Multivariate Exploration of Metric Dilation

Aritra Banik[1], Fedor V. Fomin[2], Petr A. Golovach[2], Tanmay Inamdar[3], Satyabrata Jana[4], Saket Saurabh[2,5]

[1]NISER Bhubaneswar, India

[2]UiB, Bergen, Norway [3]IIT Jodhpur, India

[4]University of Warwick, UK

[5]IMSc, Chennai, India

SYMPOSIUM ON THEORETICAL ASPECTS OF COMPUTER SCIENCE

Jena, Germany
05 Mar 2025

Outline

Basic Definitions

Basic Definitions

- **Graph embedded in a *Metric Space***

Basic Definitions

- **Graph embedded in a Metric Space**

  An edge weighted graph G is embedded on $\mathcal{M} = (V, d_M)$

Basic Definitions

- **Graph embedded in a Metric Space**

An edge weighted graph G is embedded on $\mathcal{M} = (V, d_M)$

$$\Downarrow$$

for each edge $e \in E(G)$, weight $(e) = d_M(u, v)$

Basic Definitions

- **Graph embedded in a Metric Space**

  An edge weighted graph G is embedded on $\mathcal{M} = (V, d_M)$

  $\Downarrow$

  for each edge $e \in E(G)$, weight $(e) = d_M(u, v)$

- G **has dialation** t

Basic Definitions

- **Graph embedded in a Metric Space**

  An edge weighted graph G is embedded on $\mathcal{M} = (V, d_M)$

  $$\Downarrow$$

  for each edge $e \in E(G)$, weight $(e) = d_M(u, v)$

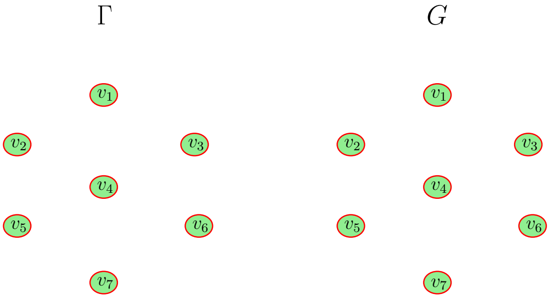
- G **has dialation** t

  for each pair $u, v \in V(G)$,

  $$d_G(u, v) \leq t \cdot d_M(u, v)$$

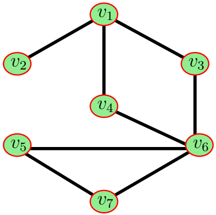
Example

Example

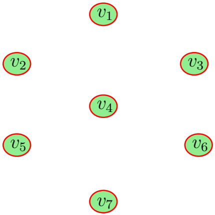


Example

Γ

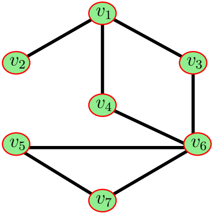


G

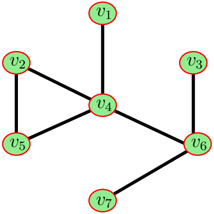


Example

Γ

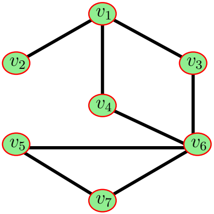


G

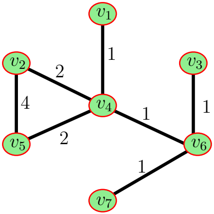


Example

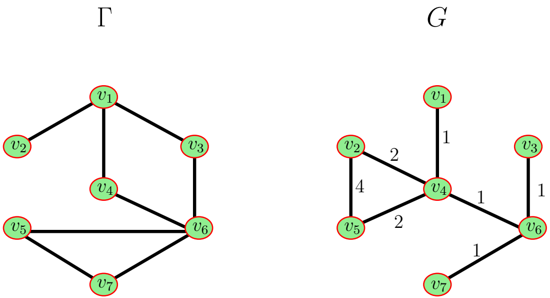
Γ



G

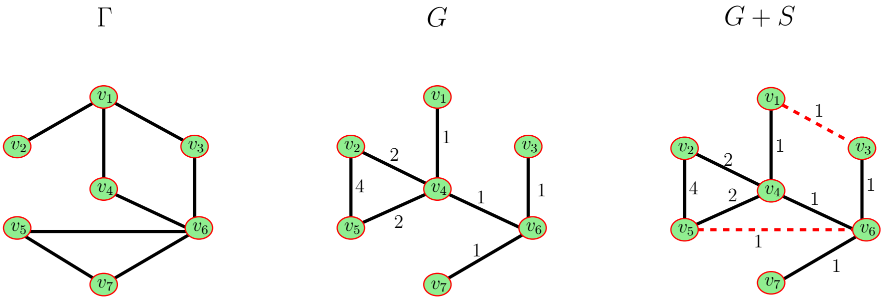


Example



$d_\Gamma(v_1, v_3) = 1$ $\quad d_G(v_1, v_3) = 3$ $\quad d_\Gamma(v_5, v_7) = 1$ $\quad d_G(v_5, v_7) = 4$

Example



$d_\Gamma(v_1, v_3) = 1 \qquad d_G(v_1, v_3) = 3$

$d_{G+S}(v_1, v_3) = 1$

$d_\Gamma(v_5, v_7) = 1 \qquad d_G(v_5, v_7) = 4$

$d_{G+S}(v_5, v_7) = 2$

Problem Definition

Problem Definition

Dilation t-Augmentation

Input:

Task:

Problem Definition

Dilation t-Augmentation

Input: A graph G embedded in a metric space $\mathcal{M}$ on $V(G)$, $k \in \mathbb{N}$.
Task:

Problem Definition

DILATION t-AUGMENTATION

Input: A graph G embedded in a metric space $\mathcal{M}$ on $V(G)$, $k \in \mathbb{N}$.
Task: Add k edges to G such that dialation becomes at most t.

Problem Definition

Dilation t-Augmentation

Input: A graph G embedded in a metric space $\mathcal{M}$ on $V(G)$, $k \in \mathbb{N}$.
Task: Add k edges to G such that dialation becomes at most t.

☞ For which $\mathcal{M}$, Dilation t-Augmentation admit an FPT algorithm?

Problem Definition

Dilation t-Augmentation

Input: A graph G embedded in a metric space $\mathcal{M}$ on $V(G)$, $k \in \mathbb{N}$.
Task: Add k edges to G such that dialation becomes at most t.

- ☞ For which $\mathcal{M}$, Dilation t-Augmentation admit an FPT algorithm?
- ☞ For which graph class $\mathcal{G}$, Dilation t-Augmentation admit an FPT algorithm?

Problem Definition

Dilation t-Augmentation

Input: A graph G embedded in a metric space $\mathcal{M}$ on $V(G)$, $k \in \mathbb{N}$.
Task: Add k edges to G such that dialation becomes at most t.

- ☞ For which $\mathcal{M}$, Dilation t-Augmentation admit an FPT algorithm?
- ☞ For which graph class $\mathcal{G}$, Dilation t-Augmentation admit an FPT algorithm?
- ☞ For which pairs, $(\mathcal{G}, \mathcal{M})$, Dilation t-Augmentation admit an FPT algorithm?

Outline

What is known?

What is known?

Diameter 2-Augmentation

Input:
Task:

What is known?

Diameter 2-Augmentation

Input: A graph G, an integer k.

Task:

What is known?

Diameter 2-Augmentation

Input: A graph G, an integer k.

Task: Add k edges to G such that diameter of G becomes 2.

What is known?

Diameter 2-Augmentation

Input: A graph G, an integer k.

Task: Add k edges to G such that diameter of G becomes 2.

☞ This problem is W[2]-hard [Gao, Hare, and Nastos.; DAM '13]

What is known?

DIAMETER 2-AUGMENTATION

Input: A graph G, an integer k.

Task: Add k edges to G such that diameter of G becomes 2.

☞ This problem is W[2]-hard [Gao, Hare, and Nastos.; DAM '13]

$\Downarrow$

What is known?

Diameter 2-Augmentation

Input: A graph G, an integer k.
Task: Add k edges to G such that diameter of G becomes 2.

☞ This problem is W[2]-hard [Gao, Hare, and Nastos.; DAM '13]

$\Downarrow$

☞ Dilation t-Augmentation **can not** be solved in time $g(k) \cdot n^{\mathcal{O}(t)}$?

Related Problem 1

Related Problem 1

For a given real number $t > 1$, we say that $G' \subseteq G$ is a t-spanner of G, if the dilation of G' with respect to G, i.e. $\max_{u,v} \frac{d_{G'}(u,v)}{d_G(u,v)}$ is at most t.

Related Problem 1

For a given real number $t > 1$, we say that $G' \subseteq G$ is a t-spanner of G, if the dilation of G' with respect to G, i.e. $\max_{u,v} \frac{d_{G'}(u,v)}{d_G(u,v)}$ is at most t.

Spanner with few edges

Input:
Question:

Related Problem 1

For a given real number $t > 1$, we say that $G' \subseteq G$ is a t-spanner of G, if the dilation of G' with respect to G, i.e. $\max_{u,v} \frac{d_{G'}(u,v)}{d_G(u,v)}$ is at most t.

Spanner with few edges

Input: A connected weighted graph G, a real number t.
Question:

Related Problem 1

For a given real number $t > 1$, we say that $G' \subseteq G$ is a t-spanner of G, if the dilation of G' with respect to G, i.e. $\max_{u,v} \frac{d_{G'}(u,v)}{d_G(u,v)}$ is at most t.

Spanner with few edges

Input: A connected weighted graph G, a real number t.
Question: Does G contains a t-spanner with **few** edges?

Related Problem 1

For a given real number $t > 1$, we say that $G' \subseteq G$ is a t-spanner of G, if the dilation of G' with respect to G, i.e. $\max_{u,v} \frac{d_{G'}(u,v)}{d_G(u,v)}$ is at most t.

Spanner with few edges

Input: A connected weighted graph G, a real number t.
Question: Does G contains a t-spanner with **few** edges?

☞ Any connected weighted graph with n vertices contains a t-spanner with $\mathcal{O}(tn^{1+\frac{2}{t+1}})$ edges [Baswana and Sen.; ICALP '03]

Related Problem 1

For a given real number $t > 1$, we say that $G' \subseteq G$ is a t-spanner of G, if the dilation of G' with respect to G, i.e. $\max_{u,v} \frac{d_{G'}(u,v)}{d_G(u,v)}$ is at most t.

SPANNER WITH FEW EDGES

Input: A connected weighted graph G, a real number t.
Question: Does G contains a t-spanner with **few** edges?

- ☞ Any connected weighted graph with n vertices contains a t-spanner with $\mathcal{O}(tn^{1+\frac{2}{t+1}})$ edges [Baswana and Sen.; ICALP '03]
- ☞ Deciding whether a geometric graph contains a t-spanner with $\leq k$ edges is NP-hard? [Gudmundsson and Michiel Smid .; SWAT '06]

Related Problem 2

Related Problem 2

For a given real number $t > 1$, a geometric graph G (drawn in straight line in plane) is a t-spanner or has dilation t if $\max_{u,v} \frac{d_G(u,v)}{|uv|}$ is at most t.

Related Problem 2

For a given real number $t > 1$, a geometric graph G (drawn in straight line in plane) is a t-spanner or has dilation t if $\max_{u,v} \frac{d_G(u,v)}{|uv|}$ is at most t.

Minimum Dialation

Input:
Task:

Related Problem 2

For a given real number $t > 1$, a geometric graph G (drawn in straight line in plane) is a t-spanner or has dilation t if $\max_{u,v} \frac{d_G(u,v)}{|uv|}$ is at most t.

Minimum Dialation

Input: A set of n points in the plane, a positive integer k.
Task:

Related Problem 2

For a given real number $t > 1$, a geometric graph G (drawn in straight line in plane) is a t-spanner or has dilation t if $\max_{u,v} \frac{d_G(u,v)}{|uv|}$ is at most t.

Minimum Dialation

Input: A set of n points in the plane, a positive integer k.
Task: Compute a geometric minimum-dilation graph using k edges

Related Problem 2

For a given real number $t > 1$, a geometric graph G (drawn in straight line in plane) is a t-spanner or has dilation t if $\max_{u,v} \frac{d_G(u,v)}{|uv|}$ is at most t.

Minimum Dialation

Input: A set of n points in the plane, a positive integer k.
Task: Compute a geometric minimum-dilation graph using k edges

☞ This problem is NP-hard [Giannopoulos, Klein, Knauer, Kutz, and Marx.; IJCGA '10]

Outline

Our Results

Our Results

G	Metric (Γ)	Parameter(s)	t	Complexity
$\mathcal{K}_{d,d}$-free	General	$k + d$	2	FPT
Star forest	General	k	3	W[1]-hard
General	Star	k	3	W[1]-hard
General	Tree	k	2	P
General	Max deg-δ	$k + t + \delta$	t	FPT
Max deg-δ	General	$k + t + \delta$	t	FPT
Subcubic	$(1, w)$ -weighted	k	$2 + \varepsilon$	W[1]-hard
Edgeless	General	k	2	NP-hard
General	Clique	k	2	W[2]-hard

This Talk

G	Metric (Γ)	Parameter(s)	t	Complexity
$\mathcal{K}_{d,d}$-free	General	$k + d$	2	FPT
Star forest	General	k	3	W[1]-hard
General	Star	k	3	W[1]-hard
General	Tree	k	2	P
General	Max deg-δ	$k + t + \delta$	t	FPT
Max deg-δ	General	$k + t + \delta$	t	FPT
Subcubic	$(1, w)$ -weighted	k	$2 + \varepsilon$	W[1]-hard
Edgeless	General	k	2	NP-hard
General	Clique	k	2	W[2]-hard

Outline

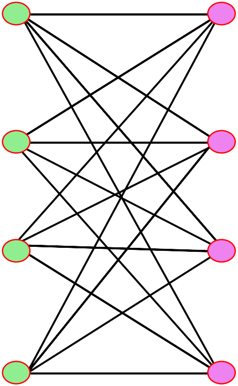
What is $\mathcal{K}_{d,d}$-free graphs?

What is $\mathcal{K}_{d,d}$-free graphs?

It does not contain a **complete bipartite** graph as a subgraph.

What is $\mathcal{K}_{d,d}$-free graphs?

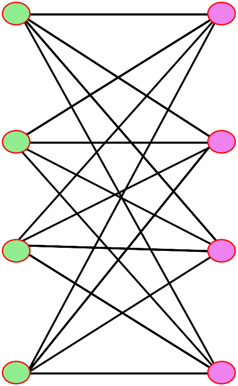
It does not contain a **complete bipartite** graph as a subgraph.



$\mathcal{K}_{4,4}$

What is $\mathcal{K}_{d,d}$-free graphs?

It does not contain a **complete bipartite** graph as a subgraph.



$\mathcal{K}_{4,4}$

It includes trees, planar graphs, H minor-free graphs, graphs of bounded expansion, nowhere dense graphs, and graphs of bounded degeneracy

Conflict versus Adjacent Conflicts

Conflict versus Adjacent Conflicts

u and v are said to be in **conflict** if $d_G(u, v) > t \cdot d_M(u, v)$.

Conflict versus Adjacent Conflicts

u and v are said to be in **conflict** if $d_G(u, v) > t \cdot d_M(u, v)$.

Definition u and v are said to be in **adjacent conflict** if

1. u and v are in conflict.
2. u and v are adjacent in Γ.

Conflict versus Adjacent Conflicts

u and v are said to be in **conflict** if $d_G(u,v) > t \cdot d_M(u,v)$.

Definition u and v are said to be in **adjacent conflict** if

1. u and v are in conflict.
2. u and v are adjacent in Γ.

Lemma

- Consider a set of edges S
- $G + S$ is conflict-free $\Leftrightarrow$ $G + S$ is adjacent conflict-free.

Conflict versus Adjacent Conflicts

u and v are said to be in **conflict** if $d_G(u, v) > t \cdot d_M(u, v)$.

Definition u and v are said to be in **adjacent conflict** if

1. u and v are in conflict.
2. u and v are adjacent in Γ.

Lemma

- Consider a set of edges S
- $G + S$ is conflict-free $\Leftrightarrow$ $G + S$ is adjacent conflict-free.

$\Downarrow$

☞ **We only need to focus on adjacent conflicts**

Our First Target

Our First Target

Bound the number of conflict pairs.

Conflict Graph and Vertex Cover

Conflict Graph and Vertex Cover

Definition (Conflict Graph $\mathbb{C}$)

- It captures all adjacent conflicts.
- $V(\mathbb{C}) = V(G)$
- $E(\mathbb{C}) : uv \in E(\mathbb{C}) \Leftrightarrow u$ and v are in adjacent conflict.

Conflict Graph and Vertex Cover

Definition (Conflict Graph $\mathbb{C}$)

- It captures all adjacent conflicts.
- $V(\mathbb{C}) = V(G)$
- $E(\mathbb{C}) : uv \in E(\mathbb{C}) \Leftrightarrow u$ and v are in adjacent conflict.

Lemma

- Consider a solution S
- Every edge in $E(\mathbb{C})$ has an endpoint in V_S

Conflict Graph and Vertex Cover

Definition (Conflict Graph $\mathbb{C}$)

- It captures all adjacent conflicts.
- $V(\mathbb{C}) = V(G)$
- $E(\mathbb{C}) : uv \in E(\mathbb{C}) \Leftrightarrow u$ and v are in adjacent conflict.

Lemma

- Consider a solution S
- Every edge in $E(\mathbb{C})$ has an endpoint in V_S

☞ **the size of a maximum matching in $\mathbb{C}$ is bounded**

Conflict Graph and Vertex Cover

Definition (Conflict Graph $\mathbb{C}$)

- It captures all adjacent conflicts.
- $V(\mathbb{C}) = V(G)$
- $E(\mathbb{C})$: $uv \in E(\mathbb{C}) \Leftrightarrow u$ and v are in adjacent conflict.

Lemma

- Consider a solution S
- Every edge in $E(\mathbb{C})$ has an endpoint in V_S

☞ **the size of a maximum matching in $\mathbb{C}$ is bounded**

$\Downarrow$

☞ **the size of a minimum vertex cover in $\mathbb{C}$ is bounded**

First Reduction Rule

First Reduction Rule

Reduction Rule 1.

- If the size of a maximum matching in $\mathbb{C}$ is $> 2k$, return `No`.

First Reduction Rule

Reduction Rule 1.

- If the size of a maximum matching in $\mathbb{C}$ is $> 2k$, return `No`.

$$\Downarrow$$

- **the size of a minimum vertex cover in $\mathbb{C}$ is at most $4k$**

First Reduction Rule

Reduction Rule 1.

- If the size of a maximum matching in $\mathbb{C}$ is $> 2k$, return `No`.

$$\Downarrow$$

- **the size of a minimum vertex cover in $\mathbb{C}$ is at most $4k$**
- We can compute a $4k$ size vertex cover R in $n^{\mathcal{O}(1)}$ time.

First Reduction Rule

Reduction Rule 1.

- If the size of a maximum matching in $\mathbb{C}$ is $> 2k$, return No.

$\Downarrow$

- **the size of a minimum vertex cover in $\mathbb{C}$ is at most $4k$**
- We can compute a $4k$ size vertex cover R in $n^{\mathcal{O}(1)}$ time.
- Towards Bound $|E(\mathbb{C})|$,

  We bound the degree of each vertex in R in $\mathbb{C}$.

Brief Sketch

Brief Sketch

- **First we guess the edges of S that have both end-points in R**

Brief Sketch

- **First we guess the edges of S that have both end-points in R**
- Now we are not allowed to add an edge to a solution that does not have an end-point outside R (**invariant**).

Brief Sketch

- **First we guess the edges of S that have both end-points in R**
- Now we are not allowed to add an edge to a solution that does not have an end-point outside R (**invariant**).
- Consider a vertex of R that have high degrees in $\mathbb{C}$

Brief Sketch

- **First we guess the edges of S that have both end-points in R**
- Now we are not allowed to add an edge to a solution that does not have an end-point outside R (**invariant**).
- Consider a vertex of R that have high degrees in $\mathbb{C}$
- We will add at least one vertex from $I = V \setminus R$ to R

Brief Sketch

- **First we guess the edges of S that have both end-points in R**
- Now we are not allowed to add an edge to a solution that does not have an end-point outside R (**invariant**).
- Consider a vertex of R that have high degrees in $\mathbb{C}$
- We will add at least one vertex from $I = V \setminus R$ to R
- We add one new edge to S.

Brief Sketch

- **First we guess the edges of S that have both end-points in R**
- Now we are not allowed to add an edge to a solution that does not have an end-point outside R (**invariant**).
- Consider a vertex of R that have high degrees in $\mathbb{C}$
- We will add at least one vertex from $I = V \setminus R$ to R
- We add one new edge to S.
- So in each step, the budget k reduces by at least one

Brief Sketch

- **First we guess the edges of S that have both end-points in R**
- Now we are not allowed to add an edge to a solution that does not have an end-point outside R (**invariant**).
- Consider a vertex of R that have high degrees in $\mathbb{C}$
- We will add at least one vertex from $I = V \setminus R$ to R
- We add one new edge to S.
- So in each step, the budget k reduces by at least one
- Stop when $k = 0$ or degree of each vertex of R gets bounded.

Bounding the Degree of $v \in R$ in $\mathbb{C}$

Bounding the Degree of $v \in R$ in $\mathbb{C}$

- Consider a vertex $v \in R$ with at least $f(0) + 1$ neighbors in I in $\mathbb{C}$.

Bounding the Degree of $v \in R$ in $\mathbb{C}$

- Consider a vertex $v \in R$ with at least $f(0) + 1$ neighbors in I in $\mathbb{C}$.
- Let $U :=$ the set of neighbours of v in I

Bounding the Degree of $v \in R$ in $\mathbb{C}$

- Consider a vertex $v \in R$ with at least $f(0) + 1$ neighbors in I in $\mathbb{C}$.
- Let $U :=$ the set of neighbours of v in I

Reduction Rule 2.

If there is no $w \in I$ with $|U \cap N_G(w)| > f(1)$ then return `No`.

Bounding the Degree of $v \in R$ in $\mathbb{C}$

- Consider a vertex $v \in R$ with at least $f(0)+1$ neighbors in I in $\mathbb{C}$.
- Let $U :=$ the set of neighbours of v in I

Reduction Rule 2.

If there is no $w \in I$ with $|U \cap N_G(w)| > f(1)$ then return `No`.

- For `Yes` instance, we get a vertex $w \in I$ with $|U \cap N_G(w)| > f(1)$.

Bounding the Degree of $v \in R$ in $\mathbb{C}$

- Consider a vertex $v \in R$ with at least $f(0) + 1$ neighbors in I in $\mathbb{C}$.
- Let $U :=$ the set of neighbours of v in I

Reduction Rule 2.

If there is no $w \in I$ with $|U \cap N_G(w)| > f(1)$ then return `No`.

- For `Yes` instance, we get a vertex $w \in I$ with $|U \cap N_G(w)| > f(1)$.
- We use w to identify a small set of edges that must intersect S.

Bounding the Degree of $v \in R$ in $\mathbb{C}$

Bounding the Degree of $v \in R$ in $\mathbb{C}$

- We use w to identify a small set of edges that must intersect S.

Bounding the Degree of $v \in R$ in $\mathbb{C}$

- We use w to identify a small set of edges that must intersect S.

Lemma.

There exists a polynomial-time algorithm to find a set W_v such that

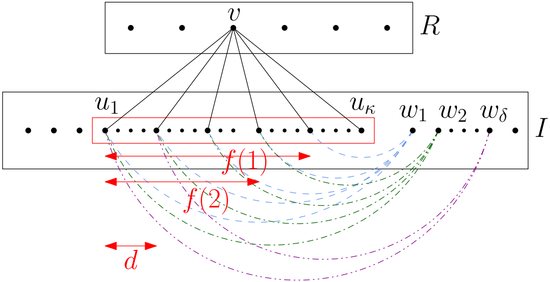
1. $|W_v| < d$
2. For Yes instance, $S \cap \{(v, w) : w \in W_v\} \neq \emptyset$.

Bounding the Degree of $v \in R$ in $\mathbb{C}$

- We use w to identify a small set of edges that must intersect S.

Lemma.
There exists a polynomial-time algorithm to find a set W_v such that

1. $|W_v| < d$
2. For Yes instance, $S \cap \{(v, w) : w \in W_v\} \neq \emptyset$.



Branching

- We use w to identify a small set of edges that must intersect S.

Lemma.
There exists a polynomial-time algorithm to find a set W_v such that

1. $|W_v| < d$
2. For Yes instance, $S \cap \{(v, w) : w \in W_v\} \neq \emptyset$.

Branching

- We use w to identify a small set of edges that must intersect S.

Lemma.
There exists a polynomial-time algorithm to find a set W_v such that

1. $|W_v| < d$
2. For Yes instance, $S \cap \{(v, w) : w \in W_v\} \neq \emptyset$.

- Guess the set $W' \subseteq W_v$ for which such edges belong to the solution.

Branching

- We use w to identify a small set of edges that must intersect S.

Lemma.
There exists a polynomial-time algorithm to find a set W_v such that

1. $|W_v| < d$
2. For Yes instance, $S \cap \{(v, w) : w \in W_v\} \neq \emptyset$.

- Guess the set $W' \subseteq W_v$ for which such edges belong to the solution.
- Guess the edges in S which is in b/w W' and R (due to invariant).

Branching

- We use w to identify a small set of edges that must intersect S.

Lemma.
There exists a polynomial-time algorithm to find a set W_v such that

1. $|W_v| < d$
2. For Yes instance, $S \cap \{(v, w) : w \in W_v\} \neq \emptyset$.

- Guess the set $W' \subseteq W_v$ for which such edges belong to the solution.
- Guess the edges in S which is in b/w W' and R (due to invariant).
- Add W' to the vertex cover R

How is f?

How is f?

$$f(i) = \begin{cases} d & \text{if } i = d \\ d \cdot k + k^2 + k & \text{if } i = d - 1 \\ d \cdot k^{d-i} + k^{d-i+1} + \{2 \cdot \sum_{j=2}^{d-i} k^j\} + k & \text{if } 0 \leq i \leq d - 2 \end{cases}$$

- For any $1 \leq i \leq d$, $f(i-1) = (f(i) + k) \cdot k + k$.

Reduced Instance

Reduced Instance

- Now each vertex in R has at most $f(0)$ neighbors in I in the graph $\mathbb{C}$.

Reduced Instance

- Now each vertex in R has at most $f(0)$ neighbors in I in the graph $\mathbb{C}$.
- So total number of vertices in the conflict is at most $5k \cdot f(0)$

Reduced Instance

- Now each vertex in R has at most $f(0)$ neighbors in I in the graph $\mathbb{C}$.
- So total number of vertices in the conflict is at most $5k \cdot f(0)$

Annotated Dilation 2-Augmentation

Input: G, Γ, k, R, V_c. Here $|R| \leq 5k$ and $|V_c| = 5k \cdot f(0)$
Task: Find a dilation 2-augmentation set S such that every edge in S has at least one end-point outside R.

Solving Annotated Instances with bounded V_c

Solving Annotated Instances with bounded V_c

- We cannot completely forget about the vertices outside V_c

Solving Annotated Instances with bounded V_c

- We cannot completely forget about the vertices outside V_c
- We do not need to keep all such vertices, but it suffices to keep **one representative** from each **equivalence class**.

Solving Annotated Instances with bounded V_c

- We cannot completely forget about the vertices outside V_c
- We do not need to keep all such vertices, but it suffices to keep **one representative** from each **equivalence class**.
- Let $O := V(G) \setminus V_c$

Solving Annotated Instances with bounded V_c

- We cannot completely forget about the vertices outside V_c
- We do not need to keep all such vertices, but it suffices to keep **one representative** from each **equivalence class**.
- Let $O := V(G) \setminus V_c$

Definition For each $A, B \subseteq V_c$ with $A \cap B = \emptyset$, let $O(A, B)$ denote the set of vertices $v \in O$ satisfying

1. $A = \{u \ : \ u \in V_c,\ d_G(u, v) = 1\}$, and
2. $B = \{w \ : \ w \in V_c,\ (v, w) \notin E(G),\ d_\Gamma(v, w) = 1\}$.

Solving Annotated Instances with bounded V_c

Definition For each $A, B \subseteq V_c$ with $A \cap B = \emptyset$, let $O(A, B)$ denote the set of vertices $v \in O$ satisfying

1. $A = \{u \ : \ u \in V_c, \ d_G(u, v) = 1\}$, and
2. $B = \{w \ : \ w \in V_c, \ (v, w) \notin E(G), \ d_\Gamma(v, w) = 1\}$.

Solving Annotated Instances with bounded V_c

Definition For each $A, B \subseteq V_c$ with $A \cap B = \emptyset$, let $O(A, B)$ denote the set of vertices $v \in O$ satisfying

1. $A = \{u \ : \ u \in V_c, \ d_G(u, v) = 1\}$, and
2. $B = \{w \ : \ w \in V_c, \ (v, w) \notin E(G), \ d_\Gamma(v, w) = 1\}$.

Observation

1. $\mathcal{P} = \{O(A, B) : A, B \subseteq V_c, A \cap B = \emptyset\}$ forms a partition of O.

Solving Annotated Instances with bounded V_c

Definition For each $A, B \subseteq V_c$ with $A \cap B = \emptyset$, let $O(A, B)$ denote the set of vertices $v \in O$ satisfying

1. $A = \{u \ : \ u \in V_c, \ d_G(u, v) = 1\}$, and
2. $B = \{w \ : \ w \in V_c, \ (v, w) \notin E(G), \ d_\Gamma(v, w) = 1\}$.

Observation

1. $\mathcal{P} = \{O(A, B) : A, B \subseteq V_c, A \cap B = \emptyset\}$ forms a partition of O.
2. $|\mathcal{P}| \leq 3^{|V_c|} \leq g(k, d)$ for some computable function g.

Solving Annotated Instances with bounded V_c

Observation

1. $\mathcal{P} = \{O(A, B) : A, B \subseteq V_c, A \cap B = \emptyset\}$ forms a partition of O.
2. $|\mathcal{P}| \leq 3^{|V_c|} \leq g(k, d)$ for some computable function g.

Solving Annotated Instances with bounded V_c

Observation

1. $\mathcal{P} = \{O(A, B) : A, B \subseteq V_c, A \cap B = \emptyset\}$ forms a partition of O.
2. $|\mathcal{P}| \leq 3^{|V_c|} \leq g(k, d)$ for some computable function g.

- **Marking Scheme:** For each non-empty $O(A, B) \in \mathcal{P}$, we mark an arbitrary vertex $v(A, B) \in O(A, B)$.

Solving Annotated Instances with bounded V_c

Observation

1. $\mathcal{P} = \{O(A, B) : A, B \subseteq V_c, A \cap B = \emptyset\}$ forms a partition of O.
2. $|\mathcal{P}| \leq 3^{|V_c|} \leq g(k, d)$ for some computable function g.

- **Marking Scheme:** For each non-empty $O(A, B) \in \mathcal{P}$, we mark an arbitrary vertex $v(A, B) \in O(A, B)$.

Reduction Rule 3.

Eliminate all the unmarked vertices of O from G and Γ.

Solving Annotated Instances with bounded V_c

Observation

1. $\mathcal{P} = \{O(A, B) : A, B \subseteq V_c, A \cap B = \emptyset\}$ forms a partition of O.
2. $|\mathcal{P}| \leq 3^{|V_c|} \leq g(k, d)$ for some computable function g.

- **Marking Scheme:** For each non-empty $O(A, B) \in \mathcal{P}$, we mark an arbitrary vertex $v(A, B) \in O(A, B)$.

Reduction Rule 3.

Eliminate all the unmarked vertices of O from G and Γ.

- **Smaller Instance Size:** $|V(G)|$ gets bounded by some $f(k, d)$.

Solving Annotated Instances with bounded V_c

Observation

1. $\mathcal{P} = \{O(A,B) : A, B \subseteq V_c, A \cap B = \emptyset\}$ forms a partition of O.
2. $|\mathcal{P}| \leq 3^{|V_c|} \leq g(k,d)$ for some computable function g.

- **Marking Scheme:** For each non-empty $O(A,B) \in \mathcal{P}$, we mark an arbitrary vertex $v(A,B) \in O(A,B)$.

Reduction Rule 3.

Eliminate all the unmarked vertices of O from G and Γ.

- **Smaller Instance Size:** $|V(G)|$ gets bounded by some $f(k,d)$.
- Now we **guess** end-points of the solution edges.

Final Result

Final Result

Theorem

DILATION 2-AUGMENTATION can be solved in time $f(k,d) \cdot n^{\mathcal{O}(1)}$ when G is a $\mathcal{K}_{d,d}$-free graph for any $d \in \mathbb{N}$.

Outline

Conclusion

Conclusion

- We explored Dilation 2-Augmentation in FPT setting by restricting either the graph class to which Γ or G could belong.

Conclusion

- We explored Dilation 2-Augmentation in FPT setting by restricting either the graph class to which Γ or G could belong.
- Other considerable special graph classes include intersection graphs such as interval graphs, unit-disk graphs, disk-graphs, string graphs.

Conclusion

- We explored Dilation 2-Augmentation in FPT setting by restricting either the graph class to which Γ or G could belong.
- Other considerable special graph classes include intersection graphs such as interval graphs, unit-disk graphs, disk-graphs, string graphs.
- Exploring these problems from the perspective of FPT-approximation

Thank you!