Parameterized Complexity of Maximum Edge Colorable Subgraph

A. Agrawal[1] M. Kundu[2] A. Sahu[3] S. Saurabh[3, 4] P. Tale[5]

[1]Ben Gurion University of the Negev, Israel.
[2]Indian Statistical Institute, Kolkata, India.
[3]The Institute of Mathematical Sciences, HBNI, Chennai, India.
[4]University of Bergen, Bergen, Norway.
[5]Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany.

August 25, 2020

Maximum Edge-Colorable Subgraph

Problem

Maximum Edge-Colorable Subgraph Problem
Input: A graph G, integers l and p.
Question: Find a subgraph H of G and a *p-edge coloring* of H, such that $|E(H)| \geq l$.

p-Edge Coloring of G: Function $\psi : E(G) \rightarrow \{1, 2, \ldots, p\}$ s.t. any two edges having common end point receive different colors.
Maximum Matching is a special case of Maximum Edge Colorable Subgraph when $p = 1$.

Maximum Edge-Colorable Subgraph

Problem

Maximum Edge-Colorable Subgraph Problem
Input: A graph G, integers l and p.
Question: Find a subgraph H of G and a p-*edge coloring* of H, such that $|E(H)| \geq l$.

p-*Edge Coloring* of G: Function $\psi : E(G) \rightarrow \{1, 2, \ldots, p\}$ s.t. any two edges having common end point receive different colors.

Maximum Matching is a special case of Maximum Edge Colorable Subgraph when $p = 1$.

Maximum Edge-Colorable Subgraph

Problem

Maximum Edge-Colorable Subgraph Problem
Input: A graph G, integers l and p.
Question: Find a subgraph H of G and a *p-edge coloring* of H, such that $|E(H)| \geq l$.

p-Edge Coloring of G: Function $\psi : E(G) \rightarrow \{1, 2, \ldots, p\}$ s.t. any two edges having common end point receive different colors.

Maximum Matching is a special case of Maximum Edge Colorable Subgraph when $p = 1$.

Maximum Edge-Colorable Subgraph

Problem

Maximum Edge-Colorable Subgraph Problem
Input: A graph G, integers l and p.
Question: Find a subgraph H of G and a p-*edge coloring* of H, such that $|E(H)| \geq l$.

p-*Edge Coloring* of G: Function $\psi : E(G) \rightarrow \{1, 2, \ldots, p\}$ s.t. any two edges having common end point receive different colors.
Maximum Matching is a special case of Maximum Edge Colorable Subgraph when $p = 1$.

Example

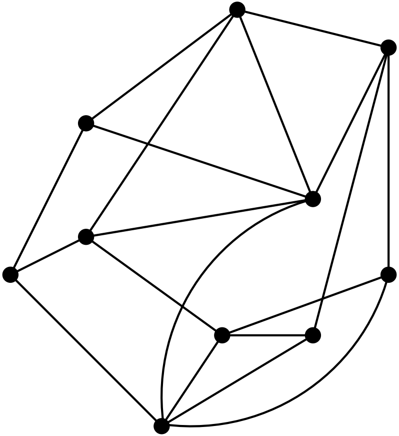

Figure: $p = 2$ and $l = 9$

Example

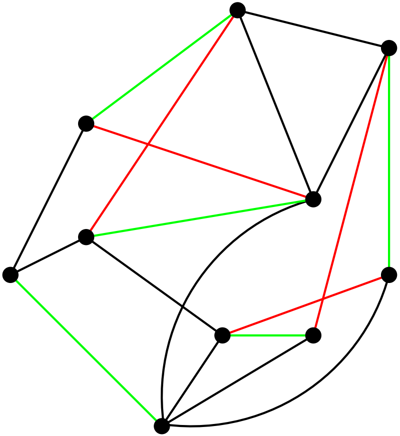

Figure: $p = 2$ and $l = 9$

Outline

- Parameterized Complexity
- Known Results
- Our Contributions
- Techniques Used
- Open Questions

Parameterized Complexity

- Goal is to find better ways of solving NP-Hard problems
- Associate (small) parameter k to each instance I
- Restrict the combinatorial explosion to a parameter k
- Parameterized problem (I, k) is *fixed-parameter tractable (FPT)* if there is an algorithm that solves the problem in time $f(k) \cdot |I|^{\mathcal{O}(1)}$.

Vertex Cover(G, k)	$\mathcal{O}(1.27^k \cdot n^2)$
Feedback Vertex Set(G, k)	$\mathcal{O}(3.46^k \cdot n^c)$
Independent Set(G, k)	No $f(k) \cdot |I|^{\mathcal{O}(1)}$

Parameterized Complexity

- Goal is to find better ways of solving NP-Hard problems
- Associate (small) parameter k to each instance I
- Restrict the combinatorial explosion to a parameter k
- Parameterized problem (I, k) is *fixed-parameter tractable (FPT)* if there is an algorithm that solves the problem in time $f(k) \cdot |I|^{\mathcal{O}(1)}$.

Vertex Cover(G, k)	$\mathcal{O}(1.27^k \cdot n^2)$
Feedback Vertex Set(G, k)	$\mathcal{O}(3.46^k \cdot n^c)$
Independent Set(G, k)	No $f(k) \cdot \lvert I \rvert^{\mathcal{O}(1)}$

Parameterized Complexity

- Goal is to find better ways of solving NP-Hard problems
- Associate (small) parameter k to each instance I
- Restrict the combinatorial explosion to a parameter k
- Parameterized problem (I, k) is *fixed-parameter tractable (FPT)* if there is an algorithm that solves the problem in time $f(k) \cdot |I|^{\mathcal{O}(1)}$.

Vertex Cover(G, k)	$\mathcal{O}(1.27^k \cdot n^2)$
Feedback Vertex Set(G, k)	$\mathcal{O}(3.46^k \cdot n^c)$
Independent Set(G, k)	No $f(k) \cdot \|I\|^{\mathcal{O}(1)}$

Parameterized Complexity

- Goal is to find better ways of solving NP-Hard problems
- Associate (small) parameter k to each instance I
- Restrict the combinatorial explosion to a parameter k
- Parameterized problem (I, k) is *fixed-parameter tractable (FPT)* if there is an algorithm that solves the problem in time $f(k) \cdot |I|^{\mathcal{O}(1)}$.

Vertex Cover(G, k)	$\mathcal{O}(1.27^k \cdot n^2)$
Feedback Vertex Set(G, k)	$\mathcal{O}(3.46^k \cdot n^c)$
Independent Set(G, k)	No $f(k) \cdot \|I\|^{\mathcal{O}(1)}$

Parameterized Complexity

- Goal is to find better ways of solving NP-Hard problems
- Associate (small) parameter k to each instance I
- Restrict the combinatorial explosion to a parameter k
- Parameterized problem (I, k) is *fixed-parameter tractable (FPT)* if there is an algorithm that solves the problem in time $f(k) \cdot |I|^{\mathcal{O}(1)}$.

Vertex Cover(G, k)	$\mathcal{O}(1.27^k \cdot n^2)$
Feedback Vertex Set(G, k)	$\mathcal{O}(3.46^k \cdot n^c)$
Independent Set(G, k)	No $f(k) \cdot \lvert I\rvert^{\mathcal{O}(1)}$

Parameterized Complexity

- Goal is to find better ways of solving NP-Hard problems
- Associate (small) parameter k to each instance I
- Restrict the combinatorial explosion to a parameter k
- Parameterized problem (I, k) is *fixed-parameter tractable (FPT)* if there is an algorithm that solves the problem in time $f(k) \cdot |I|^{\mathcal{O}(1)}$.

Vertex Cover(G, k)	$\mathcal{O}(1.27^k \cdot n^2)$
Feedback Vertex Set(G, k)	$\mathcal{O}(3.46^k \cdot n^c)$
Independent Set(G, k)	No $f(k) \cdot |I|^{\mathcal{O}(1)}$

Kernelization

- A parameterized problem admits a $h(k)$-kernel if there is a polynomial time algorithm that reduces the input instance to an equivalent smaller instance with size upper bounded by $h(k)$.
- Goal is to bound the size of instance with some function of parameter.

Vertex Cover(G, k)	$\mathcal{O}(k)$
Feedback Vertex Set(G, k)	$\mathcal{O}(k^3)$
Independent Set(G, k)	No such $h(k)$

Kernelization

- A parameterized problem admits a $h(k)$-kernel if there is a polynomial time algorithm that reduces the input instance to an equivalent smaller instance with size upper bounded by $h(k)$.
- Goal is to bound the size of instance with some function of parameter.

Vertex Cover(G, k)	$\mathcal{O}(k)$
Feedback Vertex Set(G, k)	$\mathcal{O}(k^3)$
Independent Set(G, k)	No such $h(k)$

Kernelization

- A parameterized problem admits a $h(k)$-kernel if there is a polynomial time algorithm that reduces the input instance to an equivalent smaller instance with size upper bounded by $h(k)$.
- Goal is to bound the size of instance with some function of parameter.

Vertex Cover(G, k)	$\mathcal{O}(k)$
Feedback Vertex Set(G, k)	$\mathcal{O}(k^3)$
Independent Set(G, k)	No such $h(k)$

Kernelization

- A parameterized problem admits a $h(k)$-kernel if there is a polynomial time algorithm that reduces the input instance to an equivalent smaller instance with size upper bounded by $h(k)$.
- Goal is to bound the size of instance with some function of parameter.

Vertex Cover(G, k)	$\mathcal{O}(k)$
Feedback Vertex Set(G, k)	$\mathcal{O}(k^3)$
Independent Set(G, k)	No such $h(k)$

Structural Parameterization

- Associating right parameter is an art.
- Obvious choice : solution size (in this case l)
- Structural Parameters:
  - *Vertex cover number, $vc(G)$*
    minimum number of vertices needed to be deleted to obtain independent set.
  - $l - mm(G)$
    $\mathtt{mm}(G)$ is the size of a maximum matching in G.
  - *Deg-1-modulator*
    For a graph G, a set $X \subseteq V(G)$ is a *deg-1-modulator* of G, if the degree of each vertex in $G - X$ is at most 1.

Structural Parameterization

- Associating right parameter is an art.
- Obvious choice : solution size (in this case l)
- Structural Parameters:
  - *Vertex cover number,* $vc(G)$
    minimum number of vertices needed to be deleted to obtain independent set.
  - $l - mm(G)$
    $\mathtt{mm}(G)$ is the size of a maximum matching in G.
  - *Deg*-1-*modulator*
    For a graph G, a set $X \subseteq V(G)$ is a *deg*-1-*modulator* of G, if the degree of each vertex in $G - X$ is at most 1.

Structural Parameterization

- Associating right parameter is an art.
- Obvious choice : solution size (in this case l)
- Structural Parameters:
  - *Vertex cover number, $vc(G)$*
    minimum number of vertices needed to be deleted to obtain independent set.
  - $l - mm(G)$
    $\mathtt{mm}(G)$ is the size of a maximum matching in G.
  - *Deg-1-modulator*
    For a graph G, a set $X \subseteq V(G)$ is a *deg-1-modulator* of G, if the degree of each vertex in $G - X$ is at most 1.

Structural Parameterization

- Associating right parameter is an art.
- Obvious choice : solution size (in this case l)
- Structural Parameters:
  - *Vertex cover number, $vc(G)$*
    minimum number of vertices needed to be deleted to obtain independent set.
  - $l - mm(G)$
    $\mathtt{mm}(G)$ is the size of a maximum matching in G.
  - *Deg-1-modulator*
    For a graph G, a set $X \subseteq V(G)$ is a *deg-1-modulator* of G, if the degree of each vertex in $G - X$ is at most 1.

Structural Parameterization

- Associating right parameter is an art.
- Obvious choice : solution size (in this case l)
- Structural Parameters:
  - *Vertex cover number,* $vc(G)$
    minimum number of vertices needed to be deleted to obtain independent set.
  - $l - mm(G)$
    $\mathtt{mm}(G)$ is the size of a maximum matching in G.
  - *Deg*-1-*modulator*
    For a graph G, a set $X \subseteq V(G)$ is a *deg*-1-*modulator* of G, if the degree of each vertex in $G - X$ is at most 1.

Structural Parameterization

- Associating right parameter is an art.
- Obvious choice : solution size (in this case l)
- Structural Parameters:
  - *Vertex cover number,* $vc(G)$
    minimum number of vertices needed to be deleted to obtain independent set.
  - $l - mm(G)$
    $\texttt{mm}(G)$ is the size of a maximum matching in G.
  - *Deg*-1-*modulator*
    For a graph G, a set $X \subseteq V(G)$ is a *deg*-1-*modulator* of G, if the degree of each vertex in $G - X$ is at most 1.

Structural Parameterization

- Associating right parameter is an art.
- Obvious choice : solution size (in this case l)
- Structural Parameters:
  - *Vertex cover number,* $vc(G)$
    minimum number of vertices needed to be deleted to obtain independent set.
  - $l - mm(G)$
    $\mathtt{mm}(G)$ is the size of a maximum matching in G.
  - *Deg-*1*-modulator*
    For a graph G, a set $X \subseteq V(G)$ is a *deg-*1*-modulator* of G, if the degree of each vertex in $G - X$ is at most 1.

Problem Statement with parameters

Problem

Maximum Edge-Colorable Subgraph Problem

Input: A graph G, integers l and p.

Parameters: k where k is

- The solution size(l)
- vertex cover number, $\texttt{vc}(G)$
- $l - \texttt{mm}(G)$

Question: Find a subgraph H of G and a p-edge-coloring of H, such that $|E(H)| \geq l$.

Known Results

- Vizing Theorem : $\Delta(G) \leq \chi'(G) \leq \Delta(G) + 1$
  - $\Delta(G)$ is the maximum degree of a vertex in G
  - $\chi'(G)$ is the smallest integer p for which G is p-edge colorable (also called *chromatic index*)
- Holyer (1981): Deciding chromatic index of G is $\Delta(G)$ or $\Delta(G) + 1$ is NP-Hard (even for cubic graphs).
- Feige (2002): Maximum Edge Colorable Subgraph is NP-Hard even for $p = 2$.

Known Results

- Vizing Theorem : $\Delta(G) \leq \chi'(G) \leq \Delta(G) + 1$
  - $\Delta(G)$ is the maximum degree of a vertex in G
  - $\chi'(G)$ is the smallest integer p for which G is p-edge colorable (also called *chromatic index*)
- Holyer (1981): Deciding chromatic index of G is $\Delta(G)$ or $\Delta(G) + 1$ is NP-Hard (even for cubic graphs).
- Feige (2002): Maximum Edge Colorable Subgraph is NP-Hard even for $p = 2$.

Known Results

- Vizing Theorem : $\Delta(G) \leq \chi'(G) \leq \Delta(G) + 1$
  - $\Delta(G)$ is the maximum degree of a vertex in G
  - $\chi'(G)$ is the smallest integer p for which G is p-edge colorable (also called *chromatic index*)
- Holyer (1981): Deciding chromatic index of G is $\Delta(G)$ or $\Delta(G) + 1$ is NP-Hard (even for cubic graphs).
- Feige (2002): Maximum Edge Colorable Subgraph is NP-Hard even for $p = 2$.

Known Results

- Feige (2002): A constant factor approximation algorithm. NP-Hard to get $(1 - \epsilon)$-approximation algorithm for every fixed $p \geq 2$.
- Aloisioa and Mkrtchyana (2019): The problem is FPT w.r.t. structural graph parameters like path-width, curving-width when $p = 2$.

Known Results

- Feige (2002): A constant factor approximation algorithm. NP-Hard to get $(1-\epsilon)$-approximation algorithm for every fixed $p \geq 2$.
- Aloisioa and Mkrtchyana (2019): The problem is FPT w.r.t. structural graph parameters like path-width, curving-width when $p = 2$.

Known Results

- Feige (2002): A constant factor approximation algorithm. NP-Hard to get $(1-\epsilon)$-approximation algorithm for every fixed $p \geq 2$.
- Aloisioa and Mkrtchyana (2019): The problem is FPT w.r.t. structural graph parameters like path-width, curving-width when $p = 2$.

Known Results

- Grüttemeier (2020): Obtained kernels, when the parameter is $p + k$, where k is one of the following:
  - the number of edges that needs to be deleted from G, to obtain a graph with maximum degree at most $p - 1$
  - the deletion set size to a graph whose connected components have at most p vertices
- Galby (2019): Proved Edge Coloring is FPT when parameterized by the number of colors,(p) and the number of vertices having the maximum degree.

Known Results

- Grüttemeier (2020): Obtained kernels, when the parameter is $p + k$, where k is one of the following:
  - the number of edges that needs to be deleted from G, to obtain a graph with maximum degree at most $p - 1$
  - the deletion set size to a graph whose connected components have at most p vertices
- Galby (2019): Proved Edge Coloring is FPT when parameterized by the number of colors,(p) and the number of vertices having the maximum degree.

Known Results

- Grüttemeier (2020): Obtained kernels, when the parameter is $p + k$, where k is one of the following:
  - the number of edges that needs to be deleted from G, to obtain a graph with maximum degree at most $p - 1$
  - the deletion set size to a graph whose connected components have at most p vertices
- Galby (2019): Proved Edge Coloring is FPT when parameterized by the number of colors,(p) and the number of vertices having the maximum degree.

Our Contribution

We denote $k_1 \preceq k_2$ and say k_1 *is smaller than* k_2 if there exists a computable function $g(\cdot)$ such that $k_1 \leq g(k_2)$.

Observation

For a given instance (G, l, p) of Maximum Edge Colorable Subgraph, in polynomial time, we can conclude that either (G, l, p) is a Yes instance or $\texttt{vc}(G) \preceq l$ and $|X| \preceq (l - \texttt{mm}(G))$, where X is a minimum sized deg-1-modulator of G.

Our Contribution

Theorem

Maximum Edge Colorable Subgraph, parameterized by the vertex cover number, $\mathtt{vc}(\mathsf{G})$ of G, is FPT.

Our Contribution

Theorem

There exists a deterministic algorithm $\mathcal{A}$ and a randomized algorithm $\mathcal{B}$ with constant probability of success that solves Maximum Edge Colorable Subgraph. For a given instance (G, l, p), Algorithms $\mathcal{A}$ and $\mathcal{B}$ terminate in time $\mathcal{O}^*(4^{l+o(l)})$ and $\mathcal{O}^*(2^l)$, respectively.

Our Contribution

Theorem

Maximum Edge Colorable Subgraph admits a kernel with $\mathcal{O}(kp)$ vertices, for every $k \in \{\ell, \texttt{vc}(G), l - \texttt{mm}(G)\}$.

Theorem

For any $k \in \{\ell, \texttt{vc}(G), l - \texttt{mm}(G)\}$, Maximum Edge Colorable Subgraph does not admit a compression of size $\mathcal{O}(k^{1-\epsilon} \cdot f(p))$, for any $\epsilon > 0$ and computable function f, unless NP $\subseteq$ coNP/poly.

Our Contribution

Theorem

Maximum Edge Colorable Subgraph admits a kernel with $\mathcal{O}(kp)$ vertices, for every $k \in \{\ell, \texttt{vc}(G), l - \texttt{mm}(G)\}$.

Theorem

For any $k \in \{\ell, \texttt{vc}(G), l - \texttt{mm}(G)\}$, Maximum Edge Colorable Subgraph does not admit a compression of size $\mathcal{O}(k^{1-\epsilon} \cdot f(p))$, for any $\epsilon > 0$ and computable function f, unless NP $\subseteq$ coNP/poly.

Techniques Used for FPT Paremeterized by vc(G)

- For a given instance, our algorithm creates $f(\mathrm{vc}(G))$-many instances of ILP with nr. of variable bounded by $h(\mathrm{vc}(G))$.
- Kannan, Lenstra: ILP is FPT para. by nr. of variable
- For the instance (G, l, p), let (H, ϕ) is the solution
- X be a vertex cover of G.
- We "guess" $H' = H[X]$ and $\phi' = \phi|_{E(H')}$.
- Use ILP to find extension of this guess.

Techniques Used for FPT Paremeterized by vc(G)

- For a given instance, our algorithm creates $f(\texttt{vc}(G))$-many instances of ILP with nr. of variable bounded by $h(\texttt{vc}(G))$.
- Kannan, Lenstra: ILP is FPT para. by nr. of variable
- For the instance (G, l, p), let (H, ϕ) is the solution
- X be a vertex cover of G.
- We "guess" $H' = H[X]$ and $\phi' = \phi|_{E(H')}$.
- Use ILP to find extension of this guess.

TECHNIQUES USED FOR FPT PAREMETERIZED BY vc(G)

- For a given instance, our algorithm creates $f(\mathtt{vc}(G))$-many instances of ILP with nr. of variable bounded by $h(\mathtt{vc}(G))$.
- Kannan, Lenstra: ILP is FPT para. by nr. of variable
- For the instance (G, l, p), let (H, ϕ) is the solution
- X be a vertex cover of G.
- We "guess" $H' = H[X]$ and $\phi' = \phi|_{E(H')}$.
- Use ILP to find extension of this guess.

Techniques Used for FPT Paremeterized by vc(G)

- For a given instance, our algorithm creates $f(\mathtt{vc}(G))$-many instances of ILP with nr. of variable bounded by $h(\mathtt{vc}(G))$.
- Kannan, Lenstra: ILP is FPT para. by nr. of variable
- For the instance (G, l, p), let (H, ϕ) is the solution
- X be a vertex cover of G.
- We "guess" $H' = H[X]$ and $\phi' = \phi|_{E(H')}$.
- Use ILP to find extension of this guess.

Techniques Used for FPT Paremeterized by vc(G)

- For a given instance, our algorithm creates $f(\mathtt{vc}(G))$-many instances of ILP with nr. of variable bounded by $h(\mathtt{vc}(G))$.
- Kannan, Lenstra: ILP is FPT para. by nr. of variable
- For the instance (G, l, p), let (H, ϕ) is the solution
- X be a vertex cover of G.
- We "guess" $H' = H[X]$ and $\phi' = \phi|_{E(H')}$.
- Use ILP to find extension of this guess.

Techniques Used for FPT Paremeterized by vc(G)

- For a given instance, our algorithm creates $f(\mathtt{vc}(G))$-many instances of ILP with nr. of variable bounded by $h(\mathtt{vc}(G))$.
- Kannan, Lenstra: ILP is FPT para. by nr. of variable
- For the instance (G, l, p), let (H, ϕ) is the solution
- X be a vertex cover of G.
- We "guess" $H' = H[X]$ and $\phi' = \phi|_{E(H')}$.
- Use ILP to find extension of this guess.

Techniques Used for FPT Paremeterized by vc(G)

- For a given instance, our algorithm creates $f(\mathtt{vc}(G))$-many instances of ILP with nr. of variable bounded by $h(\mathtt{vc}(G))$.
- Kannan, Lenstra: ILP is FPT para. by nr. of variable
- For the instance (G, l, p), let (H, ϕ) is the solution
- X be a vertex cover of G.
- We "guess" $H' = H[X]$ and $\phi' = \phi|_{E(H')}$.
- Use ILP to find extension of this guess.

Techniques Used for Deterministic and Randomized Algorithms

The Algorithms $\mathcal{A}$(deterministic algorithm) and $\mathcal{B}$(randomized algorithm) use different sets of ideas.

- Algorithm $\mathcal{A}$, uses a combination of the technique of Color-Coding and Divide and Color.
- Algorithm $\mathcal{B}$ uses the algorithm to solve Rainbow Matching as a black-box.

Techniques Used for Deterministic and Randomized Algorithms

The Algorithms $\mathcal{A}$(deterministic algorithm) and $\mathcal{B}$(randomized algorithm) use different sets of ideas.

- Algorithm $\mathcal{A}$, uses a combination of the technique of Color-Coding and Divide and Color.
- Algorithm $\mathcal{B}$ uses the algorithm to solve Rainbow Matching as a black-box.

Techniques Used for Deterministic and Randomized Algorithms

The Algorithms $\mathcal{A}$(deterministic algorithm) and $\mathcal{B}$(randomized algorithm) use different sets of ideas.

- Algorithm $\mathcal{A}$, uses a combination of the technique of Color-Coding and Divide and Color.
- Algorithm $\mathcal{B}$ uses the algorithm to solve Rainbow Matching as a black-box.

Techniques Used for Deterministic and Randomized Algorithms

The Algorithms $\mathcal{A}$(deterministic algorithm) and $\mathcal{B}$(randomized algorithm) use different sets of ideas.

- Algorithm $\mathcal{A}$, uses a combination of the technique of Color-Coding and Divide and Color.
- Algorithm $\mathcal{B}$ uses the algorithm to solve Rainbow Matching as a black-box.

Techniques Used for Kernel

The technique that is used to find kernel is Expansion Lemma

t-**expansion** *of* P *into* Q is a set of edges $M \subseteq E(G)$ if (i) every vertex of P is incident with exactly t edges of M, and (ii) the number of vertices in Q which are incident with at least one edge in M is exactly $t|P|$.

Lemma: Expansion Lemma

Let t be a positive integer and G be a bipartite graph with vertex bipartition (P, Q) such that $|Q| \geq t|P|$ and there are no isolated vertices in Q. Then there exist nonempty vertex sets $P' \subseteq P$ and $Q' \subseteq Q$ such that

- P' has a t-expansion into Q'
- no vertex in Q' has a neighbour outside P'.

Furthermore two such sets P' and Q' can be found in time polynomial in the size of G.

Techniques Used for Kernel

The technique that is used to find kernel is Expansion Lemma t-**expansion** *of* P *into* Q is a set of edges $M \subseteq E(G)$ if (i) every vertex of P is incident with exactly t edges of M, and (ii) the number of vertices in Q which are incident with at least one edge in M is exactly $t|P|$.

Lemma: Expansion Lemma

Let t be a positive integer and G be a bipartite graph with vertex bipartition (P, Q) such that $|Q| \geq t|P|$ and there are no isolated vertices in Q. Then there exist nonempty vertex sets $P' \subseteq P$ and $Q' \subseteq Q$ such that

- P' has a t-expansion into Q'
- no vertex in Q' has a neighbour outside P'.

Furthermore two such sets P' and Q' can be found in time polynomial in the size of G.

Techniques Used for Kernel

The technique that is used to find kernel is Expansion Lemma
t-**expansion** *of* P *into* Q is a set of edges $M \subseteq E(G)$ if (i) every vertex of P is incident with exactly t edges of M, and (ii) the number of vertices in Q which are incident with at least one edge in M is exactly $t|P|$.

Lemma: Expansion Lemma

Let t be a positive integer and G be a bipartite graph with vertex bipartition (P, Q) such that $|Q| \geq t|P|$ and there are no isolated vertices in Q. Then there exist nonempty vertex sets $P' \subseteq P$ and $Q' \subseteq Q$ such that

- P' has a t-expansion into Q'
- no vertex in Q' has a neighbour outside P'.

Furthermore two such sets P' and Q' can be found in time polynomial in the size of G.

Techniques Used: For kernel lower bounds

- Reduction from Red Blue Dominating Set (RBDS)
- Given instance (G, R, B, k) of RBDS reduction produce (G', ℓ, p) of Maximum Edge Colorable Subgraph s.t. $\ell = |E(G)| - k$ and p is a constant.
- Jansen and Pieterse (2015): RBDS does not admit a compression of size $\mathcal{O}(n^{2-\epsilon})$.
- This implies the desired lower bound.

Techniques Used: For kernel lower bounds

- Reduction from Red Blue Dominating Set (RBDS)
- Given instance (G, R, B, k) of RBDS reduction produce (G', ℓ, p) of Maximum Edge Colorable Subgraph s.t. $\ell = |E(G)| - k$ and p is a constant.
- Jansen and Pieterse (2015): RBDS does not admit a compression of size $\mathcal{O}(n^{2-\epsilon})$.
- This implies the desired lower bound.

Techniques Used: For kernel lower bounds

- Reduction from Red Blue Dominating Set (RBDS)
- Given instance (G, R, B, k) of RBDS reduction produce (G', ℓ, p) of Maximum Edge Colorable Subgraph s.t. $\ell = |E(G)| - k$ and p is a constant.
- Jansen and Pieterse (2015): RBDS does not admit a compression of size $\mathcal{O}(n^{2-\epsilon})$.
- This implies the desired lower bound.

Techniques Used: For kernel lower bounds

- Reduction from Red Blue Dominating Set (RBDS)
- Given instance (G, R, B, k) of RBDS reduction produce (G', ℓ, p) of Maximum Edge Colorable Subgraph s.t. $\ell = |E(G)| - k$ and p is a constant.
- Jansen and Pieterse (2015): RBDS does not admit a compression of size $\mathcal{O}(n^{2-\epsilon})$.
- This implies the desired lower bound.

Techniques Used: For kernel lower bounds

- Reduction from Red Blue Dominating Set (RBDS)
- Given instance (G, R, B, k) of RBDS reduction produce (G', ℓ, p) of Maximum Edge Colorable Subgraph s.t. $\ell = |E(G)| - k$ and p is a constant.
- Jansen and Pieterse (2015): RBDS does not admit a compression of size $\mathcal{O}(n^{2-\epsilon})$.
- This implies the desired lower bound.

Open Questions

We showed –

- Kernel with $\mathcal{O}(kp)$ vertices for $k \in \{l, \texttt{vc}, l - \texttt{mm}(G)\}$
- No kernel of size $\mathcal{O}(k^{1-\epsilon} f(p))$.

Q Can we bridge the gap?

Problem is NP-Hard on regular graphs of degree $p + 1$.
For such graphs classes, kernel lower bound is tight.

Open Questions

We showed –

- FPT when para. by $\mathtt{vc}$
- FPT when para. by ℓ
- Admits a polynomial kernel when para. by ℓ

Q Is it FPT when para. by $\ell - \mathtt{mm}(G)$?

Q Does it admits a polynomial kernel when parameterized by $\mathtt{vc}$?

Thank You!