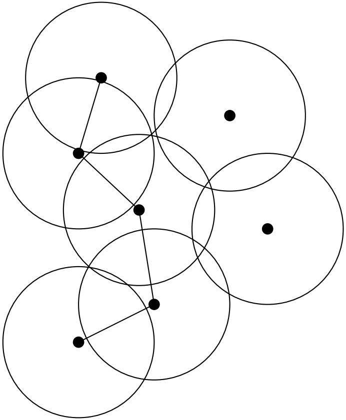
Construction of wireless network [Chamaret et al., 1997]

- Maximize area with reception



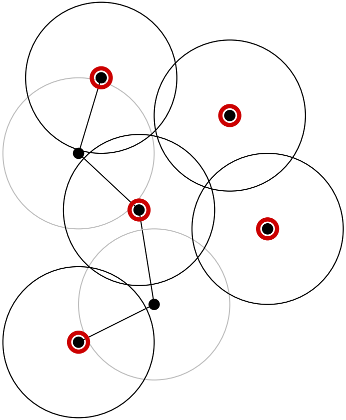
Construction of wireless network [Chamaret et al., 1997]

- Maximize area with reception
- Building base stations may be expensive $\rightarrow$ overlap is inefficient.



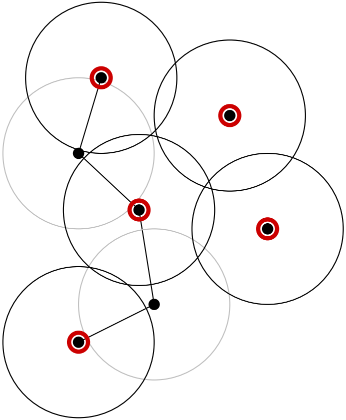
Construction of wireless network [Chamaret et al., 1997]

- Maximize area with reception
- Building base stations may be expensive → overlap is inefficient.



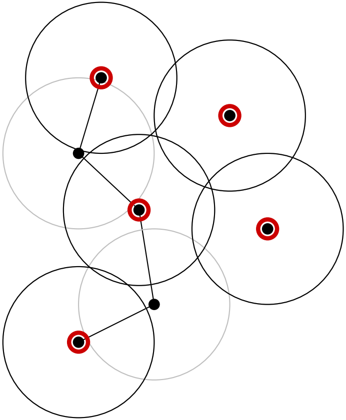
Construction of wireless network [Chamaret et al., 1997]

- Maximize area with reception
- Building base stations may be expensive $\rightarrow$ overlap is inefficient.



Construction of wireless network [Chamaret et al., 1997]

- Maximize area with reception
- Building base stations may be expensive → overlap is inefficient.
- Can also be used for assigning frequencies
  - More or less a graph coloring problem
  - Smallest number of necessary frequencies (≙ colors) is the chromatic number [McDiarmid and Müller, 2011] which is related to IS.



Polynomial time approximation scheme

- Finding IS of graph $\mathcal{NP}$- hard — therefore finding approximative solutions in polynomial time would already be nice.
- Some random estimate rather uninteresting—approximation should have some quality estimate.

Polynomial time ρ – approximation scheme

Finds solution with a solution quality that is at least $\frac{S_{OPT}}{\rho}$ in polynomial time (for fixed $\rho > 1$)

Chapter

Intersection graph

M(W)IS in Unit Disk Graphs

- Problem description
- Unweighted MIS
- Weighted MIS

MIS for Rectangles

PTAS for M(W)IS problem in UDG [Nieberg et al., 2004]

- Geometric objects are only disks with radius 1
- Two variants of the problem – with or without geometric representation
- If geometric representation is known seperation alongside some grid is possible which allows more efficient approaches (shifting strategy [Fonseca et al., 2014])
- Finding geometric representation from intersection graph is $\mathcal{NP}$- hard [Hlineny and Kratochvil, 2001]

Unweighted MIS of UDG

Given: Some graph $G = (E, V)$ which is a UDG iff there is a mapping $f : V \to \mathbb{R}^2$ s.t.:

$$(u, v) \in E \leftrightarrow ||f(u) - f(v)|| \leq 2 \qquad \forall u, v \in V, (u \neq v)$$

Desired: Subset $I \subset V$ s.t. $|I| \cdot \rho \geq \alpha(G)$
$\alpha(G)$ . . . size maximum IS

Unweighted MIS of UDG

Given: Some graph $G = (E, V)$ which is a UDG iff there is a mapping $f : V \rightarrow \mathbb{R}^2$ s.t.:

$$(u, v) \in E \leftrightarrow ||f(u) - f(v)|| \leq 2 \qquad \forall u, v \in V, (u \neq v)$$

Desired: Subset $I \subset V$ s.t. $|I| \cdot \rho \geq \alpha(G)$
$\alpha(G)$ ... size maximum IS

- Idea of algorithm
  1. Take subsets of graph with bounded size
  2. Calculate MIS of subsets
  3. Combine MIS of subsets to get IS of whole graph

Step 1: Subsets of finite size

Define the following sets for some arbitary node $v_0 \in V$

$$N_r := \{w | w \in V, w \text{ has distance at most } r \text{ from } v_0\}$$

Step 1: Subsets of finite size

Define the following sets for some arbitary node $v_0 \in V$

$$N_r := \{w | w \in V, w \text{ has distance at most } r \text{ from } v_0\}$$

and calculate MIS $I_r \subset N_r$ for $r = 0, 1, \dots \bar{r}$ where $\bar{r}$ is defined as the smallest r s.t.

$$I_{r+1} > \rho |I_r|$$

does not hold.

Bounding size of $I_{\bar{r}}$

$$\forall w \in N_r : \quad ||f(v_0) - f(w)|| \leq 2r$$

therefore it's possible to draw a circle with radius $R = 2r + 1$ and centerpoint v_0 that contains all disks in I_r and

$$|I_r| \leq \pi R^2 / \pi = O(r^2).$$

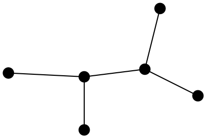
By definition of $\bar{r}$

$$|I_r| > \rho |I_{r-1}| > \ldots > \rho^r |I_0| = \rho^r$$

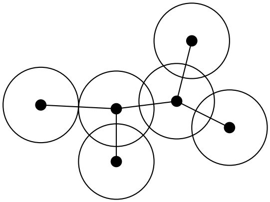
also holds. Combining these two results:

$$\rho^r < |I_r| \leq O(r^2)$$

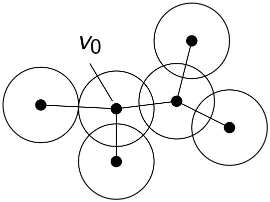
$\implies$ constant (depending only on ρ) bound on $\bar{r}$



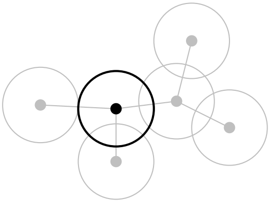
UDG



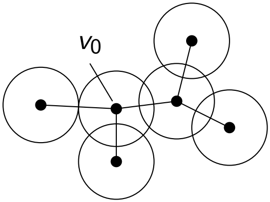
UD arrangement & UDG



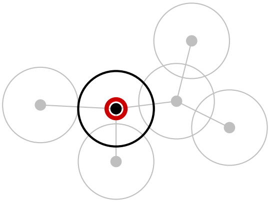
UD arrangement & UDG



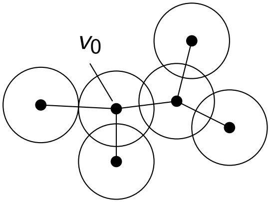
N_0



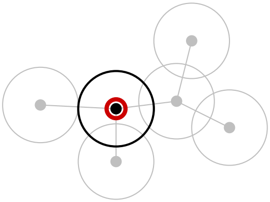
UD arrangement & UDG



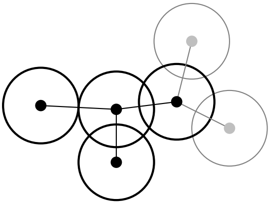
N_0 (black) and I_0 (red)



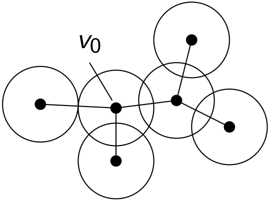
UD arrangement & UDG



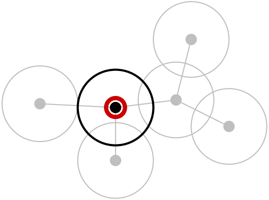
N_0 (black) and I_0 (red)



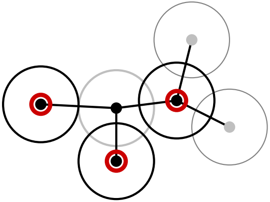
N_1



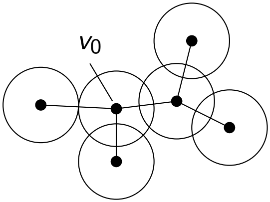
UD arrangement & UDG



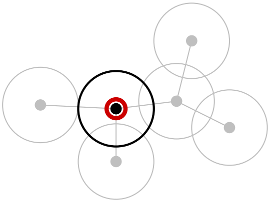
N_0 (black) and I_0 (red)



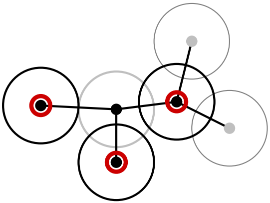
N_1 and I_1



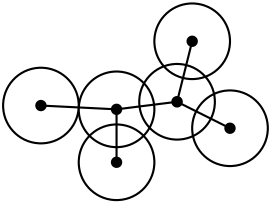
UD arrangement & UDG



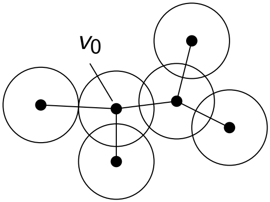
N_0 (black) and I_0 (red)



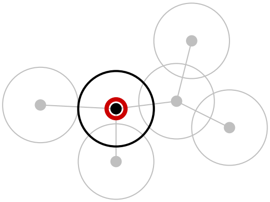
N_1 and I_1



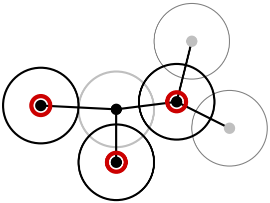
N_2



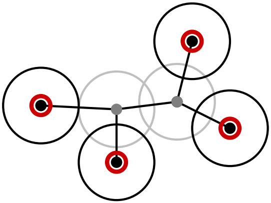
UD arrangement & UDG



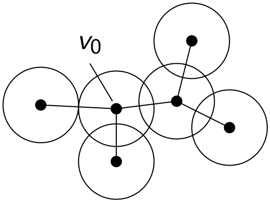
N_0 (black) and I_0 (red)



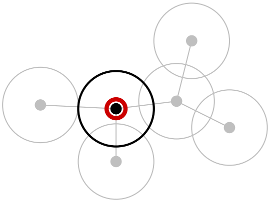
N_1 and I_1



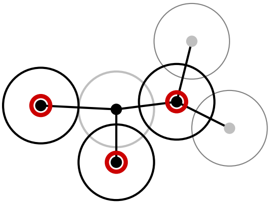
N_2 and I_2



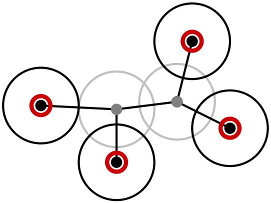
UD arrangement & UDG



N_0 (black) and I_0 (red)



N_1 and I_1



N_2 and I_2, $\bar{r} = 1$

From the definition of N_r it follows that for the subgraph $G' = G[N_{\bar{r}+1}]$ the maximum independet set size is bounded:

$$\alpha(G') \leq \rho|I_{\bar{r}}|.$$

$H = G \setminus G'$ has no vertices adjacent to vertices in $N_{\bar{r}}$ and therefore a ρ-approximate IS for H (I_H) combined with $I_{\bar{r}}$ yields a ρ-approximate IS for G.

$$\alpha(G) \leq \alpha(H) + \alpha(G') \leq \rho|I_H \cup I_{\bar{r}}|$$

Algorithm

1. Define sets N_r for arbitary node $v \in V$ and calculate independent sets I_r until stopping criterion reached
   - Remember that r has constant bound $\implies$ determining I_r is possible in $O(n^{C^2})$.
2. Repeat step 1. for subgraph $H = G \setminus N_{\bar{r}+1}$ until $H = \varnothing$.
3. Output independent set $I = \bigcup I_{\bar{r}}$ where $I_{\bar{r}}$ are alle the IS found in step 1.

Algorithm for maximum weighted independet set problem

When defining sets N_r start with v such that

$$\omega(v) \geq \omega(v') \forall v' \in V \qquad (= \text{argmax}\, \omega(v))$$

and use

$$\omega(I_{r+1}) > \rho\omega(I_r)$$

as stopping criterion (for $\bar{r}$) to get bound on the weight of the IS of the subsets.

Chapter

Intersection graph

M(W)IS in Unit Disk Graphs

MIS for Rectangles

- MIS for arbitrary rectangles
- MIS for unit height rectangles
- Take away points

MIS for arbitrary rectangles [Agarwal et al., 1998]

- Geometric objects are rectangles in $\mathbb{R}^2$ with parallel edges
- Desired: Biggest subset of those rectangles that don't intersect each other
- $\log n$ – approximate algorithm for this case with $O(n \log n)$ runtime:
  - Divide and concquer: Split into subsets and calculate independet sets for them
  - Combine IS of subsets.

Maximum independet set for arbitrary rectangles

Given: Set S of rectangles.

- Sort edges by x and y coordinates — $O(n \log n)$

Maximum independet set for arbitrary rectangles

Given: Set S of rectangles.

- Sort edges by x and y coordinates — $O(n \log n)$
- 1. Find median in x coordinate — $O(n)$

Maximum independet set for arbitrary rectangles

Given: Set S of rectangles.

- Sort edges by x and y coordinates — $O(n \log n)$
- 1. Find median in x coordinate — $O(n)$
  2. Divide rectangles in 3 sets: S_M (intersecting median), S_L and S_R (left and right of median)

Maximum independet set for arbitrary rectangles

Given: Set S of rectangles.

- Sort edges by x and y coordinates — $O(n \log n)$
- 1. Find median in x coordinate — $O(n)$
  2. Divide rectangles in 3 sets: S_M (intersecting median), S_L and S_R (left and right of median)
  3. Apply algorithm 1-4 recursively to S_L and S_R if their size is ≥ 2 to get approximate solution, else calculate MIS in $O(1)$.

Maximum independet set for arbitrary rectangles

Given: Set S of rectangles.

- Sort edges by x and y coordinates — $O(n \log n)$
- 1. Find median in x coordinate — $O(n)$
  2. Divide rectangles in 3 sets: S_M (intersecting median), S_L and S_R (left and right of median)
  3. Apply algorithm 1-4 recursively to S_L and S_R if their size is ≥ 2 to get approximate solution, else calculate MIS in $O(1)$.
  4. Calculate MIS of S_M with greedy approach (outlined on blackboard) — $O(n)$

Maximum independet set for arbitrary rectangles

Given: Set S of rectangles.

- Sort edges by x and y coordinates — $O(n \log n)$
- 1. Find median in x coordinate — $O(n)$
  2. Divide rectangles in 3 sets: S_M (intersecting median), S_L and S_R (left and right of median)
  3. Apply algorithm 1-4 recursively to S_L and S_R if their size is ≥ 2 to get approximate solution, else calculate MIS in $O(1)$.
  4. Calculate MIS of S_M with greedy approach (outlined on blackboard) — $O(n)$
  5. Take either $I_L \cup I_R$ or I_M

Maximum independet set for arbitrary rectangles

Given: Set S of rectangles.

- Sort edges by x and y coordinates — $O(n \log n)$
- 1. Find median in x coordinate — $O(n)$
  2. Divide rectangles in 3 sets: S_M (intersecting median), S_L and S_R (left and right of median)
  3. Apply algorithm 1-4 recursively to S_L and S_R if their size is ≥ 2 to get approximate solution, else calculate MIS in $O(1)$.
  4. Calculate MIS of S_M with greedy approach (outlined on blackboard) — $O(n)$
  5. Take either $I_L \cup I_R$ or I_M

Overall runtime: $O(n \log n)$

MIS for unit height rectangles [Agarwal et al., 1998]

- Map labels often have similar/identical height

MIS for unit height rectangles [Agarwal et al., 1998]

- Map labels often have similar/identical height
- Use height to split problem in small subproblems that don't depend on each other

MIS for unit height rectangles [Agarwal et al., 1998]

- Map labels often have similar/identical height
- Use height to split problem in small subproblems that don't depend on each other
- Apply greedy scheme from before for subproblems

MIS for unit height rectangles (UHR)

Draw $m+1$ horizontal lines that fulfill the following criteria:

- y - distance > 1
- Each line intersects ≥ 1 rectangles
- Each rectangle is intersected by 1 line.

MIS for unit height rectangles (UHR)

Draw $m + 1$ horizontal lines that fulfill the following criteria:

- y - distance > 1
- Each line intersects ≥ 1 rectangles
- Each rectangle is intersected by 1 line.

Solve MIS for subsets S_0 to S_m (one for every line) with greedy algorithm.

MIS for unit height rectangles (UHR)

Draw $m+1$ horizontal lines that fulfill the following criteria:

- y - distance > 1
- Each line intersects ≥ 1 rectangles
- Each rectangle is intersected by 1 line.

Solve MIS for subsets S_0 to S_m (one for every line) with greedy algorithm.

Take union of either odd or even subsets

Take away points

For difficult problems use all available information about objects at hand to make problem solvable (or at least get good approximate solutions):

- 1st algorithm (UDG): minimum area of a independet set of geometric shapes used to get an upper bound on size of subsets
- 2nd algorithm (greedy heuristic): reduce dimensionality of problem by line that intersects all rectangles
- 3rd algorithm (UHR): decompose problem into subproblems that can be solved easily

References I

Agarwal, P. K., van Kreveld, M., and Suri, S. (1998).
Label placement by maximum independent set in rectangles.
Computational Geometry, 11(3–4):209–218.

Chamaret, B., Josselin, S., Kuonen, P., Pizarroso, M., Salas-Manzanedo, B., Ubeda, S., and Wagner, D. (1997).
Radio network optimization with maximum independent set search.
In *Vehicular Technology Conference, 1997, IEEE 47th*, volume 2, pages 770–774 vol.2.

References II

- Fonseca, G. D. d., Sá, V. G. P. d., and Figueiredo, C. M. H. d. (2014).
  Linear-Time Approximation Algorithms for Unit Disk Graphs.
  In Bampis, E. and Svensson, O., editors, *Approximation and Online Algorithms*, number 8952 in Lecture Notes in Computer Science, pages 132–143. Springer International Publishing.
  DOI: 10.1007/978-3-319-18263-6_12.

- Hlineny, P. and Kratochvil, J. (2001).
  Representing graphs by disks and balls (a survey of recognition-complexity results).
  Discrete Mathematics, 229(1–3):101–124.

References III

McDiarmid, C. and Müller, T. (2011).
On the chromatic number of random geometric graphs.
Combinatorica, 31(4):423–488.
arXiv: 1101.6065.

Nieberg, T., Hurink, J., and Kern, W. (2004).
A Robust PTAS for Maximum Weight Independent Sets in Unit Disk Graphs.
In Hromkovič, J., Nagl, M., and Westfechtel, B., editors, *Graph-Theoretic Concepts in Computer Science*, number 3353 in Lecture Notes in Computer Science, pages 214–221.
Springer Berlin Heidelberg.
DOI: 10.1007/978-3-540-30559-0_18.

Remarks I

Robustness of PTAS for M(W)IS on UDG

Remarks I

Robustness of PTAS for M(W)IS on UDG

- Polynomial runtime only possible on UDG

Remarks I

Robustness of PTAS for M(W)IS on UDG

- Polynomial runtime only possible on UDG as it assumes bounded area of IS with given size
- UDG are subset of all graphs — proving that graph is UDG is $\mathcal{NP}$- hard

Remarks I

Robustness of PTAS for M(W)IS on UDG

- Polynomial runtime only possible on UDG as it assumes bounded area of IS with given size
- UDG are subset of all graphs — proving that graph is UDG is $\mathcal{NP}$- hard
- Robustness would mean: Determine in polynomial runtime whether runtime will be polynomial.

Remarks I

Robustness of PTAS for M(W)IS on UDG

- Polynomial runtime only possible on UDG as it assumes bounded area of IS with given size
- UDG are subset of all graphs — proving that graph is UDG is $\mathcal{NP}$- hard
- Robustness would mean: Determine in polynomial runtime whether runtime will be polynomial. Is this possible?
- Actually possible in this case: If $|I_r > (2r+1)^2$ is found $\implies$ graph is not UDG.

Remarks II

Improving PTAS for unit height rectangles to $(1+\varepsilon)$–approximative scheme:

- Algorithm described solves n independet subproblems optimally and discards results of $\frac{n}{2}$

Remarks II

Improving PTAS for unit height rectangles to $(1+\varepsilon)$–approximative scheme:

- Algorithm described solves n independet subproblems optimally and discards results of $\frac{n}{2}$
- Discard solutions of subproblems that are boundaries of used solutions

Remarks II

Improving PTAS for unit height rectangles to $(1+\varepsilon)$–approximative scheme:

- Algorithm described solves n independet subproblems optimally and discards results of $\frac{n}{2}$
- Discard solutions of subproblems that are boundaries of used solutions
- Could be improved by solving bigger subproblems with less discarded "boundary-problems" in between

Remarks II

Improving PTAS for unit height rectangles to $(1+\varepsilon)$–approximative scheme:

- Algorithm described solves n independet subproblems optimally and discards results of $\frac{n}{2}$
- Discard solutions of subproblems that are boundaries of used solutions
- Could be improved by solving bigger subproblems with less discarded “boundary-problems” in between
- Solving subproblems of k lines each only every $\frac{1}{k}$-th subproblem has to be discarded $\rightarrow \left(1+\frac{1}{k}\right)-$ approximation.