Kernelization of MAXIMUM MINIMAL VERTEX COVER

Ignasi Sau

LIRMM, Université de Montpellier, CNRS, Montpellier, France

Júlio Araújo Univ. Federal do Ceará, Fortaleza, Brazil
Marin Bougeret LIRMM, Université de Montpellier, France
Victor A. Campos Univ. Federal do Ceará, Fortaleza, Brazil

[Article available at arXiv:2102.02484]

Séminaire AlGCo
March 25h, 2021

Outline of the talk

Next section is...

For a minimization problem, it is natural to consider its "maximum minimal" version: worst-case of a greedy heuristic.

For a minimization problem, it is natural to consider its "maximum minimal" version: worst-case of a greedy heuristic.

- MAXIMUM MINIMAL DOMINATING SET: UPPER DOMINATION.
- MAXIMUM MINIMAL HITTING SET.
- MAXIMUM MINIMAL FEEDBACK VERTEX SET.

For a minimization problem, it is natural to consider its "maximum minimal" version: worst-case of a greedy heuristic.

In this talk:

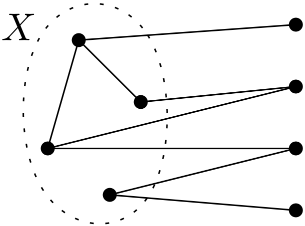
Maximum Minimal Vertex Cover (MMVC)
Input: A graph G and an integer k.
Question: Does G contain a minimal vertex cover of size at least k?

For a minimization problem, it is natural to consider its "maximum minimal" version: worst-case of a greedy heuristic.

In this talk:

Maximum Minimal Vertex Cover (MMVC)
Input: A graph G and an integer k.
Question: Does G contain a minimal vertex cover of size at least k?

Dual problem of MMVC: Minimum Independent Dominating Set.

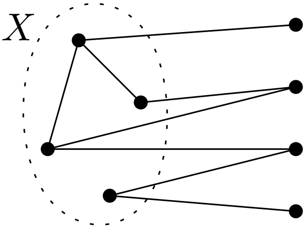


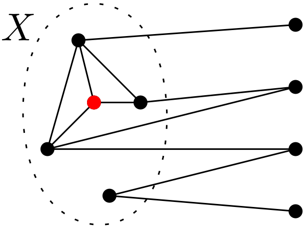
For a minimization problem, it is natural to consider its "maximum minimal" version: worst-case of a greedy heuristic.

In this talk:

MAXIMUM MINIMAL VERTEX COVER (MMVC)	
Input:	A graph G and an integer k.
Question:	Does G contain a minimal vertex cover of size at least k?

Dual problem of MMVC: MINIMUM INDEPENDENT DOMINATING SET.





A set $X \subseteq V(G)$ is a minimal vertex cover of G $\Leftrightarrow$
X is a vertex cover of G and, for every vertex $v \in X$, $N(v) \not\subseteq X$.

Previous work

- FPT algorithms and general remarks. [Fernau. 2005]

Previous work

- FPT algorithms and general remarks. [Fernau. 2005]
- FPT algorithms (solution size, treewidth, size of a min. vertex cover), $n^{1/2}$-approximation, and $n^{1/2-\varepsilon}$-inapproximability. [Boria, Della Croce, Paschos. 2015]

Previous work

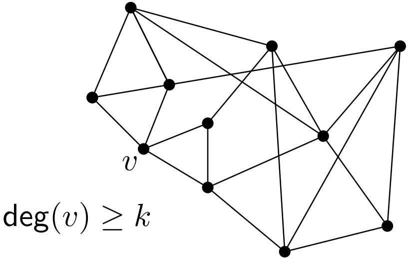
- FPT algorithms and general remarks. [Fernau. 2005]
- FPT algorithms (solution size, treewidth, size of a min. vertex cover), $n^{1/2}$-approximation, and $n^{1/2-\varepsilon}$-inapproximability. [Boria, Della Croce, Paschos. 2015]
- Tight FPT algorithms (weighted version) parameterized by the size of a minimum vertex cover. [Zehavi. 2017]

Previous work

- FPT algorithms and general remarks. [Fernau. 2005]
- FPT algorithms (solution size, treewidth, size of a min. vertex cover), $n^{1/2}$-approximation, and $n^{1/2-\varepsilon}$-inapproximability. [Boria, Della Croce, Paschos. 2015]
- Tight FPT algorithms (weighted version) parameterized by the size of a minimum vertex cover. [Zehavi. 2017]
- Inapproximability of MMVC in subexponential time. [Bonnet, Paschos. 2018] [Bonnet, Lampis, Paschos. 2018]

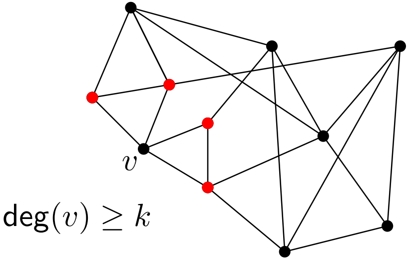
What about the kernelization of MMVC?

What about the kernelization of MMVC?



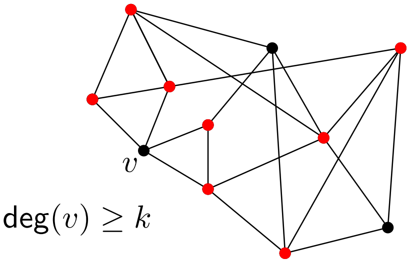
If there exists v with $\deg(v) \geq k$

What about the kernelization of MMVC?



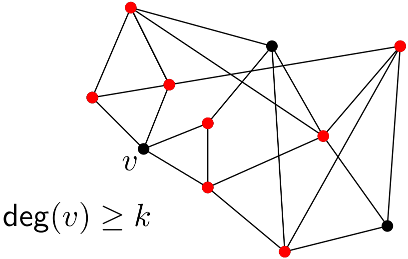
If there exists v with $\deg(v) \geq k$

What about the kernelization of MMVC?



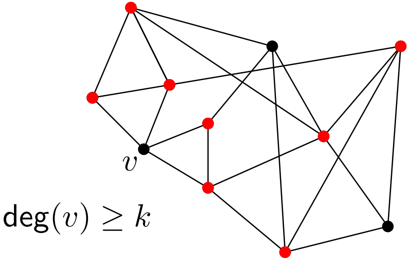
If there exists v with $\deg(v) \geq k$

What about the kernelization of MMVC?



If there exists v with $\deg(v) \geq k$ $\Rightarrow$ we have a YES-instance.

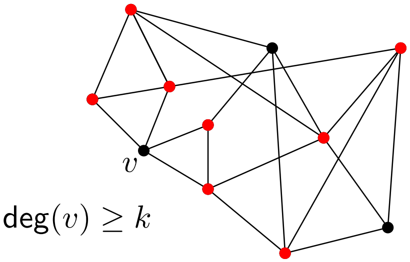
What about the kernelization of MMVC?



If there exists v with $\deg(v) \geq k \;\Rightarrow\;$ we have a YES-instance.

Thus, we may assume that $\Delta(G) \leq k - 1$.

What about the kernelization of MMVC?

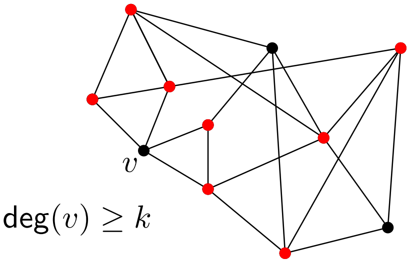


If there exists v with $\deg(v) \geq k \;\Rightarrow\;$ we have a YES-instance.

Thus, we may assume that $\Delta(G) \leq k - 1$.

Any vertex cover X of G covers at most $|X| \cdot (k-1)$ edges.

What about the kernelization of MMVC?



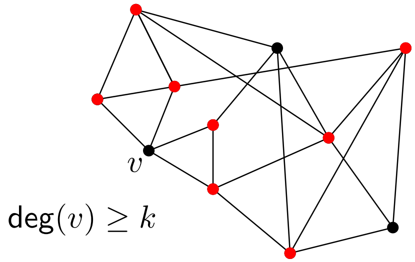
If there exists v with $\deg(v) \geq k \;\Rightarrow\;$ we have a YES-instance.

Thus, we may assume that $\Delta(G) \leq k - 1$.

Any vertex cover X of G covers at most $|X| \cdot (k-1)$ edges.

By removing isolated vertices, it follows that $|V(G)| \leq |X| \cdot k$.

What about the kernelization of MMVC?



If there exists v with $\deg(v) \geq k \Rightarrow$ we have a YES-instance.

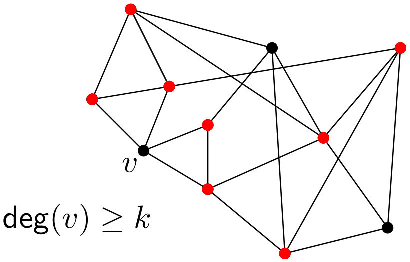
Thus, we may assume that $\Delta(G) \leq k - 1$.

Any vertex cover X of G covers at most $|X| \cdot (k - 1)$ edges.

By removing isolated vertices, it follows that $|V(G)| \leq |X| \cdot k$.

If $|V(G)| \geq k^2 \Rightarrow$ we have a YES-instance.

What about the kernelization of MMVC?



If there exists v with $\deg(v) \geq k$ $\Rightarrow$ we have a YES-instance.

Thus, we may assume that $\Delta(G) \leq k-1$.

Any vertex cover X of G covers at most $|X| \cdot (k-1)$ edges.

By removing isolated vertices, it follows that $|V(G)| \leq |X| \cdot k$.

If $|V(G)| \geq k^2$ $\Rightarrow$ we have a YES-instance.

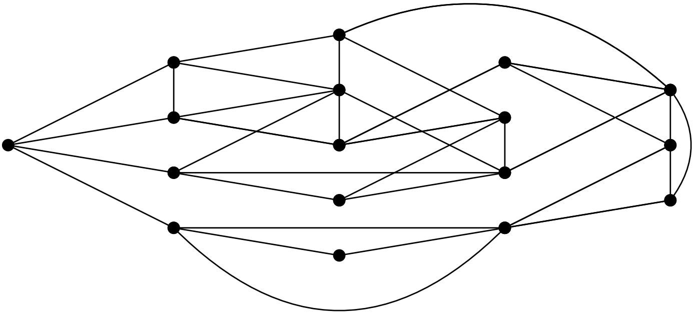
Thus, we trivially have a kernel with $|V(G)| < k^2$.

A linear kernel

Strategy to obtain a linear kernel: [Fernau. 2005]

A linear kernel

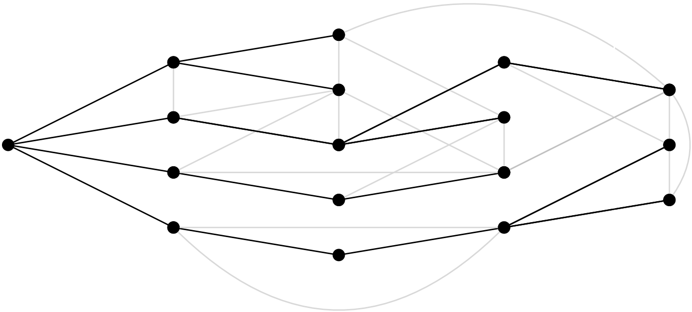
Strategy to obtain a linear kernel: [Fernau. 2005]



Given a graph G and a parameter k.

A linear kernel

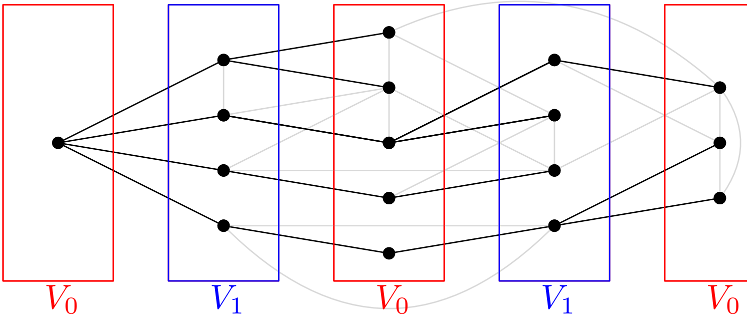
Strategy to obtain a linear kernel: [Fernau. 2005]



Consider an arbitrary spanning tree T and root it at a vertex r.

A linear kernel

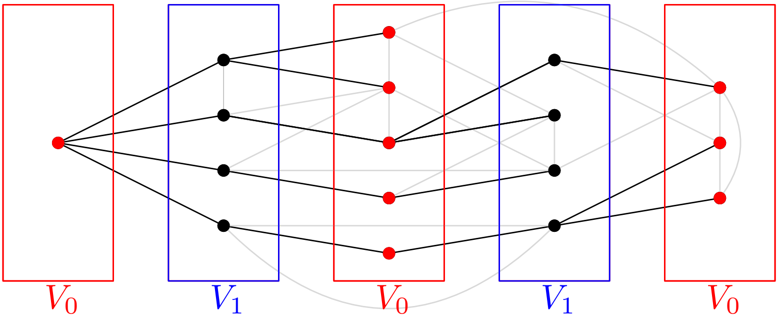
Strategy to obtain a linear kernel: [Fernau. 2005]



Partition $V(G)$ into V_0 and V_1 according to the distance from r.

A linear kernel

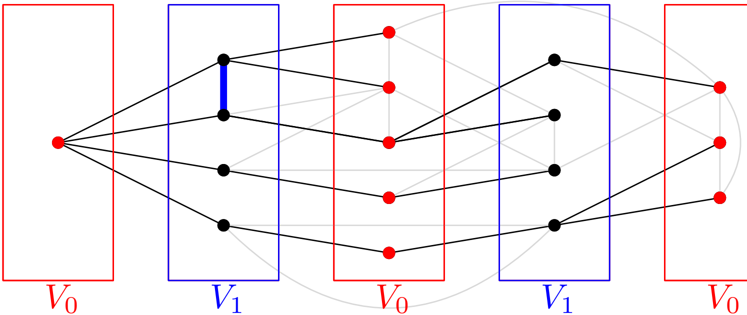
Strategy to obtain a linear kernel: [Fernau. 2005]



Suppose w.l.o.g. that V_0 is the largest of the two sets.

A linear kernel

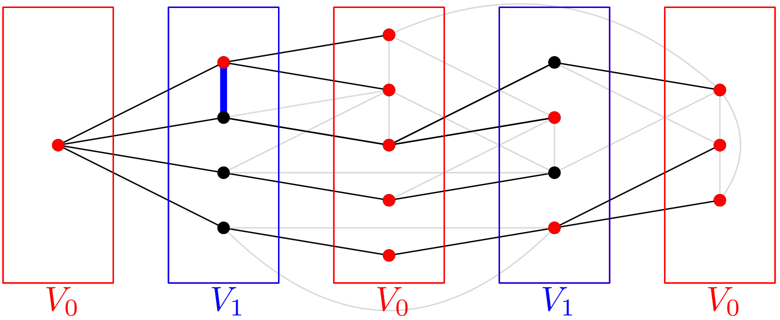
Strategy to obtain a linear kernel: [Fernau. 2005]



Suppose w.l.o.g. that V_0 is the largest of the two sets.

A linear kernel

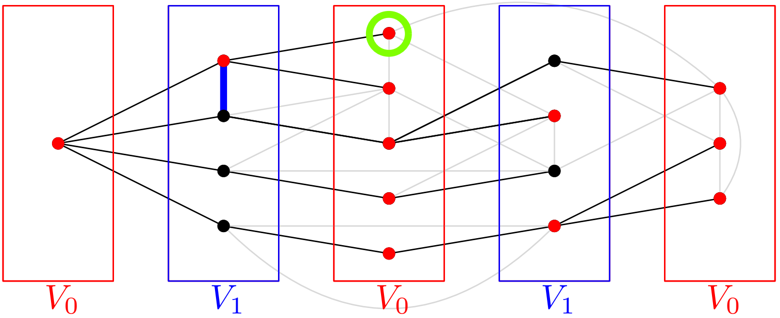
Strategy to obtain a linear kernel: [Fernau. 2005]



Extend V_0 into a minimal vertex cover of G: kernel of size $2k$.

A linear kernel

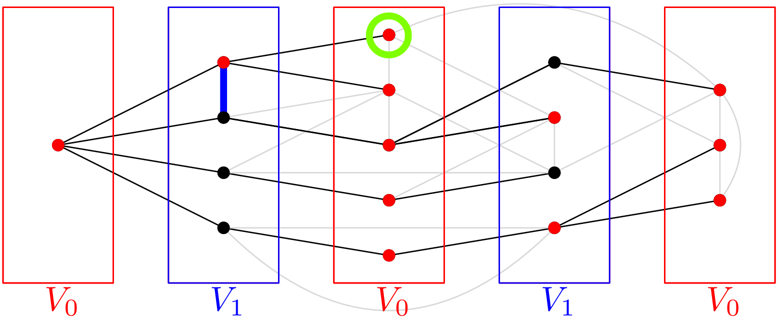
Strategy to obtain a linear kernel: [Fernau. 2005]



Extend V_0 into a minimal vertex cover of G: kernel of size $2k$.

A linear kernel ...that doesn't work!

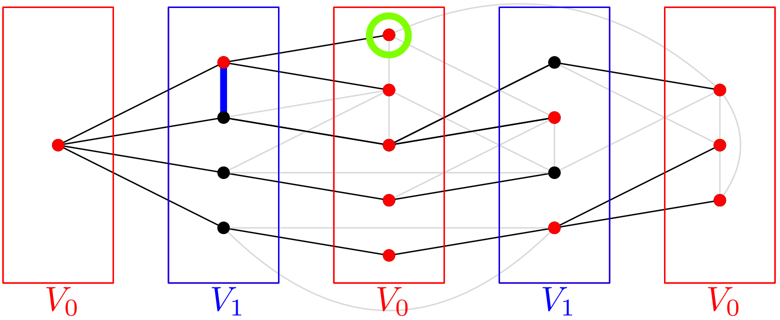
Strategy to obtain a linear kernel: [Fernau. 2005]



Problem: when extending V_0, we may lose vertices!

A linear kernel ...that doesn't work!

Strategy to obtain a linear kernel: [Fernau. 2005]

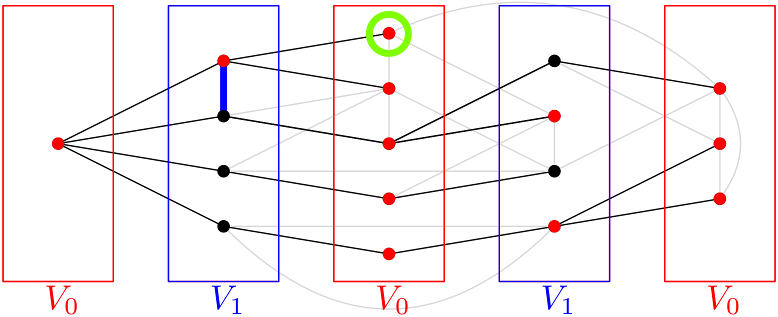


Problem: when extending V_0, we may lose vertices!

Deciding whether $S \subseteq V(G)$ can the extended to a minimal vertex cover of G is NP-complete. [Casel, Fernau, Ghadikolaei, Monnot, Sikora. 2019]

A linear kernel ...that doesn't work!

Strategy to obtain a linear kernel: [Fernau. 2005]



Problem: when extending V_0, we may lose vertices!

Deciding whether $S \subseteq V(G)$ can the extended to a minimal vertex cover of G is NP-complete. [Casel, Fernau, Ghadikolaei, Monnot, Sikora. 2019]

The existence of a kernel with $o(k^2)$ vertices has been asked by [Boria, Della Croce, Paschos. 2015]

Next section is...

What about smaller parameters?

Do polynomial kernels exist for parameters smaller than the solution size?

What about smaller parameters?

Do polynomial kernels exist for parameters smaller than the solution size?

Natural candidate: size of a minimum vertex cover.

What about smaller parameters?

Do polynomial kernels exist for parameters smaller than the solution size?

Natural candidate: size of a minimum vertex cover.

Theorem

The Maximum Minimal Vertex Cover *problem parameterized by the size of a minimum vertex cover of the input graph does not admit a polynomial kernel unless* $\mathsf{NP} \subseteq \mathsf{coNP}/\mathsf{poly}$, *even on bipartite graphs.*

What about smaller parameters?

Do polynomial kernels exist for parameters smaller than the solution size?

Natural candidate: size of a minimum vertex cover.

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by the size of a minimum vertex cover of the input graph does not admit a polynomial kernel unless* NP ⊆ coNP/poly, *even on bipartite graphs.*

This complements the FPT algorithms for MMVC with this parameter.

[Boria, Della Croce, Paschos. 2015]

[Zehavi. 2017]

What about smaller parameters?

Do polynomial kernels exist for parameters smaller than the solution size?

Natural candidate: size of a minimum vertex cover.

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by the size of a minimum vertex cover of the input graph does not admit a polynomial kernel unless* NP $\subseteq$ coNP/poly, *even on bipartite graphs.*

This complements the FPT algorithms for MMVC with this parameter.

[Boria, Della Croce, Paschos. 2015]

[Zehavi. 2017]

Our result rules out the existence of polynomial kernels for MMVC parameterized by treewidth as well.

Can a subquadratic kernel for MMVC exist?

From now on, we consider the solution size k as the parameter.

Can a subquadratic kernel for MMVC exist?

From now on, we consider the solution size k as the parameter.

Question Does MMVC admit a kernel of size $\mathcal{O}(k^{2-\varepsilon})$?

Can a subquadratic kernel for MMVC exist?

From now on, we consider the solution size k as the parameter.

Question Does MMVC admit a kernel of size $\mathcal{O}(k^{2-\varepsilon})$?

We introduce a new framework to provide kernelization lower bounds.

Can a subquadratic kernel for MMVC exist?

From now on, we consider the solution size k as the parameter.

Question Does MMVC admit a kernel of size $\mathcal{O}(k^{2-\varepsilon})$?

We introduce a new framework to provide kernelization lower bounds.

We consider a general vertex-maximization problem Π, parameterized by the solution size k.

Can a subquadratic kernel for MMVC exist?

From now on, we consider the solution size k as the parameter.

Question Does MMVC admit a kernel of size $\mathcal{O}(k^{2-\varepsilon})$?

We introduce a new framework to provide kernelization lower bounds.

We consider a general vertex-maximization problem Π,
parameterized by the solution size k.

(The definitions can be adapted to vertex-minimization problems as well.)

A kernel for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \to \mathbb{N}$, called the size of the kernel, s.t.

1. (G', k') Yes-instance $\Rightarrow$ (G, k) Yes-instance.
2. (G, k) Yes-instance $\Rightarrow$ (G', k') Yes-instance.

A kernel for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \rightarrow \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_{\Pi}(G') \geq k' \Rightarrow \text{opt}_{\Pi}(G) \geq k$.
2. $\text{opt}_{\Pi}(G) \geq k \Rightarrow \text{opt}_{\Pi}(G') \geq k'$.

A kernel for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \to \mathbb{N}$, called the size of the kernel, s.t.

1. $\mathrm{opt}_\Pi(G') \geq k' \Rightarrow \mathrm{opt}_\Pi(G) \geq k$.
2. $\mathrm{opt}_\Pi(G) \geq k \Rightarrow \mathrm{opt}_\Pi(G') \geq k'$.

Slight restriction: "large optimal preserving" kernel, or lop-kernel for short:

A kernel for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \to \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq k'$.

Slight restriction: "large optimal preserving" kernel, or lop-kernel for short:

A lop-kernel for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \to \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq \text{opt}_\Pi(G) - (k - k')$

A **kernel** for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \rightarrow \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq k'$.

Slight restriction: "large optimal preserving" kernel, or lop-kernel for short:

A **lop-kernel** for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \rightarrow \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq \text{opt}_\Pi(G) - (k - k') \Rightarrow \text{opt}_\Pi(G') \geq k'$.

A **kernel** for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \to \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq k'$.

Slight restriction: "large optimal preserving" kernel, or lop-kernel for short:

A **lop-kernel** for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \to \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq \text{opt}_\Pi(G) - (k - k') \Rightarrow \text{opt}_\Pi(G') \geq k'$.

We call a reduction rule as above a lop-rule.

A **kernel** for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \rightarrow \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq k'$.

Slight restriction: "large optimal preserving" kernel, or lop-kernel for short:

A **lop-kernel** for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \rightarrow \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq \text{opt}_\Pi(G) - (k - k') \Rightarrow \text{opt}_\Pi(G') \geq k'$.

We call a reduction rule as above a lop-rule.

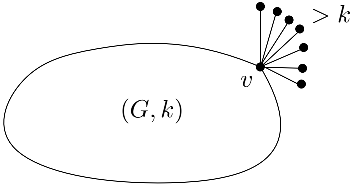
We also allow a lop-kernel to answer 'YES' directly.

lop-rules are a particular (?) type of reduction rules

Example of a rule that is not a lop-rule for MMVC:

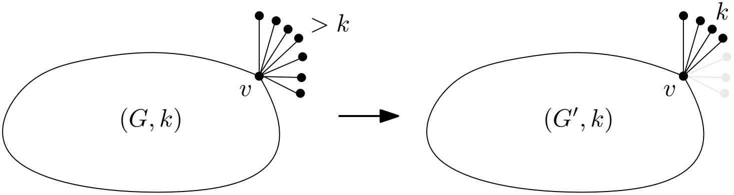
lop-rules are a particular (?) type of reduction rules

Example of a rule that is not a lop-rule for MMVC:



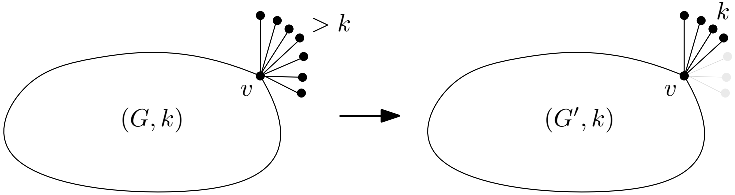
lop-rules are a particular (?) type of reduction rules

Example of a rule that is not a lop-rule for MMVC:



lop-rules are a particular (?) type of reduction rules

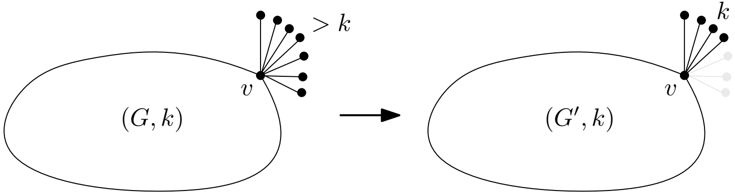
Example of a rule that is not a lop-rule for MMVC:



Want: mmvc(G) $\geq$ k

lop-rules are a particular (?) type of reduction rules

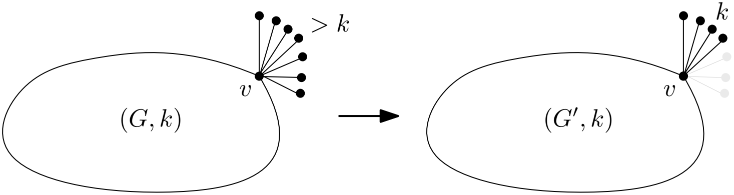
Example of a rule that is not a lop-rule for MMVC:



Want: $\text{mmvc}(G) \geq k \Rightarrow \text{mmvc}(G') \geq \text{mmvc}(G) - (k - k')$

lop-rules are a particular (?) type of reduction rules

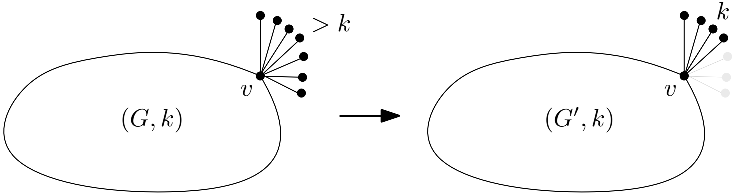
Example of a rule that is not a lop-rule for MMVC:



Want: $\text{mmvc}(G) \geq k \Rightarrow \text{mmvc}(G') \geq \text{mmvc}(G) - (k - k') = \text{mmvc}(G)$.

lop-rules are a particular (?) type of reduction rules

Example of a rule that is not a lop-rule for $\textsc{MMVC}$:

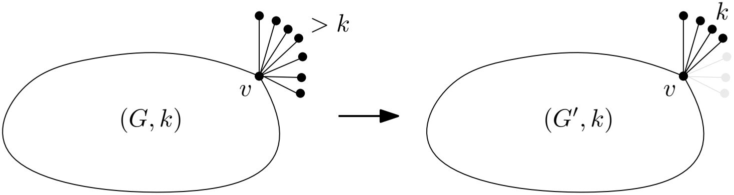


Want: $\text{mmvc}(G) \geq k \Rightarrow \text{mmvc}(G') \geq \text{mmvc}(G) - (k - k') = \text{mmvc}(G)$.

But $\text{mmvc}(G) > \text{mmvc}(G')$ may happen.

lop-rules are a particular (?) type of reduction rules

Example of a rule that is not a lop-rule for MMVC:



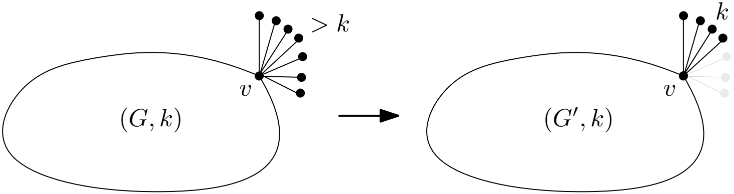
Want: $\text{mmvc}(G) \geq k \Rightarrow \text{mmvc}(G') \geq \text{mmvc}(G) - (k - k') = \text{mmvc}(G)$.

But $\text{mmvc}(G) > \text{mmvc}(G')$ may happen.

Anyway, we can just answer 'YES', so no problem!

lop-rules are a particular (?) type of reduction rules

Example of a rule that is not a lop-rule for MMVC:



Want: $\text{mmvc}(G) \geq k \Rightarrow \text{mmvc}(G') \geq \text{mmvc}(G) - (k - k') = \text{mmvc}(G)$.

But $\text{mmvc}(G) > \text{mmvc}(G')$ may happen.

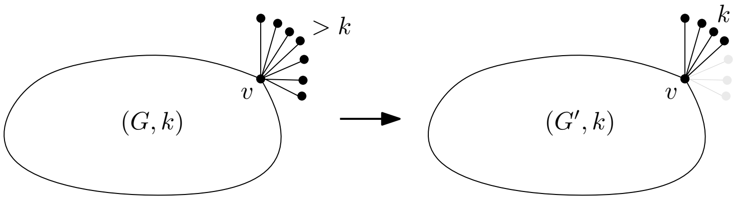
Anyway, we can just answer 'YES', so no problem!

Known examples of lop-rules:

- Classical reduction rules for VERTEX COVER:
  high-degree, crown decomposition, Nemhauser-Trotter.

lop-rules are a particular (?) type of reduction rules

Example of a rule that is not a lop-rule for MMVC:



Want: $\text{mmvc}(G) \geq k \Rightarrow \text{mmvc}(G') \geq \text{mmvc}(G) - (k - k') = \text{mmvc}(G)$.

But $\text{mmvc}(G) > \text{mmvc}(G')$ may happen.

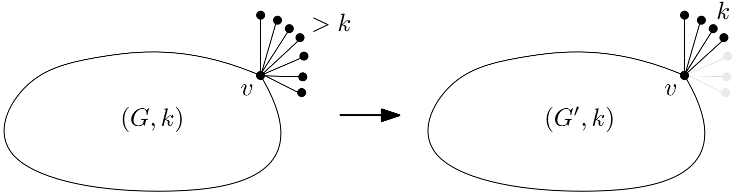
Anyway, we can just answer 'YES', so no problem!

Known examples of lop-rules:

- Classical reduction rules for VERTEX COVER:
  high-degree, crown decomposition, Nemhauser-Trotter.
- Rules based on protrusion replacement, matroids...

lop-rules are a particular (?) type of reduction rules

Example of a rule that is not a lop-rule for MMVC:



Want: $\text{mmvc}(G) \geq k \Rightarrow \text{mmvc}(G') \geq \text{mmvc}(G) - (k - k') = \text{mmvc}(G)$.

But $\text{mmvc}(G) > \text{mmvc}(G')$ may happen.

Anyway, we can just answer 'YES', so no problem!

Known examples of lop-rules:

- Classical reduction rules for VERTEX COVER:
  high-degree, crown decomposition, Nemhauser-Trotter.
- Rules based on protrusion replacement, matroids...

So far, we don't know of any reduction rule that is not a lop-rule!

A general result

Idea: lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

A general result

Idea: lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Theorem

Let Π be a vertex-maximization problem.
Let r and ε be real numbers in the interval $(0,1)$.

A general result

Idea: lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Theorem

Let Π be a vertex-maximization problem.

Let r and ε be real numbers in the interval $(0,1)$.

If Π parameterized by the solution size admits a lop-kernel with $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon})$ vertices,

A general result

Idea: lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Theorem

Let Π be a vertex-maximization problem.
Let r and ε be real numbers in the interval $(0, 1)$.
If Π parameterized by the solution size admits a lop-kernel with $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon})$ vertices, then Π admits a polynomial-time approximation algorithm with ratio $\mathcal{O}(n^{r-\varepsilon'})$ on n-vertex graphs, for some constant $\varepsilon' > 0$.

A general result

Idea: lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Theorem

Let Π be a vertex-maximization problem.
Let r and ε be real numbers in the interval $(0,1)$.
If Π parameterized by the solution size admits a lop-kernel with $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon})$ vertices, then Π admits a polynomial-time approximation algorithm with ratio $\mathcal{O}(n^{r-\varepsilon'})$ on n-vertex graphs, for some constant $\varepsilon' > 0$.

Thus, inapproximability results directly yield kernel lower bounds.

A general result

Idea: lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Theorem

Let Π be a vertex-maximization problem.
Let r and ε be real numbers in the interval $(0,1)$.
If Π parameterized by the solution size admits a lop-kernel with $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon})$ vertices, then Π admits a polynomial-time approximation algorithm with ratio $\mathcal{O}(n^{r-\varepsilon'})$ on n-vertex graphs, for some constant $\varepsilon' > 0$.

Thus, inapproximability results directly yield kernel lower bounds.

(Similar statement for vertex-minimization problems.)

Other frameworks to obtain kernelization lower bounds

Other frameworks to obtain kernelization lower bounds

- Rule out polynomial kernels, assuming NP $\not\subseteq$ coNP/poly.

  [Bodlaender, Downey, Fellows, Hermelin. 2009]

  [Bodlaender, Jansen, Kratsch. 2014]

Other frameworks to obtain kernelization lower bounds

- Rule out polynomial kernels, assuming $\mathsf{NP} \not\subseteq \mathsf{coNP}/\mathsf{poly}$.

  [Bodlaender, Downey, Fellows, Hermelin. 2009]

  [Bodlaender, Jansen, Kratsch. 2014]

- Weak compositions: lower bounds on the degree of polynomial kernels, assuming $\mathsf{NP} \not\subseteq \mathsf{coNP}/\mathsf{poly}$.

  [Dell and van Melkebeek. 2010] [Dell, Marx. 2012] [Hermelin, Wu. 2012]

Other frameworks to obtain kernelization lower bounds

- Rule out polynomial kernels, assuming NP $\not\subseteq$ coNP/poly.
  [Bodlaender, Downey, Fellows, Hermelin. 2009]
  [Bodlaender, Jansen, Kratsch. 2014]
- Weak compositions: lower bounds on the degree of polynomial kernels, assuming NP $\not\subseteq$ coNP/poly.
  [Dell and van Melkebeek. 2010] [Dell, Marx. 2012] [Hermelin, Wu. 2012]
- Transfer lower bounds: polynomial parameter transformations.
  [Bodlaender, Thomassé, Yeo. 2009]
  [Fernau, Fomin, Lokshtanov, Raible, Saurabh, Villanger. 2009]

Other frameworks to obtain kernelization lower bounds

- Rule out polynomial kernels, assuming NP $\not\subseteq$ coNP/poly.
  [Bodlaender, Downey, Fellows, Hermelin. 2009]
  [Bodlaender, Jansen, Kratsch. 2014]
- Weak compositions: lower bounds on the degree of polynomial kernels, assuming NP $\not\subseteq$ coNP/poly.
  [Dell and van Melkebeek. 2010] [Dell, Marx. 2012] [Hermelin, Wu. 2012]
- Transfer lower bounds: polynomial parameter transformations.
  [Bodlaender, Thomassé, Yeo. 2009]
  [Fernau, Fomin, Lokshtanov, Raible, Saurabh, Villanger. 2009]
- Lower bounds on the coefficients of linear kernels, assuming P $\neq$ NP.
  [Chen, Fernau, Kanj, Xia. 2007]

Other frameworks to obtain kernelization lower bounds

- Rule out polynomial kernels, assuming $\mathsf{NP} \not\subseteq \mathsf{coNP}/\mathsf{poly}$.
  [Bodlaender, Downey, Fellows, Hermelin. 2009]
  [Bodlaender, Jansen, Kratsch. 2014]
- Weak compositions: lower bounds on the degree of polynomial kernels, assuming $\mathsf{NP} \not\subseteq \mathsf{coNP}/\mathsf{poly}$.
  [Dell and van Melkebeek. 2010] [Dell, Marx. 2012] [Hermelin, Wu. 2012]
- Transfer lower bounds: polynomial parameter transformations.
  [Bodlaender, Thomassé, Yeo. 2009]
  [Fernau, Fomin, Lokshtanov, Raible, Saurabh, Villanger. 2009]
- Lower bounds on the coefficients of linear kernels, assuming $\mathsf{P} \neq \mathsf{NP}$.
  [Chen, Fernau, Kanj, Xia. 2007]
- Lossy kernelization. [Lokshtanov, Panolan, Ramanujan, Saurabh. 2017]

Other frameworks to obtain kernelization lower bounds

- Rule out polynomial kernels, assuming NP $\not\subseteq$ coNP/poly.
  [Bodlaender, Downey, Fellows, Hermelin. 2009]
  [Bodlaender, Jansen, Kratsch. 2014]
- Weak compositions: lower bounds on the degree of polynomial kernels, assuming NP $\not\subseteq$ coNP/poly.
  [Dell and van Melkebeek. 2010] [Dell, Marx. 2012] [Hermelin, Wu. 2012]
- Transfer lower bounds: polynomial parameter transformations.
  [Bodlaender, Thomassé, Yeo. 2009]
  [Fernau, Fomin, Lokshtanov, Raible, Saurabh, Villanger. 2009]
- Lower bounds on the coefficients of linear kernels, assuming P $\neq$ NP.
  [Chen, Fernau, Kanj, Xia. 2007]
- Lossy kernelization. [Lokshtanov, Panolan, Ramanujan, Saurabh. 2017]

Strong points immediate application, weak hypothesis (P $\neq$ NP).

Other frameworks to obtain kernelization lower bounds

- Rule out polynomial kernels, assuming NP $\not\subseteq$ coNP/poly.
  [Bodlaender, Downey, Fellows, Hermelin. 2009]
  [Bodlaender, Jansen, Kratsch. 2014]
- Weak compositions: lower bounds on the degree of polynomial kernels, assuming NP $\not\subseteq$ coNP/poly.
  [Dell and van Melkebeek. 2010] [Dell, Marx. 2012] [Hermelin, Wu. 2012]
- Transfer lower bounds: polynomial parameter transformations.
  [Bodlaender, Thomassé, Yeo. 2009]
  [Fernau, Fomin, Lokshtanov, Raible, Saurabh, Villanger. 2009]
- Lower bounds on the coefficients of linear kernels, assuming P $\neq$ NP.
  [Chen, Fernau, Kanj, Xia. 2007]
- Lossy kernelization. [Lokshtanov, Panolan, Ramanujan, Saurabh. 2017]

Strong points immediate application, weak hypothesis (P $\neq$ NP).

Weak points needs strong inapproximability result, only vertex problems.

A consequence of our general result

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

A consequence of our general result

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Maximum Minimal Vertex Cover does not admit an $\mathcal{O}(n^{\frac{1}{2}-\varepsilon})$-approximation, unless $\mathsf{P} = \mathsf{NP}$. [Boria, Della Croce, Paschos. 2015]

A consequence of our general result

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

MAXIMUM MINIMAL VERTEX COVER does not admit an $\mathcal{O}(n^{\frac{1}{2}-\varepsilon})$-approximation, unless $\mathsf{P} = \mathsf{NP}$. [Boria, Della Croce, Paschos. 2015]

By just plugging $r = \frac{1}{2}$ in our general result we obtain:

Corollary

MAXIMUM MINIMAL VERTEX COVER *parameterized by the solution size does not admit a* lop-*kernel* *with* $\mathcal{O}(k^{2-\varepsilon})$ *vertices, unless* $\mathsf{P} = \mathsf{NP}$.

A consequence of our general result

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Maximum Minimal Vertex Cover does not admit an $\mathcal{O}(n^{\frac{1}{2}-\varepsilon})$-approximation, unless $\mathsf{P} = \mathsf{NP}$. [Boria, Della Croce, Paschos. 2015]

By just plugging $r = \frac{1}{2}$ in our general result we obtain:

Corollary

Maximum Minimal Vertex Cover *parameterized by the solution size does not admit a lop-kernel with $\mathcal{O}(k^{2-\varepsilon})$ vertices, unless $\mathsf{P} = \mathsf{NP}$.*

Thus, the trivial quadratic kernel is "essentially" optimal.

Another consequence of our general result

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Another consequence of our general result

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Maximum Minimal Feedback Vertex Set parameterized by the solution size k admits a kernel of size $\mathcal{O}(k^3)$.

[Dublois, Hanaka, Ghadikolaei, Lampis, Melissinos. 2020]

Another consequence of our general result

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Maximum Minimal Feedback Vertex Set parameterized by the solution size k admits a kernel of size $\mathcal{O}(k^3)$.

[Dublois, Hanaka, Ghadikolaei, Lampis, Melissinos. 2020]

Open problem: does a kernel smaller than $\mathcal{O}(k^3)$ exist?

Another consequence of our general result

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Maximum Minimal Feedback Vertex Set parameterized by the solution size k admits a kernel of size $\mathcal{O}(k^3)$.

[Dublois, Hanaka, Ghadikolaei, Lampis, Melissinos. 2020]

Open problem: does a kernel smaller than $\mathcal{O}(k^3)$ exist?

Maximum Minimal Feedback Vertex Set does not admit an $\mathcal{O}(n^{\frac{2}{3}-\varepsilon})$-approximation, unless $\mathsf{P} = \mathsf{NP}$. [Dublois et al. 2020]

Another consequence of our general result

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

MAXIMUM MINIMAL FEEDBACK VERTEX SET parameterized by the solution size k admits a kernel of size $\mathcal{O}(k^3)$.

[Dublois, Hanaka, Ghadikolaei, Lampis, Melissinos. 2020]

Open problem: does a kernel smaller than $\mathcal{O}(k^3)$ exist?

MAXIMUM MINIMAL FEEDBACK VERTEX SET does not admit an $\mathcal{O}(n^{\frac{2}{3}-\varepsilon})$-approximation, unless $\mathsf{P} = \mathsf{NP}$. [Dublois et al. 2020]

By just plugging $r = \frac{2}{3}$ in our general result we obtain:

Corollary

If $\mathsf{P} \neq \mathsf{NP}$, MAXIMUM MINIMAL FEEDBACK VERTEX SET parameterized by the solution size does not admit a lop-kernel with $\mathcal{O}(k^{3-\varepsilon})$ vertices.

Subquadratic kernels on particular graph classes

We use, for the first time, the Erdős-Hajnal property in kernelization.

Subquadratic kernels on particular graph classes

We use, for the first time, the Erdős-Hajnal property in kernelization.

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by* k *restricted to bull-free graphs admits a kernel with* $\mathcal{O}(k^{7/4})$ *vertices.*

Subquadratic kernels on particular graph classes

We use, for the first time, the Erdős-Hajnal property in kernelization.

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by* k *restricted to bull-free graphs admits a kernel with* $\mathcal{O}(k^{7/4})$ *vertices.*

Theorem

For every $t \geq 3$, MAXIMUM MINIMAL VERTEX COVER *parameterized by* k *restricted to* K_t*-free graphs admits a kernel with* $\mathcal{O}(k^{\frac{2t-3}{t-1}})$ *vertices.*

Subquadratic kernels on particular graph classes

We use, for the first time, the Erdős-Hajnal property in kernelization.

Theorem

The Maximum Minimal Vertex Cover *problem parameterized by k restricted to bull-free graphs admits a kernel with $\mathcal{O}(k^{7/4})$ vertices.*

Theorem

For every $t \geq 3$, Maximum Minimal Vertex Cover *parameterized by k restricted to K_t-free graphs admits a kernel with $\mathcal{O}(k^{\frac{2t-3}{t-1}})$ vertices.*

Theorem

The Maximum Minimal Vertex Cover *problem parameterized by k restricted to paw-free graphs admits a kernel with $\mathcal{O}(k^{5/3})$ vertices.*

Next section is...

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by the size of a minimum vertex cover of the input graph does not admit a polynomial kernel unless* NP $\subseteq$ coNP/poly, *even on bipartite graphs.*

Theorem

The Maximum Minimal Vertex Cover *problem parameterized by the size of a minimum vertex cover of the input graph does not admit a polynomial kernel unless* NP $\subseteq$ coNP/poly, *even on bipartite graphs.*

Polynomial parameter transformation (PPT):

Instance (x, k) of A → polynomial time → Instance (x', k') of B

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by the size of a minimum vertex cover of the input graph does not admit a polynomial kernel unless* NP $\subseteq$ coNP/poly, *even on bipartite graphs.*

Polynomial parameter transformation (PPT):

Instance (x, k) of A → polynomial time → Instance (x', k') of B

1. (x, k) is a YES-instance of A $\Leftrightarrow$ (x', k') is a YES-instance of B.
2. $k' = \mathcal{O}(k^c)$ for some constant c.

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by the size of a minimum vertex cover of the input graph does not admit a polynomial kernel unless* $\mathsf{NP} \subseteq \mathsf{coNP}/\mathsf{poly}$, *even on bipartite graphs.*

Polynomial parameter transformation (PPT):

Instance (x, k) of A → **polynomial time** → Instance (x', k') of B

1. (x, k) is a YES-instance of A $\Leftrightarrow$ (x', k') is a YES-instance of B.
2. $k' = \mathcal{O}(k^c)$ for some constant c.

If A does not admit a polynomial kernel and $\exists$ a PPT from A to B, then B does not admit a polynomial kernel, assuming $\mathsf{NP} \subseteq \mathsf{coNP}/\mathsf{poly}$.

Theorem

The Maximum Minimal Vertex Cover *problem parameterized by the size of a minimum vertex cover of the input graph does not admit a polynomial kernel unless* $\mathsf{NP} \subseteq \mathsf{coNP}/\mathsf{poly}$*, even on bipartite graphs.*

Polynomial parameter transformation (PPT):

1. (x, k) is a Yes-instance of A $\Leftrightarrow$ (x', k') is a Yes-instance of B.
2. $k' = \mathcal{O}(k^c)$ for some constant c.

If A does not admit a polynomial kernel and $\exists$ a PPT from A to B, then B does not admit a polynomial kernel, assuming $\mathsf{NP} \subseteq \mathsf{coNP}/\mathsf{poly}$.

We present a PPT from the Monotone Sat problem parameterized by the number of variables, which is known not to admit a polynomial kernel.

[Fortnow, Santhanam. 2011]

Let ϕ be an instance of MONOTONE SAT, with n variables and m clauses.

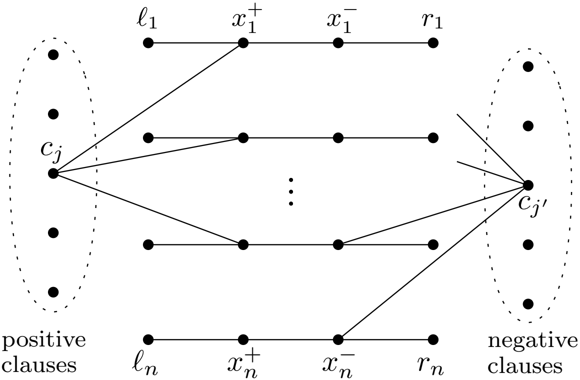
Let ϕ be an instance of MONOTONE SAT, with n variables and m clauses.

The literals in each clause of ϕ are either all positive or all negative.

Let ϕ be an instance of MONOTONE SAT, with n variables and m clauses.

The literals in each clause of ϕ are either all positive or all negative.

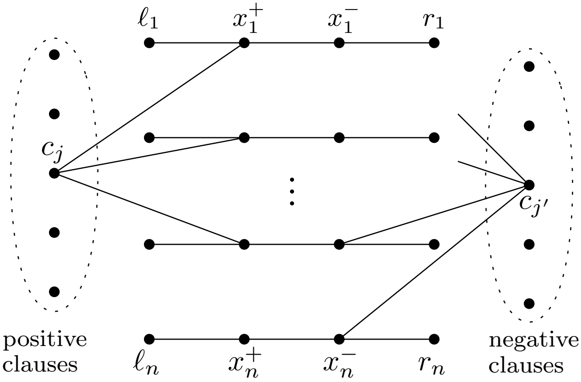
We construct in poly time an instance (G, k) of MMVC with $k := 2n + m$:



Let ϕ be an instance of MONOTONE SAT, with n variables and m clauses.

The literals in each clause of ϕ are either all positive or all negative.

We construct in poly time an instance (G, k) of MMVC with $k := 2n + m$:

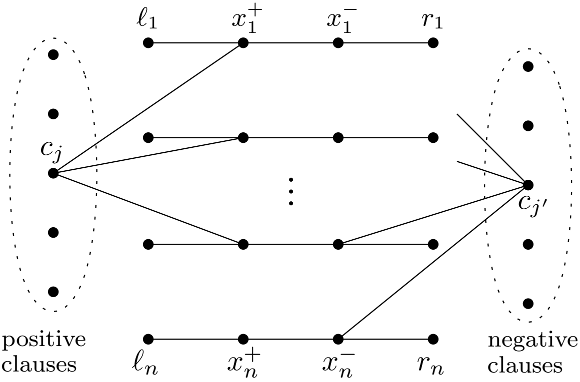


The set $\{x_i^+, x_i^- \mid i \in [n]\}$ is a minimum vertex cover of G of size $2n$.

Let ϕ be an instance of MONOTONE SAT, with n variables and m clauses.

The literals in each clause of ϕ are either all positive or all negative.

We construct in poly time an instance (G, k) of MMVC with $k := 2n + m$:



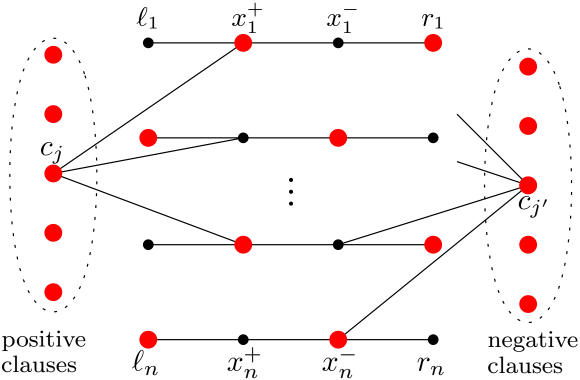
The set $\{x_i^+, x_i^- \mid i \in [n]\}$ is a minimum vertex cover of G of size $2n$.

ϕ is satisfiable $\Leftrightarrow$ G contains a minimal vertex cover of size k.

Let ϕ be an instance of MONOTONE SAT, with n variables and m clauses.

The literals in each clause of ϕ are either all positive or all negative.

We construct in poly time an instance (G, k) of MMVC with $k := 2n + m$:



The set $\{x_i^+, x_i^- \mid i \in [n]\}$ is a minimum vertex cover of G of size $2n$.

ϕ is satisfiable $\Leftrightarrow$ G contains a minimal vertex cover of size k.

Our framework to rule out subquadratic kernels for MMVC

Our framework to rule out subquadratic kernels for MMVC

A lop-kernel for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \to \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq \text{opt}_\Pi(G) - (k - k')$ $(\Rightarrow \text{opt}_\Pi(G') \geq k')$.

We call a reduction rule as above a lop-rule.

Our framework to rule out subquadratic kernels for MMVC

A lop-kernel for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \to \mathbb{N}$, called the size of the kernel, s.t.

1. $\text{opt}_\Pi(G') \geq k' \Rightarrow \text{opt}_\Pi(G) \geq k$.
2. $\text{opt}_\Pi(G) \geq k \Rightarrow \text{opt}_\Pi(G') \geq \text{opt}_\Pi(G) - (k - k')$ $(\Rightarrow \text{opt}_\Pi(G') \geq k')$.

We call a reduction rule as above a lop-rule.

We also allow a lop-kernel to answer 'YES' directly.

Our framework to rule out subquadratic kernels for MMVC

A lop-kernel for Π with parameter k is a polynomial-time algorithm that, given an instance (G, k), produces an instance (G', k') with $|V(G')| \leq s(k)$ for some function $s : \mathbb{N} \to \mathbb{N}$, called the size of the kernel, s.t.

1. $\mathrm{opt}_\Pi(G') \geq k' \Rightarrow \mathrm{opt}_\Pi(G) \geq k$.
2. $\mathrm{opt}_\Pi(G) \geq k \Rightarrow \mathrm{opt}_\Pi(G') \geq \mathrm{opt}_\Pi(G) - (k - k')$ $(\Rightarrow \mathrm{opt}_\Pi(G') \geq k')$.

We call a reduction rule as above a lop-rule.

We also allow a lop-kernel to answer 'YES' directly.

Idea: lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Sketch of proof

Let Π be a vertex-maximization problem parameterized by solution size k.

Sketch of proof

Let Π be a vertex-maximization problem parameterized by solution size k.

Suppose that Π admits a lop-kernel of size $\mathcal{O}(k^c)$ for some constant $c \geq 1$.

Sketch of proof

Let Π be a vertex-maximization problem parameterized by solution size k.

Suppose that Π admits a lop-kernel of size $\mathcal{O}(k^c)$ for some constant $c \geq 1$.

Then, given an instance (G, k), in poly time either we conclude that

Sketch of proof

Let Π be a vertex-maximization problem parameterized by solution size k.

Suppose that Π admits a lop-kernel of size $\mathcal{O}(k^c)$ for some constant $c \geq 1$.

Then, given an instance (G, k), in poly time either we conclude that

- $\text{opt}_\Pi(G) \geq k$, or

Sketch of proof

Let Π be a vertex-maximization problem parameterized by solution size k.

Suppose that Π admits a lop-kernel of size $\mathcal{O}(k^c)$ for some constant $c \geq 1$.

Then, given an instance (G, k), in poly time either we conclude that

- $\text{opt}_\Pi(G) \geq k$, or
- we obtain an equivalent instance (G', k') such that

$$\text{opt}_\Pi(G) \leq \text{opt}_\Pi(G') + (k - k')$$

Sketch of proof

Let Π be a vertex-maximization problem parameterized by solution size k.

Suppose that Π admits a lop-kernel of size $\mathcal{O}(k^c)$ for some constant $c \geq 1$.

Then, given an instance (G, k), in poly time either we conclude that

- $\text{opt}_\Pi(G) \geq k$, or
- we obtain an equivalent instance (G', k') such that

$$\text{opt}_\Pi(G) \leq \text{opt}_\Pi(G') + (k - k') \leq |V(G')| + k$$

Sketch of proof

Let Π be a vertex-maximization problem parameterized by solution size k.

Suppose that Π admits a lop-kernel of size $\mathcal{O}(k^c)$ for some constant $c \geq 1$.

Then, given an instance (G, k), in poly time either we conclude that

- $\text{opt}_\Pi(G) \geq k$, or
- we obtain an equivalent instance (G', k') such that

$$\text{opt}_\Pi(G) \leq \text{opt}_\Pi(G') + (k - k') \leq |V(G')| + k = \mathcal{O}(k^c).$$

Sketch of proof

Let Π be a vertex-maximization problem parameterized by solution size k.

Suppose that Π admits a lop-kernel of size $\mathcal{O}(k^c)$ for some constant $c \geq 1$.

Then, given an instance (G, k), in poly time either we conclude that

- $\text{opt}_\Pi(G) \geq k$, or
- we obtain an equivalent instance (G', k') such that

$$\text{opt}_\Pi(G) \leq \text{opt}_\Pi(G') + (k - k') \leq |V(G')| + k = \mathcal{O}(k^c).$$

From this, it is not difficult to see that we can obtain a polynomial-time approximation algorithm for Π with the desired ratio:

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Sketch of proof

Let Π be a vertex-maximization problem parameterized by solution size k.

Suppose that Π admits a lop-kernel of size $\mathcal{O}(k^c)$ for some constant $c \geq 1$.

Then, given an instance (G, k), in poly time either we conclude that

- $\text{opt}_\Pi(G) \geq k$, or
- we obtain an equivalent instance (G', k') such that

$$\text{opt}_\Pi(G) \leq \text{opt}_\Pi(G') + (k - k') \leq |V(G')| + k = \mathcal{O}(k^c).$$

From this, it is not difficult to see that we can obtain a polynomial-time approximation algorithm for Π with the desired ratio:

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

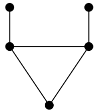
It holds with $\varepsilon' := \varepsilon^2 \cdot \frac{(1-r)^2}{r}$.

Subquadratic kernels on particular graph classes

Subquadratic kernels on particular graph classes

Theorem

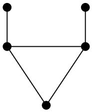
The Maximum Minimal Vertex Cover *problem parameterized by* k *restricted to bull-free graphs admits a kernel with* $\mathcal{O}(k^{7/4})$ *vertices.*



Subquadratic kernels on particular graph classes

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by* k *restricted to bull-free graphs admits a kernel with* $\mathcal{O}(k^{7/4})$ *vertices.*

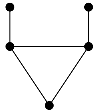


A graph H satisfies the Erdős-Hajnal property if there exists a constant $\delta > 0$ such that every H-free graph G with n vertices contains either a clique or an independent set of size n^{δ}.

Subquadratic kernels on particular graph classes

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by* k *restricted to bull-free graphs admits a kernel with* $\mathcal{O}(k^{7/4})$ *vertices.*



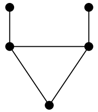
A graph H satisfies the Erdős-Hajnal property if there exists a constant $\delta > 0$ such that every H-free graph G with n vertices contains either a clique or an independent set of size n^{δ}.

Conjecture: every graph H satisfies the E-H property. [Erdős, Hajnal. 1982]

Subquadratic kernels on particular graph classes

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by k restricted to bull-free graphs admits a kernel with $\mathcal{O}(k^{7/4})$ vertices.*



A graph H satisfies the Erdős-Hajnal property if there exists a constant $\delta > 0$ such that every H-free graph G with n vertices contains either a clique or an independent set of size n^{δ}.

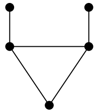
Conjecture: every graph H satisfies the E-H property. [Erdős, Hajnal. 1982]

True for graphs with at most 4 vertices, the bull, the complete graphs...
[Chudnovsky. 2014]

Subquadratic kernels on particular graph classes

Theorem

The MAXIMUM MINIMAL VERTEX COVER *problem parameterized by* k *restricted to bull-free graphs admits a kernel with* $\mathcal{O}(k^{7/4})$ *vertices.*



A graph H satisfies the Erdős-Hajnal property if there exists a constant $\delta > 0$ such that every H-free graph G with n vertices contains either a clique or an independent set of size n^{δ}.

Conjecture: every graph H satisfies the E-H property. [Erdős, Hajnal. 1982]

True for graphs with at most 4 vertices, the bull, the complete graphs... [Chudnovsky. 2014]

For all the known cases, such a clique or independent set of size n^{δ} can be found in polynomial time.

Two useful lemmas

Lemma

Let H be a graph satisfying the constructive Erdős-Hajnal property with constant $\delta > 0$, and let G be an H-free graph.

Two useful lemmas

Lemma

Let H be a graph satisfying the constructive Erdős-Hajnal property with constant $\delta > 0$, and let G be an H-free graph. Then $V(G)$ can be partitioned in polynomial time into a collection of cliques $\mathcal{C}$ and a collection of independent sets $\mathcal{I}$ such that $|\mathcal{C}| + |\mathcal{I}| = \mathcal{O}(|V(G)|^{1-\delta})$.

Two useful lemmas

Lemma

Let H be a graph satisfying the constructive Erdős-Hajnal property with constant $\delta > 0$, and let G be an H-free graph. Then $V(G)$ can be partitioned in polynomial time into a collection of cliques $\mathcal{C}$ and a collection of independent sets $\mathcal{I}$ such that $|\mathcal{C}| + |\mathcal{I}| = \mathcal{O}(|V(G)|^{1-\delta})$.

Lemma

Let G be a graph and let $S \subseteq V(G)$ be an independent set.

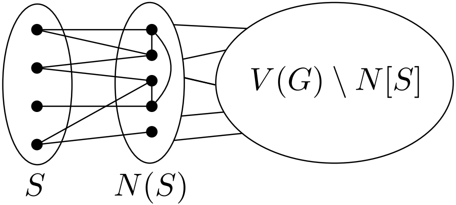
Two useful lemmas

Lemma

Let H be a graph satisfying the constructive Erdős-Hajnal property with constant $\delta > 0$, and let G be an H-free graph. Then $V(G)$ can be partitioned in polynomial time into a collection of cliques $\mathcal{C}$ and a collection of independent sets $\mathcal{I}$ such that $|\mathcal{C}| + |\mathcal{I}| = \mathcal{O}(|V(G)|^{1-\delta})$.

Lemma

Let G be a graph and let $S \subseteq V(G)$ be an independent set. Then there exists a minimal vertex cover of G containing $N(S)$.



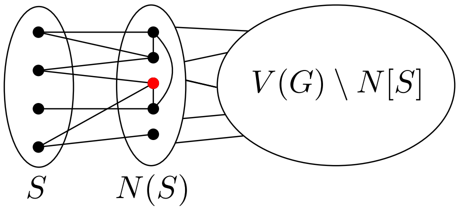
Two useful lemmas

Lemma

Let H be a graph satisfying the constructive Erdős-Hajnal property with constant $\delta > 0$, and let G be an H-free graph. Then $V(G)$ can be partitioned in polynomial time into a collection of cliques $\mathcal{C}$ and a collection of independent sets $\mathcal{I}$ such that $|\mathcal{C}| + |\mathcal{I}| = \mathcal{O}(|V(G)|^{1-\delta})$.

Lemma

Let G be a graph and let $S \subseteq V(G)$ be an independent set. Then there exists a minimal vertex cover of G containing $N(S)$.



A subquadratic kernel on bull-free graphs: sketch

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

We can assume that the maximum degree of G is at most $k - 1$.

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

We can assume that the maximum degree of G is at most $k - 1$.

We find greedily a minimal vertex cover X of G.

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

We can assume that the maximum degree of G is at most $k - 1$.

We find greedily a minimal vertex cover X of G.

We can assume $|X| \leq k - 1$.

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

We can assume that the maximum degree of G is at most $k - 1$.

We find greedily a minimal vertex cover X of G.

We can assume $|X| \leq k - 1$. Let $S = V(G) \setminus X$.

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

We can assume that the maximum degree of G is at most $k - 1$.

We find greedily a minimal vertex cover X of G.

We can assume $|X| \leq k - 1$. Let $S = V(G) \setminus X$. Goal: bound $|S|$.

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

We can assume that the maximum degree of G is at most $k - 1$.

We find greedily a minimal vertex cover X of G.

We can assume $|X| \leq k - 1$. Let $S = V(G) \setminus X$. Goal: bound $|S|$.

The bull satisfies the EH-property with $\delta = \frac{1}{4}$. [Chudnovsky, Safra. 2008]

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

We can assume that the maximum degree of G is at most $k-1$.

We find greedily a minimal vertex cover X of G.

We can assume $|X| \leq k-1$. Let $S = V(G) \setminus X$. Goal: bound $|S|$.

The bull satisfies the EH-property with $\delta = \frac{1}{4}$. [Chudnovsky, Safra. 2008]

Partition $G[X]$ into cliques $\mathcal{C}$ and indep. sets $\mathcal{I}$ with $|\mathcal{C}| + |\mathcal{I}| = \mathcal{O}(k^{3/4})$.

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

We can assume that the maximum degree of G is at most $k-1$.

We find greedily a minimal vertex cover X of G.

We can assume $|X| \leq k-1$. Let $S = V(G) \setminus X$. Goal: bound $|S|$.

The bull satisfies the EH-property with $\delta = \frac{1}{4}$. [Chudnovsky, Safra. 2008]

Partition $G[X]$ into cliques $\mathcal{C}$ and indep. sets $\mathcal{I}$ with $|\mathcal{C}| + |\mathcal{I}| = \mathcal{O}(k^{3/4})$.

Since S is an independent set and there are no isolated vertices,

$$S = \bigcup_{C \in \mathcal{C}} N_S(C) \cup \bigcup_{I \in \mathcal{I}} N_S(I).$$

A subquadratic kernel on bull-free graphs: sketch

Let (G, k) be an instance of MMVC, where G is bull-free.

We can assume that the maximum degree of G is at most $k - 1$.

We find greedily a minimal vertex cover X of G.

We can assume $|X| \leq k - 1$. Let $S = V(G) \setminus X$. Goal: bound $|S|$.

The bull satisfies the EH-property with $\delta = \frac{1}{4}$. [Chudnovsky, Safra. 2008]

Partition $G[X]$ into cliques $\mathcal{C}$ and indep. sets $\mathcal{I}$ with $|\mathcal{C}| + |\mathcal{I}| = \mathcal{O}(k^{3/4})$.

Since S is an independent set and there are no isolated vertices,

$$S = \bigcup_{C \in \mathcal{C}} N_S(C) \cup \bigcup_{I \in \mathcal{I}} N_S(I).$$

Enough: for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$

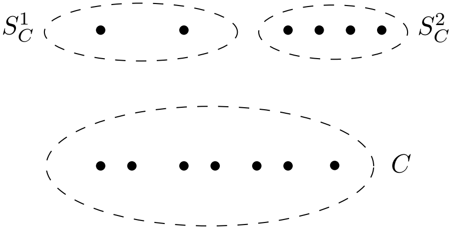
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



Partition $N_S(C) = S_C^1 \uplus S_C^2$

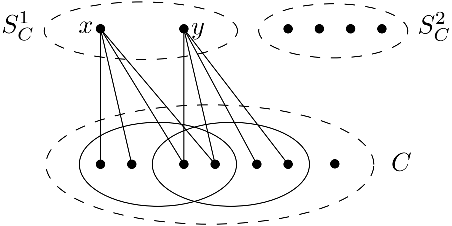
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



Partition $N_S(C) = S_C^1 \uplus S_C^2$ so that S_C^1 is a maximal subset of $N_S(C)$ s.t. the neighborhoods of its vertices pairwise do not cover all the clique C.

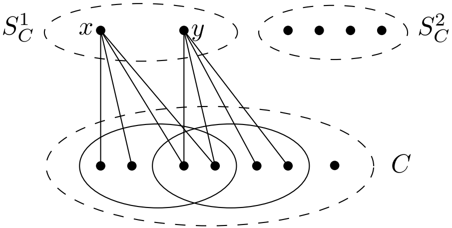
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



Claim: The vertices in S_C^1 can be ordered $x_1, \ldots, x_p$ so that $N_C(x_i) \subseteq N_C(x_j)$ whenever $i \leq j$.

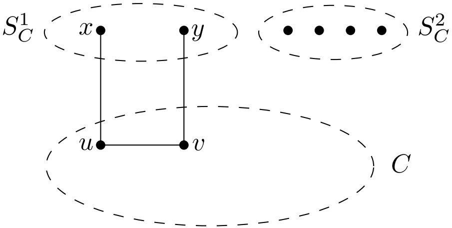
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



Claim: The vertices in S_C^1 can be ordered $x_1, \ldots, x_p$ so that $N_C(x_i) \subseteq N_C(x_j)$ whenever $i \leq j$.

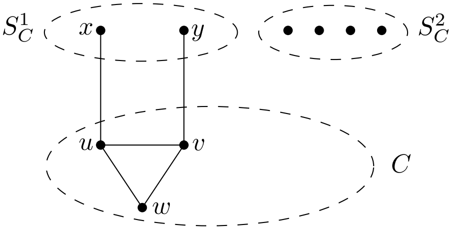
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



Claim: The vertices in S_C^1 can be ordered $x_1, \ldots, x_p$ so that $N_C(x_i) \subseteq N_C(x_j)$ whenever $i \leq j$.

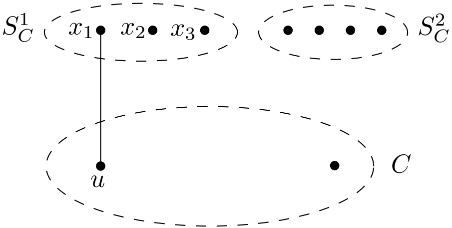
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



By the Claim, there is a vertex $u \in \bigcap_{x \in S_C^1} N_C(x)$.

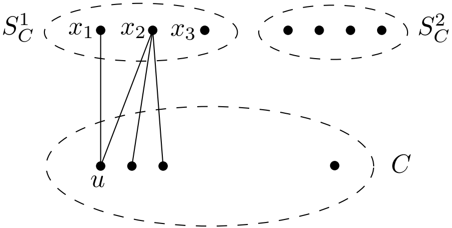
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



By the Claim, there is a vertex $u \in \bigcap_{x \in S_C^1} N_C(x)$.

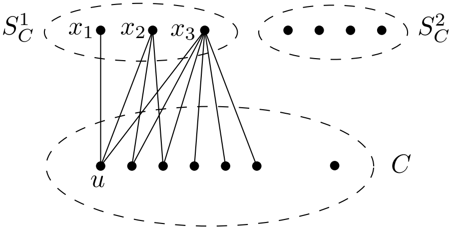
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



By the Claim, there is a vertex $u \in \bigcap_{x \in S_C^1} N_C(x)$.

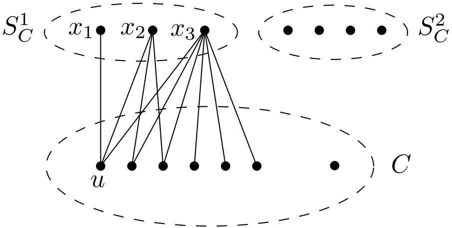
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



By the Claim, there is a vertex $u \in \bigcap_{x \in S_C^1} N_C(x)$.

Since $\deg(u) \leq k - 1$, it follows that $|S_C^1| \leq k - 1$.

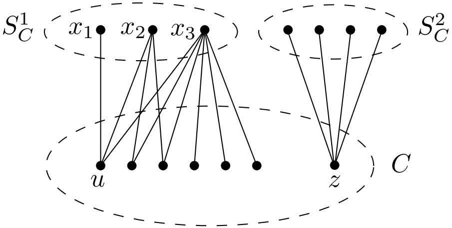
Goal for every $Y \in \mathcal{C} \cup \mathcal{I}$, show that $|N_S(Y)| = \mathcal{O}(k)$.

Suppose first that $I \in \mathcal{I}$ is an independent set.

From the second Lemma, if $|N_S(I)| \geq k$ then (G, k) is a YES-instance,

So we can assume that $|N_S(I)| \leq k - 1$.

Suppose now that $C \in \mathcal{C}$ is a clique. Goal $|N_S(C)| = \mathcal{O}(k)$



There exists a vertex $z \in C \setminus \bigcup_{y \in S_C^1} N_C(y)$.

It follows that $z \in \bigcap_{x \in S_C^2} N_C(x)$, and since $\deg(z) \leq k-1$, $|S_C^2| \leq k-1$.

Next section is...

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?
- Apply our framework to vertex-minimization problems.

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?
- Apply our framework to vertex-minimization problems.

Subquadratic kernels for MMVC on H-free graphs using the EH-property

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?
- Apply our framework to vertex-minimization problems.

Subquadratic kernels for MMVC on H-free graphs using the EH-property

- Other graphs H satisfying the E-H property: C_4, the diamond, P_5, C_5.

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?
- Apply our framework to vertex-minimization problems.

Subquadratic kernels for MMVC on H-free graphs using the EH-property

- Other graphs H satisfying the E-H property: C_4, the diamond, P_5, C_5.
- The complexity of MMVC on P_5-free graphs is open.

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?
- Apply our framework to vertex-minimization problems.

Subquadratic kernels for MMVC on H-free graphs using the EH-property

- Other graphs H satisfying the E-H property: C_4, the diamond, P_5, C_5.
- The complexity of MMVC on P_5-free graphs is open.
- If G is a graph on n vertices without isolated vertices, then
  $\mathrm{mmvc}(G) \geq \lfloor n^{1/2} \rfloor$. [Boria, Della Croce, Paschos. 2015]

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?
- Apply our framework to vertex-minimization problems.

Subquadratic kernels for MMVC on H-free graphs using the EH-property

- Other graphs H satisfying the E-H property: C_4, the diamond, P_5, C_5.
- The complexity of MMVC on P_5-free graphs is open.
- If G is a graph on n vertices without isolated vertices, then
  $\mathrm{mmvc}(G) \geq \lfloor n^{1/2} \rfloor$. [Boria, Della Croce, Paschos. 2015]

  This immediately yields a quadratic kernel for MMVC.

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?
- Apply our framework to vertex-minimization problems.

Subquadratic kernels for MMVC on H-free graphs using the EH-property

- Other graphs H satisfying the E-H property: C_4, the diamond, P_5, C_5.
- The complexity of MMVC on P_5-free graphs is open.
- If G is a graph on n vertices without isolated vertices, then
  $\mathrm{mmvc}(G) \geq \lfloor n^{1/2} \rfloor$. [Boria, Della Croce, Paschos. 2015]

  This immediately yields a quadratic kernel for MMVC.

  Is it possible that, for the H-free graphs that we considered, $\mathrm{mmvc}(G) \geq n^{1/2+\varepsilon}$, for some $\varepsilon > 0$?

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?
- Apply our framework to vertex-minimization problems.

Subquadratic kernels for MMVC on H-free graphs using the EH-property

- Other graphs H satisfying the E-H property: C_4, the diamond, P_5, C_5.
- The complexity of MMVC on P_5-free graphs is open.
- If G is a graph on n vertices without isolated vertices, then
  $\mathrm{mmvc}(G) \geq \lfloor n^{1/2} \rfloor$. [Boria, Della Croce, Paschos. 2015]

  This immediately yields a quadratic kernel for MMVC.

  Is it possible that, for the H-free graphs that we considered, $\mathrm{mmvc}(G) \geq n^{1/2+\varepsilon}$, for some $\varepsilon > 0$? Triangle-free graphs?

Further research

lop-kernel of size $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon}) \Rightarrow \mathcal{O}(n^{r-\varepsilon'})$-approximation algorithm

- Can the "lop" assumption be removed?
- Are there natural reduction rules that are not lop-rules?
- Apply our framework to vertex-minimization problems.

Subquadratic kernels for MMVC on H-free graphs using the EH-property

- Other graphs H satisfying the E-H property: C_4, the diamond, P_5, C_5.
- The complexity of MMVC on P_5-free graphs is open.
- If G is a graph on n vertices without isolated vertices, then $\text{mmvc}(G) \geq \lfloor n^{1/2} \rfloor$. [Boria, Della Croce, Paschos. 2015]

  This immediately yields a quadratic kernel for MMVC.

  Is it possible that, for the H-free graphs that we considered, $\text{mmvc}(G) \geq n^{1/2+\varepsilon}$, for some $\varepsilon > 0$? Triangle-free graphs?

  If so, it would immediately yield a subquadratic kernel.

Gràcies!