Geometric Approximation Algorithms

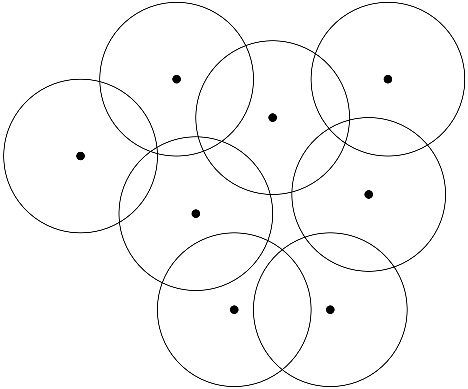
Aritra Banik[1]

Assistant Professor
National Institute of Science Education and Research

[1]Slide ideas borrowed from Subhas C. Nandy

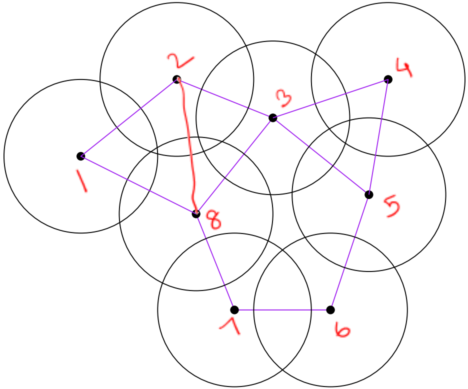
Definition

A graph that can be represented as the intersection graph of a set of circles of same radius is called the *unit disk graph*(UDG). That is, it is a graph with one vertex for each disk in the family, and with an edge between two vertices whenever the corresponding vertices lie within a unit distance of each other.



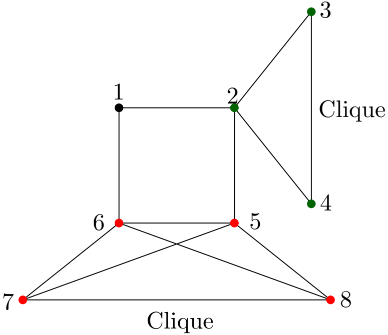
Definition

A graph that can be represented as the intersection graph of a set of circles of same radius is called the *unit disk graph*(UDG). That is, it is a graph with one vertex for each disk in the family, and with an edge between two vertices whenever the corresponding vertices lie within a unit distance of each other.

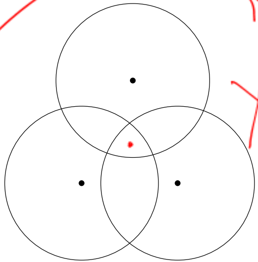


Definition

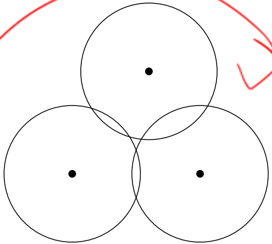
A clique is a subset of vertices of an undirected graph such that every two distinct vertices in the clique are adjacent; that is, its induced subgraph is complete.



- Geometric Clique: If a set of disks has a nonempty intersection then they form a geometric clique.



Geometric Clique

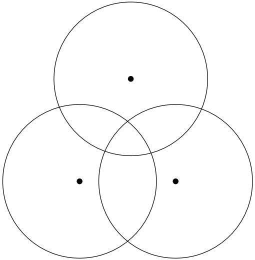


Graphical Clique

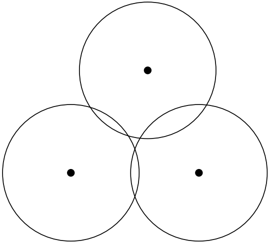
- Question1: Given a UDG find a find the maximum Graphical Clique
- Question2: Given a UDG find a find the maximum Geometric Clique

Cliques in Unit Disk Graph(UDG)

- Geometric Clique: If a set of disks has a nonempty intersection then they form a geometric clique.

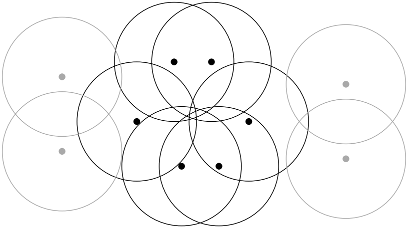


Geometric Clique



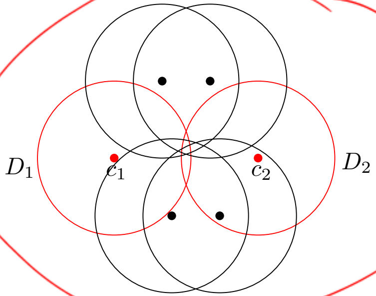
Graphical Clique

- Question1: Given a UDG find a find the maximum Graphical Clique
- Question2: Given a UDG find a find the maximum Geometric Clique

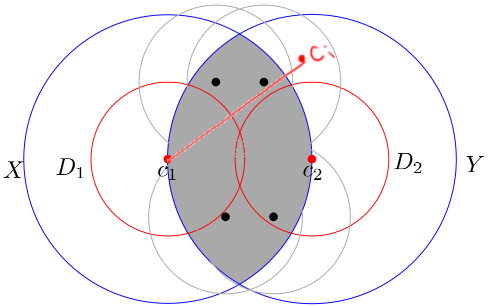


- Let $A = D_1, D_2 \ldots D_n$ be any set of unit disks with centers $c_1 \ldots c_n$
- Let $B = D_1, \ldots D_k$ forms the maximum clique
- D_1, D_2 be the farthest distant pair of disks in A.
- Let X and Y be the disk centered at c_1 and passing through c_2 and Y be the disk centered at c_2 and passing through c_1, Consider the region $X \cap Y$

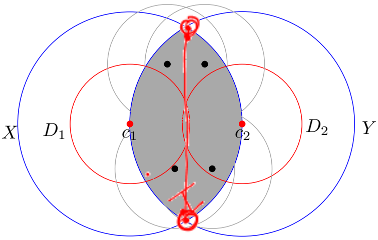
Maximum Graphical Clique



- Let $A = D_1, D_2 \dots D_n$ be any set of unit disks with centers $c_1 \dots c_n$
- Let $B = D_1, \dots D_k$ forms the maximum clique
- D_1, D_2 be the farthest distant pair of disks in A.
- Let X and Y be the disk centered at c_1 and passing through c_2 and Y be the disk centered at c_2 and passing through c_1, Consider the region $X \cap Y$

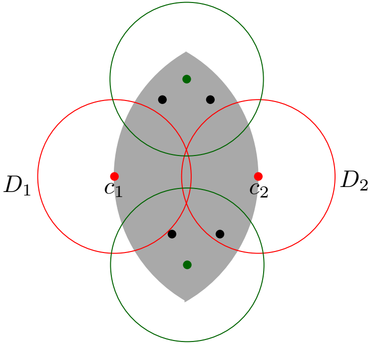


- Let $A = D_1, D_2 \ldots D_n$ be any set of unit disks with centers $c_1 \ldots c_n$
- Let $B = D_1, \ldots D_k$ forms the maximum clique
- D_1, D_2 be the farthest distant pair of disks in A.
- Let X and Y be the disk centered at c_1 and passing through c_2 and Y be the disk centered at c_2 and passing through c_1, Consider the region $X \cap Y$

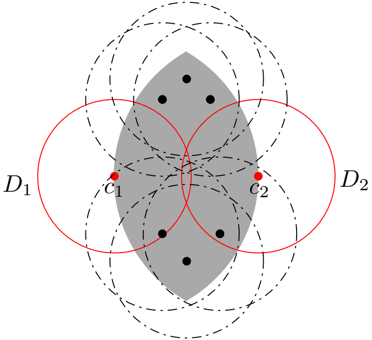


- Does $c_1, c_2 \ldots c_k$ belongs to $X \cap Y$?
- Does all the disks whose center is inside $X \cap Y$ forms a clique?
- May not be ...
- Our new objective is to find maximum clique among the disks in $X \cap Y$

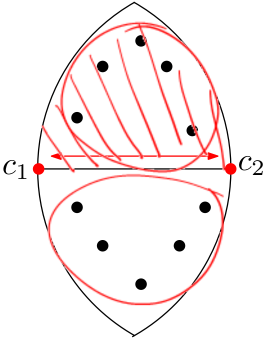
Maximum Graphical Clique



- Does $c_1, c_2 \ldots c_k$ belongs to $X \cap Y$?
- Does all the disks whose center is inside $X \cap Y$ forms a clique?
- May not be ...
- Our new objective is to find maximum clique among the disks in $X \cap Y$

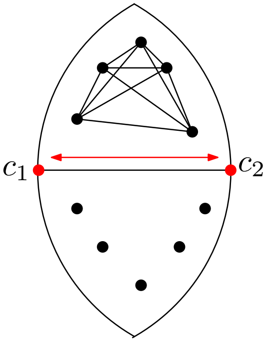


- Does $c_1, c_2 \ldots c_k$ belongs to $X \cap Y$?
- Does all the disks whose center is inside $X \cap Y$ forms a clique?
- May not be ...
- Our new objective is to find maximum clique among the disks in $X \cap Y$

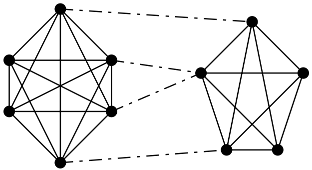


- All the disks above $[c_1, c_2]$ forms a clique.
- All the disks below $[c_1, c_2]$ forms a clique.

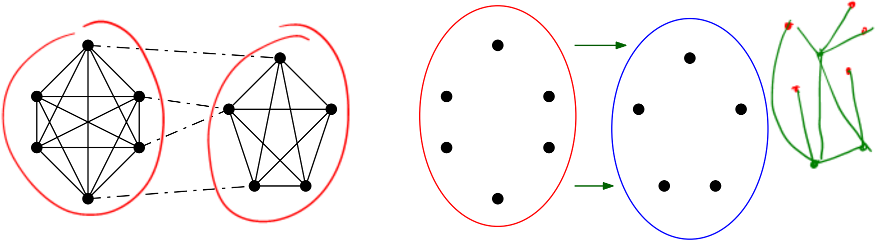
Maximum Graphical Clique



- All the disks above $[c_1, c_2]$ forms a clique.
- All the disks below $[c_1, c_2]$ forms a clique.

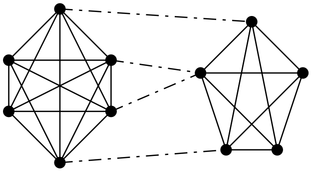


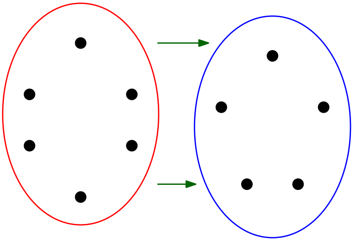
- We have a graph which can be partitioned into two sets V_1 and V_2 where V_1 forms a clique and V_2 forms a clique. There are edges going from V_1 to V_2
- Does this graph looks familiar?
- Its complement is a bipartite graph.
- In this graph we want to find the maximum clique.
- In its complement we are looking for the maximum independent set.



- We have a graph which can be partitioned into two sets V_1 and V_2 where V_1 forms a clique and V_2 forms a clique. There are edges going from V_1 to V_2
- Does this graph looks familiar?
- Its complement is a bipartite graph.
- In this graph we want to find the maximum clique.
- In its complement we are looking for the maximum independent set.

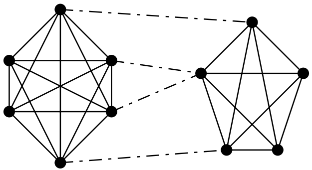
A is a ind. set in G
$V(G) \setminus A$??





- We have a graph which can be partitioned into two sets V_1 and V_2 where V_1 forms a clique and V_2 forms a clique. There are edges going from V_1 to V_2
- Does this graph looks familiar?
- Its complement is a bipartite graph.
- In this graph we want to find the maximum clique.
- In its complement we are looking for the maximum independent set.

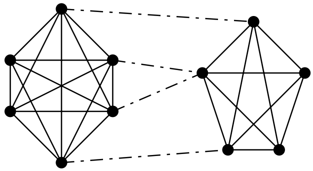
Maximum Graphical Clique

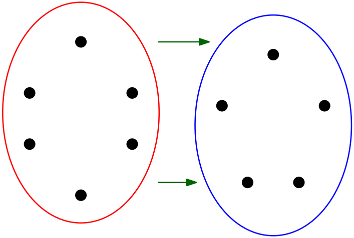


- How to find a maximum independent set in a Bipartite graph.
- The complement of a maximum independent set is a minimum vertex cover.

Theorem

Kőnig's theorem: In any bipartite graph, the number of edges in a maximum matching equals the number of vertices in a minimum vertex cover.

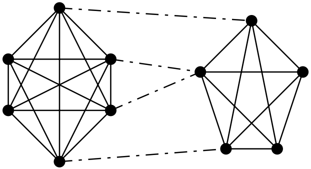


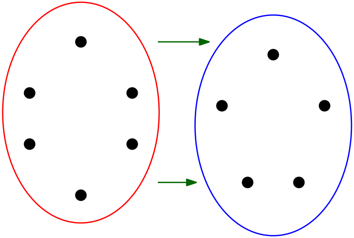


- How to find a maximum independent set in a Bipartite graph.
- The complement of a maximum independent set is a minimum vertex cover.

Theorem

Kőnig's theorem: In any bipartite graph, the number of edges in a maximum matching equals the number of vertices in a minimum vertex cover.





- How to find a maximum independent set in a Bipartite graph.
- The complement of a maximum independent set is a minimum vertex cover.

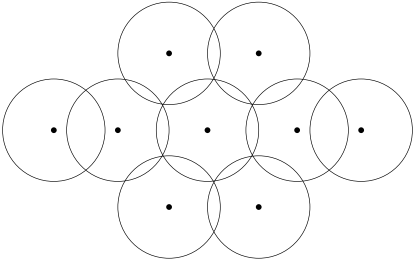
Theorem

Kőnig's theorem: In any bipartite graph, the number of edges in a maximum matching equals the number of vertices in a minimum vertex cover.

Maximum Geometric Clique

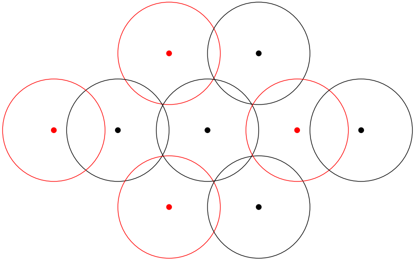
Homework: How to find maximum geometric clique in a UDG?

Maximum Independent Set for UDG

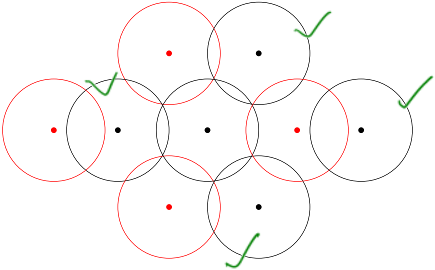


- Given a set of unit disks A a subset of disks $B \subseteq A$ are independent none of the disks in B intersects with each other.
- What about maximum independent set in UDG?
- NP Hard!!!!

Maximum Independent Set for UDG



- Given a set of unit disks A a subset of disks $B \subseteq A$ are independent none of the disks in B intersects with each other.
- What about maximum independent set in UDG?
- NP Hard!!!!



n^k

- Given a set of unit disks A a subset of disks $B \subseteq A$ are independent none of the disks in B intersects with each other.
- What about maximum independent set in UDG?
- NP Hard!!!!

Maximum Independent Set for UDG

- Given a set of unit disks A a subset of disks $B \subseteq A$ are independent none of the disks in B intersects with each other.
- What about maximum independent set in UDG?
- NP Hard!!!!

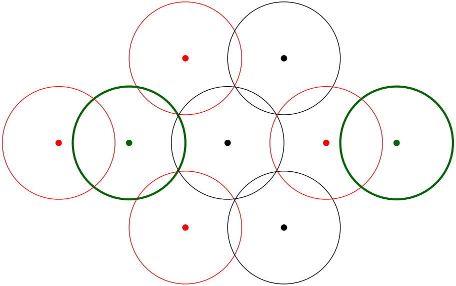
Maximum Independent Set for UDG

- An approximate algorithm does not guarantee the best solution. The goal of an approximate algorithm is to come as close as possible to the optimum value in a reasonable amount of time which is at most polynomial time.
- Let OPT be any optimal solution for the problem at hand, and ALG to denote the (worst case) quality produced by the approximation algorithm under consideration. We would like to guarantee $OPT(I) \geq ALG(I) \geq \frac{1}{\alpha} OPT(I)$ on any instance I for maximization problems, where α is known as the approximation factor.

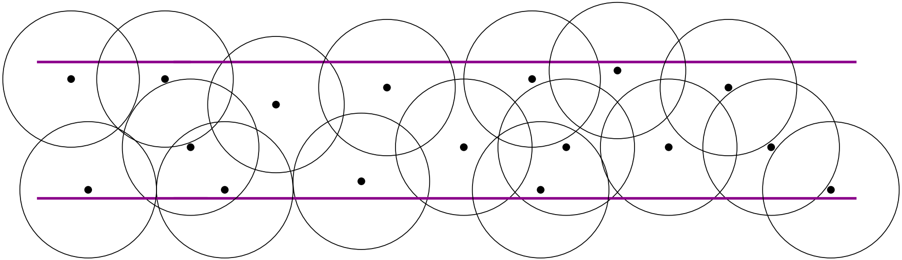
- An approximate algorithm does not guarantee the best solution. The goal of an approximate algorithm is to come as close as possible to the optimum value in a reasonable amount of time which is at most polynomial time.
- Let OPT be any optimal solution for the problem at hand, and ALG to denote the (worst case) quality produced by the approximation algorithm under consideration. We would like to guarantee $OPT(I) \geq ALG(I) \geq \frac{1}{\alpha}OPT(I)$ on any instance I for maximization problems, where α is known as the approximation factor.

- An approximate algorithm does not guarantee the best solution. The goal of an approximate algorithm is to come as close as possible to the optimum value in a reasonable amount of time which is at most polynomial time.
- Let OPT be any optimal solution for the problem at hand, and ALG to denote the (worst case) quality produced by the approximation algorithm under consideration. We would like to guarantee $\boxed{OPT(I) \geq ALG(I) \geq \frac{1}{\alpha} OPT(I)}$ on any instance I for maximization problems, where α is known as the approximation factor.

- A 2- factor approximate algorithm for MIS for UDG will produce a set of independent disks of cardinality at least half of the size of the optimal solution.

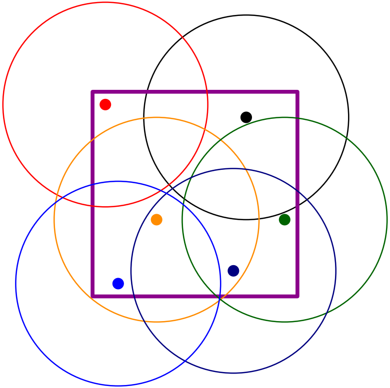


A restricted problem

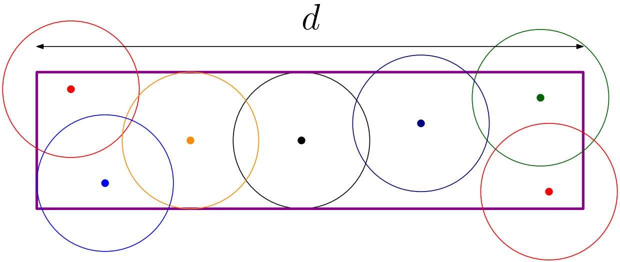


- Consider a set of unit disks centered at a horizontal strip of height 1.
- Find a MIS

More Restriction



- Consider a set of unit disks centered inside a unit box.
- Find a MIS

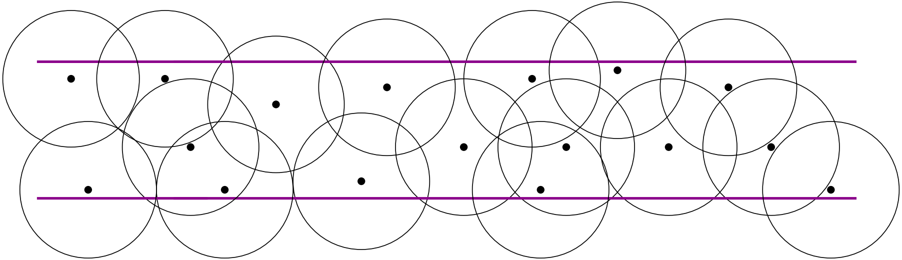


$f(d)$.

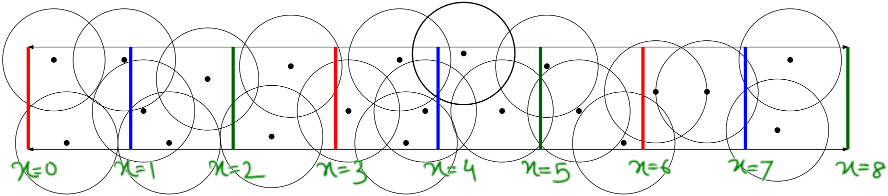
$n^{f(d)}$

- What about finding an MIS is strip of length d where d is constant?

Back to the restricted problem

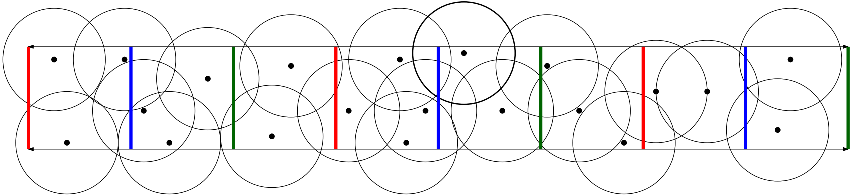


- Consider the lines $l = \{x = i : i \in \mathbb{Z}\}$
- Divide the lines into three sets
  - RED: $\{x = i : i\%3 = 0\}$
  - BLUE: $\{x = i : i\%3 = 1\}$
  - GREEN: $\{x = i : i\%3 = 2\}$

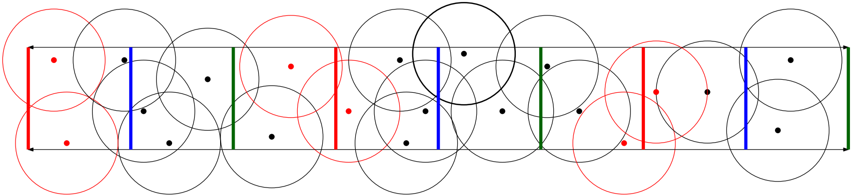


- Consider the lines $I = \{x = i : i \in \mathbb{Z}\}$
- Divide the lines into three sets
  - RED: $\{x = i : i\%3 = 0\}$
  - BLUE: $\{x = i : i\%3 = 1\}$
  - GREEN: $\{x = i : i\%3 = 2\}$

Back to the restricted problem

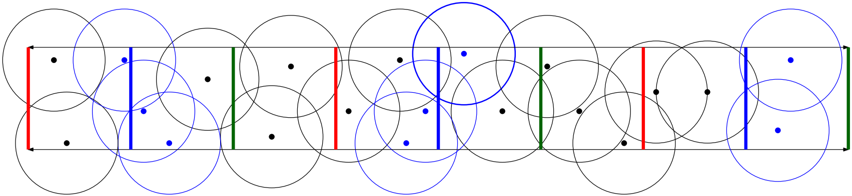


- Let D_0 be the set of disks that intersects red lines.
- Let D_1 be the set of disks that intersects blue lines.
- Let D_2 be the set of disks that intersects green lines.



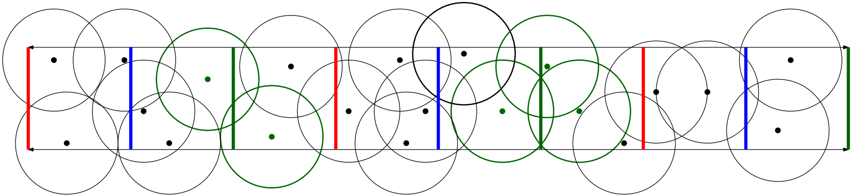
- Let D_0 be the set of disks that intersects red lines.
- Let D_1 be the set of disks that intersects blue lines.
- Let D_2 be the set of disks that intersects green lines.

Back to the restricted problem



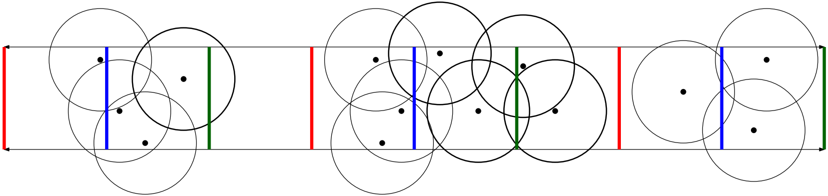
- Let D_0 be the set of disks that intersects red lines.
- Let D_1 be the set of disks that intersects blue lines.
- Let D_2 be the set of disks that intersects green lines.

Back to the restricted problem



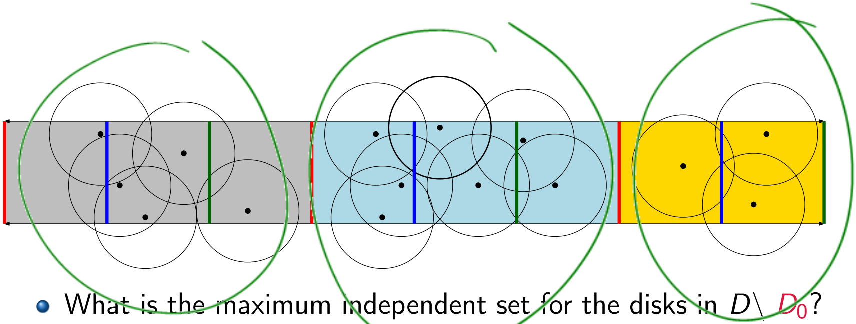
- Let D_0 be the set of disks that intersects red lines.
- Let D_1 be the set of disks that intersects blue lines.
- Let D_2 be the set of disks that intersects green lines.

Back to the restricted problem



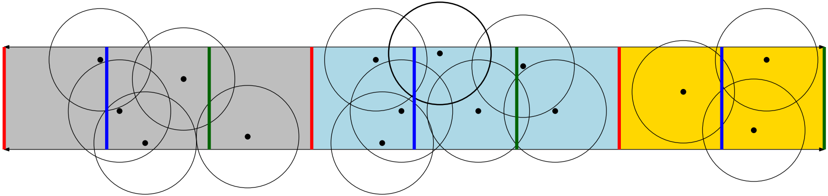
- What is the maximum independent set for the disks in $D \setminus D_0$?
- We can find the MIS for $D \setminus D_0$ in polynomial time, let's denote it by MIS_0
- Define MIS_1 and MIS_2 similarly

Back to the restricted problem



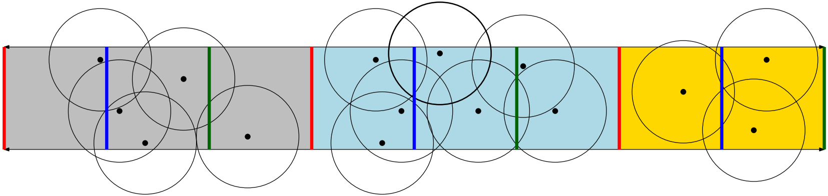
- What is the maximum independent set for the disks in $D \setminus D_0$?
- We can find the MIS for $D \setminus D_0$ in polynomial time, let's denote it by MIS_0
- Define MIS_1 and MIS_2 similarly

Back to the restricted problem



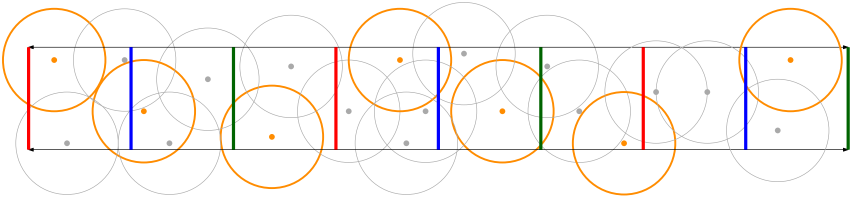
- What is the maximum independent set for the disks in $D\setminus D_0$?
- We can find the MIS for $D\setminus D_0$ in polynomial time, let's denote it by MIS_0
- Define MIS_1 and MIS_2 similarly

Back to the restricted problem



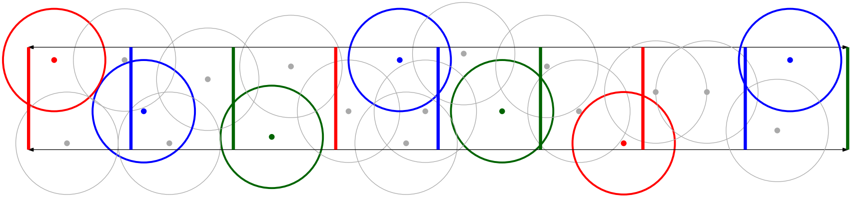
- What is the maximum independent set for the disks in $D \setminus D_0$?
- We can find the MIS for $D \setminus D_0$ in polynomial time, let's denote it by MIS_0
- Define MIS_1 and MIS_2 similarly

Back to the restricted problem



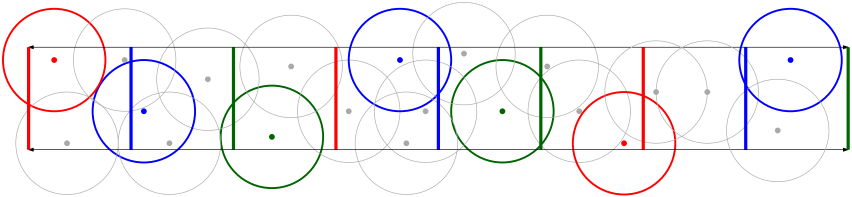
- Consider any optimal solution OPT.
- Set of disks in opt that intersects the red line be OPT_0
- $\mathrm{OPT}_0 = \mathrm{OPT} \cap D_0$
- $\mathrm{OPT}_1 = \mathrm{OPT} \cap D_1$
- $\mathrm{OPT}_2 = \mathrm{OPT} \cap D_3$

Back to the restricted problem



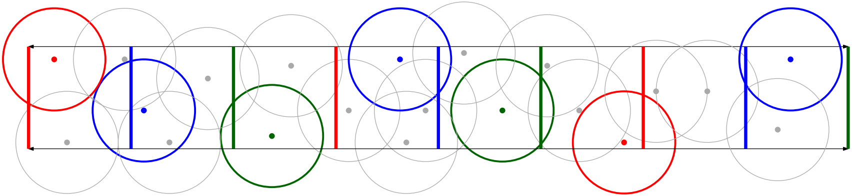
- Consider any optimal solution OPT.
- Set of disks in opt that intersects the red line be OPT_0
- $\mathrm{OPT}_0 = \mathrm{OPT} \cap D_0$
- $\mathrm{OPT}_1 = \mathrm{OPT} \cap D_1$
- $\mathrm{OPT}_2 = \mathrm{OPT} \cap D_3$

Back to the restricted problem



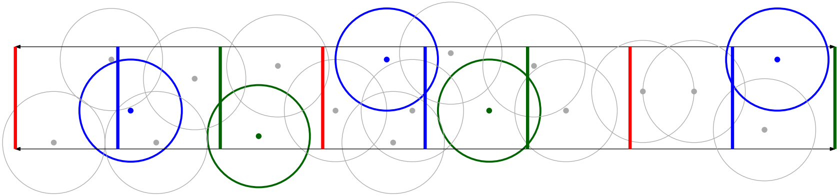
- Consider any optimal solution OPT.
- Set of disks in opt that intersects the red line be OPT_0
- $\mathrm{OPT}_0 = \mathrm{OPT} \cap \; D_0$
- $\mathrm{OPT}_1 = \mathrm{OPT} \cap \; D_1$
- $\mathrm{OPT}_2 = \mathrm{OPT} \cap \; D_3$

Back to the restricted problem

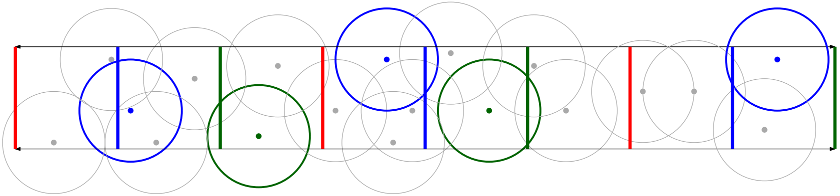


- Consider any optimal solution OPT.
- Set of disks in opt that intersects the red line be OPT_0
- $\mathrm{OPT}_0 = \mathrm{OPT} \cap D_0$
- $\mathrm{OPT}_1 = \mathrm{OPT} \cap D_1$
- $\mathrm{OPT}_2 = \mathrm{OPT} \cap D_3$

Back to the restricted problem

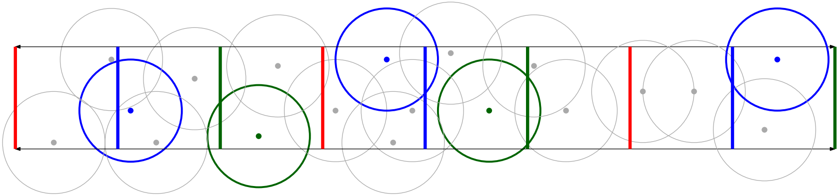


- $\mathrm{OPT} \setminus \mathrm{OPT}_0$ is a independent set for $D \setminus D_0$.
- MIS_0 is a MIS for $D \setminus D_0$.
- $|\, MIS_0 \,| \geq |\, \mathrm{OPT} \setminus \mathrm{OPT}_0 \,|$
- $|\, MIS_1 \,| \geq |\, \mathrm{OPT} \setminus \mathrm{OPT}_1 \,|$
- $|\, MIS_2 \,| \geq |\, \mathrm{OPT} \setminus \mathrm{OPT}_2 \,|$
- One of OPT_0, OPT_1, or OPT_2 is at most $\mathrm{OPT}/3$
- Say $\mathrm{OPT}_0 \leq \mathrm{OPT}/3$
- $|\, MIS_0 \,| \geq |\, \mathrm{OPT} \setminus \mathrm{OPT}_0 \,| \geq |\mathrm{OPT}| - 1/3\mathrm{OPT} \geq 2/3\mathrm{OPT}$



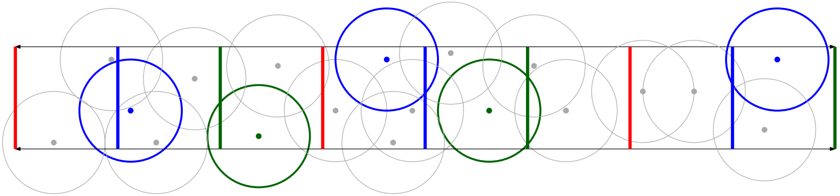
- $\mathrm{OPT} \setminus \mathrm{OPT}_0$ is a independent set for $D \setminus D_0$.
- MIS_0 is a MIS for $D \setminus D_0$.
- $|MIS_0| \geq |\mathrm{OPT} \setminus \mathrm{OPT}_0|$
- $|MIS_1| \geq |\mathrm{OPT} \setminus \mathrm{OPT}_1|$
- $|MIS_2| \geq |\mathrm{OPT} \setminus \mathrm{OPT}_2|$
- One of OPT_0, OPT_1, or OPT_2 is at most $\mathrm{OPT}/3$
- Say $\mathrm{OPT}_0 \leq \mathrm{OPT}/3$
- $|MIS_0| \geq |\mathrm{OPT} \setminus \mathrm{OPT}_0| \geq |\mathrm{OPT}| - 1/3\mathrm{OPT} \geq 2/3\mathrm{OPT}$

Back to the restricted problem

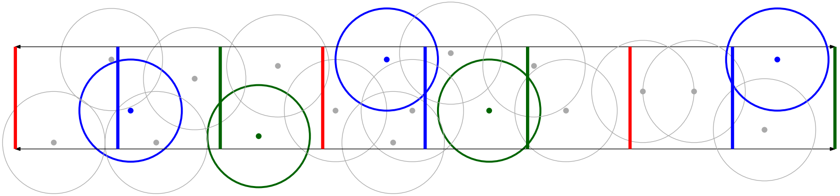


- $\text{OPT} \setminus \text{OPT}_0$ is a independent set for $D \setminus D_0$.
- MIS_0 is a MIS for $D \setminus D_0$.
- $\mid MIS_0 \mid \geq \mid \text{OPT} \setminus \text{OPT}_0 \mid$
- $\mid MIS_1 \mid \geq \mid \text{OPT} \setminus \text{OPT}_1 \mid$
- $\mid MIS_2 \mid \geq \mid \text{OPT} \setminus \text{OPT}_2 \mid$
- One of OPT_0, OPT_1, or OPT_2 is at most $\text{OPT}/3$
- Say $\text{OPT}_0 \leq \text{OPT}/3$
- $\mid MIS_0 \mid \geq \mid \text{OPT} \setminus \text{OPT}_0 \mid \geq |\text{OPT}| - 1/3\text{OPT} \geq 2/3\text{OPT}$

Back to the restricted problem



- $\text{OPT} \setminus \text{OPT}_0$ is a independent set for $D \setminus D_0$.
- MIS_0 is a MIS for $D \setminus D_0$.
- $|\ MIS_0\ | \geq |\ \text{OPT} \setminus \text{OPT}_0\ |$
- $|\ MIS_1\ | \geq |\ \text{OPT} \setminus \text{OPT}_1\ |$
- $|\ MIS_2\ | \geq |\ \text{OPT} \setminus \text{OPT}_2\ |$
- One of OPT_0, OPT_1, or OPT_2 is at most $\text{OPT}/3$
- Say $\text{OPT}_0 \leq \text{OPT}/3$
- $|\ MIS_0\ | \geq |\ \text{OPT} \setminus \text{OPT}_0\ | \geq |\text{OPT}| - 1/3\text{OPT} \geq 2/3\text{OPT}$

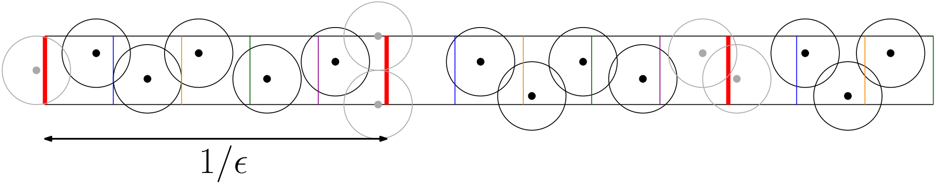


- $\textsc{opt} \setminus \textsc{opt}_0$ is a independent set for $D \setminus D_0$.
- MIS_0 is a MIS for $D \setminus D_0$.
- $| MIS_0 | \geq | \textsc{opt} \setminus \textsc{opt}_0 |$
- $| MIS_1 | \geq | \textsc{opt} \setminus \textsc{opt}_1 |$
- $| MIS_2 | \geq | \textsc{opt} \setminus \textsc{opt}_2 |$
- One of $\textsc{opt}_0$, $\textsc{opt}_1$, or $\textsc{opt}_2$ is at most $\textsc{opt}/3$
- Say $\textsc{opt}_0 \leq \textsc{opt}/3$
- $| MIS_0 | \geq | \textsc{opt} \setminus \textsc{opt}_0 | \geq |\textsc{opt}| - 1/3\textsc{opt} \geq 2/3\textsc{opt}$

- Roughly in $O(n^3)$ time we can find out a independent set of size at lease $\cancel{\mathrm{OPT}/3}$. $\mathrm{OPT}(1 - 1/3)$
- For any value $0 < \epsilon < 1$, I will divide the strip into $1/\epsilon$ colors

Back to the restricted problem

- Roughly in $O(n^3)$ time we can find out a independent set of size at lease $\text{OPT}/3$.
- For any value $0 < \epsilon < 1$, I will divide the strip into $1/\epsilon$ colors



Polynomial-time approximation scheme

Theorem

In time $O(n^{1/\epsilon})$ time we can find an independent set of size at lease $(1-\epsilon)\mathrm{OPT}$.

- This kind of algorithms are called Polynomial-time approximation scheme(PTAS)

Polynomial-time approximation scheme

Theorem

In time $O(n^{1/\epsilon})$ time we can find an independent set of size at lease $(1 - \epsilon)\mathrm{OPT}$.

- This kind of algorithms are called Polynomial-time approximation scheme(PTAS)