Optimal Data Reduction for Graph Coloring Using Low-Degree Polynomials

Bart M. P. Jansen and Astrid Pieterse

Eindhoven University of Technology

September 8, 2017

Technische Universiteit
Eindhoven
University of Technology

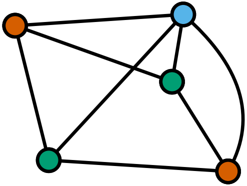
The q-Coloring problem

Can the vertices of a graph be colored with at most q colors?

- NP-hard, use a parameterized approach

Which parameter(s)?

- Number of colors
  - Uninteresting
- Structural parameters
  - In this talk: Vertex Cover
- Number of vertices (sparsification)



Previous work

- Fiala et al.: coloring problems parameterized by
  - Vertex Cover vs. Treewidth
- Jansen and Kratsch: parameter hierarchy for graph coloring

Jansen and Kratsch [Inf Comput. 2013] showed that

- q-Coloring parameterized by VC has a kernel of bitsize $O(k^q)$
- But no kernel of size $O(k^{q-1-\varepsilon})$ for $q \geq 4$ unless NP $\subseteq$ coNP/poly

Jansen and P. [Algorithmica 2017]

- q-Coloring for $q \geq 4$ has no non-trivial sparsification

Open problems

- Factor k-gap between upper- and lower bound
- Sparsification bound for $q = 3$

Previous work

- Fiala et al.: coloring problems parameterized by
  - Vertex Cover vs. Treewidth
- Jansen and Kratsch: parameter hierarchy for graph coloring

Jansen and Kratsch [Inf Comput. 2013] showed that

- q-Coloring parameterized by VC has a kernel of bitsize $O(k^q)$
- But no kernel of size $O(k^{q-1-\varepsilon})$ for $q \geq 4$ unless NP $\subseteq$ coNP/poly

Jansen and P. [Algorithmica 2017]

- q-Coloring for $q \geq 4$ has no non-trivial sparsification

Open problems

- Factor k-gap between upper- and lower bound
- Sparsification bound for $q = 3$

Our results

Theorem

q-Coloring parameterized by Vertex Cover has a kernel with $O(k^{q-1})$ vertices and bitsize $O(k^{q-1} \log k)$.

Theorem

3-Coloring with n vertices has no kernel of bitsize $O(n^{2-\varepsilon})$, unless $\mathrm{NP} \subseteq \mathrm{coNP/poly}$.

Consequently,

- 3-Coloring has no kernel of bitsize $O(k^{2-\varepsilon})$
- q-Coloring has no kernel of bitsize $O(k^{q-1-\varepsilon})$, for $q \geq 3$

Matching upper and lower bound, up to $k^{o(1)}$ factors

Our results

Theorem

q-Coloring parameterized by Vertex Cover has a kernel with $O(k^{q-1})$ vertices and bitsize $O(k^{q-1} \log k)$.

Theorem

3-*Coloring with n vertices has no kernel of bitsize $O(n^{2-\varepsilon})$, unless* NP $\subseteq$ coNP/poly.

Consequently,

- 3-Coloring has no kernel of bitsize $O(k^{2-\varepsilon})$
- q-Coloring has no kernel of bitsize $O(k^{q-1-\varepsilon})$, for $q \geq 3$

Matching upper and lower bound, up to $k^{o(1)}$ factors

Our results

Theorem

q-Coloring parameterized by Vertex Cover has a kernel with $O(k^{q-1})$ vertices and bitsize $O(k^{q-1} \log k)$.

Theorem

3-Coloring with n vertices has no kernel of bitsize $O(n^{2-\varepsilon})$, unless $\mathrm{NP} \subseteq \mathrm{coNP}/\mathrm{poly}$.

Consequently,

- 3-Coloring has no kernel of bitsize $O(k^{2-\varepsilon})$
- q-Coloring has no kernel of bitsize $O(k^{q-1-\varepsilon})$, for $q \geq 3$

Matching upper and lower bound, up to $k^{o(1)}$ factors

Our results

Theorem

q-Coloring parameterized by Vertex Cover has a kernel with $O(k^{q-1})$ vertices and bitsize $O(k^{q-1}\log k)$.

Theorem

3-Coloring with n vertices has no kernel of bitsize $O(n^{2-\varepsilon})$, unless $\mathrm{NP} \subseteq \mathrm{coNP/poly}$.

Consequently,

- 3-Coloring has no kernel of bitsize $O(k^{2-\varepsilon})$
- q-Coloring has no kernel of bitsize $O(k^{q-1-\varepsilon})$, for $q \geq 3$

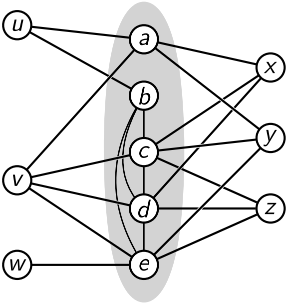
Matching upper and lower bound, up to $k^{o(1)}$ factors

Kernel for q-Coloring

Kernel: general idea

We have graph G, with vertex cover VC
The remaining vertices form independent set IS

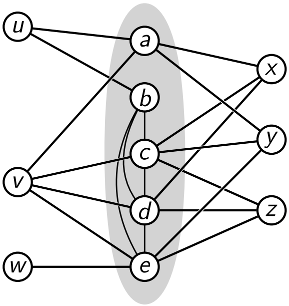
- IS may be large compared to VC
- Find redundant vertices in IS
    - Any coloring of $G - u$ can be extended to G
- Example for 3-coloring: u and w
    - Degree smaller than q
- Similarly, find redundant edges



Kernel: general idea

We have graph G, with vertex cover VC
The remaining vertices form independent set IS

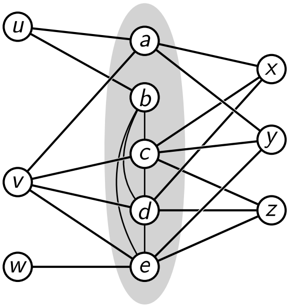
- IS may be large compared to VC
- Find redundant vertices in IS
  - Any coloring of $G - u$ can be extended to G
- Example for 3-coloring: u and w
  - Degree smaller than q
- Similarly, find redundant edges



Kernel: general idea

We have graph G, with vertex cover VC
The remaining vertices form independent set IS

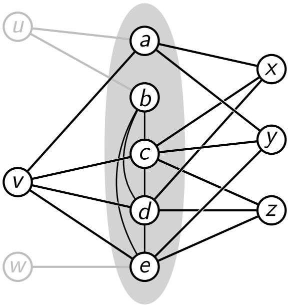
- IS may be large compared to VC
- Find redundant vertices in IS
  - Any coloring of $G - u$ can be extended to G
- Example for 3-coloring: u and w
  - Degree smaller than q
- Similarly, find redundant edges



Kernel: general idea

We have graph G, with vertex cover VC
The remaining vertices form independent set IS

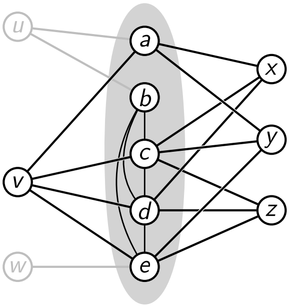
- IS may be large compared to VC
- Find redundant vertices in IS
  - Any coloring of $G - u$ can be extended to G
- Example for 3-coloring: u and w
  - Degree smaller than q
- Similarly, find redundant edges



Kernel: general idea

We have graph G, with vertex cover VC
The remaining vertices form independent set IS

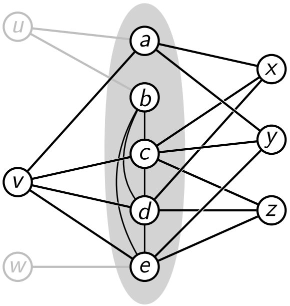
- IS may be large compared to VC
- Find redundant vertices in IS
  - Any coloring of $G - u$ can be extended to G
- Example for 3-coloring: u and w
  - Degree smaller than q
- Similarly, find redundant edges



Kernel: general idea

We have graph G, with vertex cover VC
The remaining vertices form independent set IS

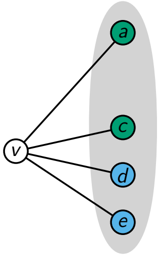
- IS may be large compared to VC
- Find redundant vertices in IS
  - Any coloring of $G - u$ can be extended to G
- Example for 3-coloring: u and w
  - Degree smaller than q
- Similarly, find redundant edges



Kernel: general idea

Vertices in IS can be colored independently

- Each vertex in IS corresponds to a constraint
  - Neighborhood does not use all q colors
- Gives constraints on the coloring of VC
  - If some coloring of VC satisfies all constraints, it can be extended to IS



Alternatively, if for all $S \subseteq N(v)$ with $|S| = q$

Some color is used twice for S

$\Updownarrow$

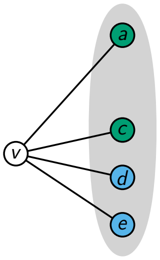
coloring can be extended to v

- Proof by contradiction

Kernel: general idea

Vertices in IS can be colored independently

- Each vertex in IS corresponds to a constraint
  - Neighborhood does not use all q colors
- Gives constraints on the coloring of VC
  - If some coloring of VC satisfies all constraints, it can be extended to IS



Alternatively, if for all $S \subseteq N(v)$ with $|S| = q$

Some color is used twice for S

$\Updownarrow$

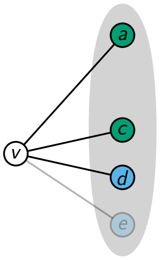
coloring can be extended to v

- Proof by contradiction

Kernel: general idea

Vertices in IS can be colored independently

- Each vertex in IS corresponds to a constraint
  - Neighborhood does not use all q colors
- Gives constraints on the coloring of VC
  - If some coloring of VC satisfies all constraints, it can be extended to IS



Alternatively, if for all $S \subseteq N(v)$ with $|S| = q$

Some color is used twice for S

$$\Updownarrow$$

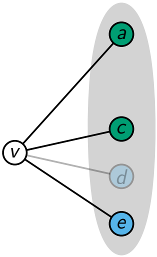
coloring can be extended to v

- Proof by contradiction

Kernel: general idea

Vertices in IS can be colored independently

- Each vertex in IS corresponds to a constraint
  - Neighborhood does not use all q colors
- Gives constraints on the coloring of VC
  - If some coloring of VC satisfies all constraints, it can be extended to IS



Alternatively, if for all $S \subseteq N(v)$ with $|S| = q$

Some color is used twice for S

$\Updownarrow$

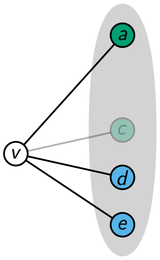
coloring can be extended to v

- Proof by contradiction

Kernel: general idea

Vertices in IS can be colored independently

- Each vertex in IS corresponds to a constraint
  - Neighborhood does not use all q colors
- Gives constraints on the coloring of VC
  - If some coloring of VC satisfies all constraints, it can be extended to IS



Alternatively, if for all $S \subseteq N(v)$ with $|S| = q$

Some color is used twice for S

$$\Updownarrow$$

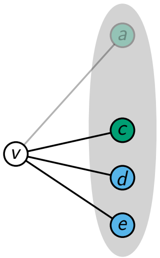
coloring can be extended to v

- Proof by contradiction

Kernel: general idea

Vertices in IS can be colored independently

- Each vertex in IS corresponds to a constraint
  - Neighborhood does not use all q colors
- Gives constraints on the coloring of VC
  - If some coloring of VC satisfies all constraints, it can be extended to IS



Alternatively, if for all $S \subseteq N(v)$ with $|S| = q$

Some color is used twice for S

$$\Updownarrow$$

coloring can be extended to v

- Proof by contradiction

Finding redundant constraints

Let L be a set of polynomial equalities of degree at most d, over n boolean variables.

$$L = \{$$
$$x_1x_2 + x_2x_3 + x_4 \equiv_2 0$$
$$x_1 + 1 \equiv_2 0$$
$$x_3x_1 + 1 \equiv_2 0$$
$$\ldots$$
$$\}$$

Theorem [Jansen and P. MFCS 2016]

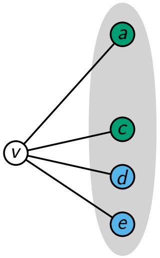
There is a polynomial-time algorithm that outputs $L' \subseteq L$, s.t.

- *An assignment satisfies L if and only if it satisfies L' and*
- $|L'| \leq n^d + 1$

Modeling vertices as constraints

Polynomial equalities

- Create q boolean variables for each vertex in VC.
  - $C_{v,i}$ denotes whether vertex v has color i
- For each vertex v in IS, $S \subseteq N(v)$ with $|S| = q$
  - Constraint: S does not use all q colors.



Which polynomial to use?

- Needs to have degree $\leq q - 1$

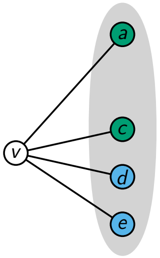
Modeling vertices as constraints

Polynomial equalities

- Create q boolean variables for each vertex in VC.
  - $C_{v,i}$ denotes whether vertex v has color i
- For each vertex v in IS, $S \subseteq N(v)$ with $|S| = q$
  - Constraint: S does not use all q colors.

Which polynomial to use?

- Needs to have degree $\leq q - 1$



Polynomial equalities for 3-Coloring

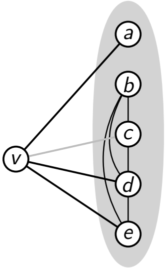
- 3 variables for each $v \in \mathrm{VC}$ v v v

Let $v \in \mathrm{IS}$, for each $S \subseteq N(v) : |S| = q$

- Polynomial equality of degree 2
  - For $S = \{a, d, e\}$:

$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e \equiv_2 1$$

- Expresses: a,d, and e do not use all 3 colors
  - Three equal colors gives $3 \equiv_2 1$
  - Two equal colors gives 1
  - Three different colors gives 0
- Also exists for q-Coloring



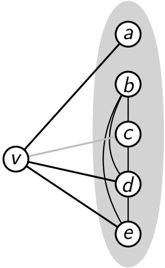
Polynomial equalities for 3-Coloring

- 3 variables for each $v \in \mathrm{VC}$ 

Let $v \in \mathrm{IS}$, for each $S \subseteq N(v) : |S| = q$

- Polynomial equality of degree 2
  - For $S = \{a, d, e\}$:

$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e \equiv_2 1$$



- Expresses: a,d, and e do not use all 3 colors
  - Three equal colors gives $3 \equiv_2 1$
  - Two equal colors gives 1
  - Three different colors gives 0
- Also exists for q-Coloring

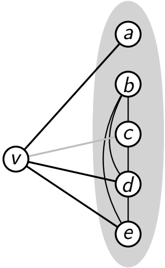
Polynomial equalities for 3-Coloring

- 3 variables for each $v \in \mathrm{VC}$ v v v

Let $v \in \mathrm{IS}$, for each $S \subseteq N(v) : |S| = q$

- Polynomial equality of degree 2
  - For $S = \{a, d, e\}$:

$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e \equiv_2 1$$



- Expresses: a,d, and e do not use all 3 colors
  - Three equal colors gives $3 \equiv_2 1$
  - Two equal colors gives 1
  - Three different colors gives 0
- Also exists for q-Coloring

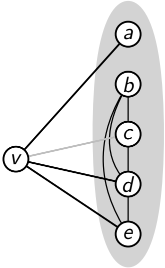
Polynomial equalities for 3-Coloring

- 3 variables for each $v \in \mathrm{VC}$ v v v

Let $v \in \mathrm{IS}$, for each $S \subseteq N(v) : |S| = q$

- Polynomial equality of degree 2
  - For $S = \{a, d, e\}$:

$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e \equiv_2 1$$



- Expresses: a,d, and e do not use all 3 colors
  - Three equal colors gives $3 \equiv_2 1$
  - Two equal colors gives 1
  - Three different colors gives 0
- Also exists for q-Coloring

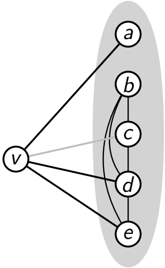
Polynomial equalities for 3-Coloring

- 3 variables for each $v \in \text{VC}$ v v v

Let $v \in \text{IS}$, for each $S \subseteq N(v) : |S| = q$

- Polynomial equality of degree 2
  - For $S = \{a, d, e\}$:

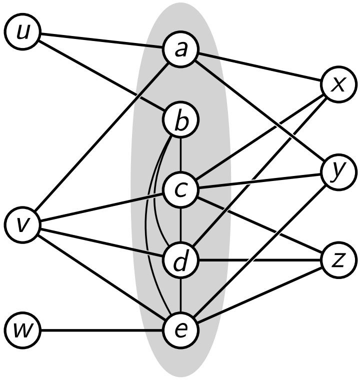
$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e +$$
$$a \wedge d + a \wedge e + d \wedge e \equiv_2 1$$



- Expresses: *a,d, and e do not use all 3 colors*
  - Three equal colors gives $3 \equiv_2 1$
  - Two equal colors gives 1
  - Three different colors gives 0
- Also exists for q-Coloring

Kernel for q-Coloring

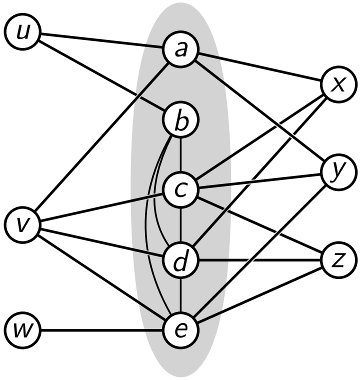
- Model vertices in IS by constraints
- Use Theorem to find subset L' of relevant constraints
- Keep only vertices and edges used for relevant constraints



u, w: $\emptyset$
v:
$C(a, c, d)$
$C(a, d, e)$
$C(c, d, e)$
$C(a, c, e)$
x:
$C(a, c, d)$
y:
$C(a, c, e)$
z:
$C(c, d, e)$

Kernel for q-Coloring

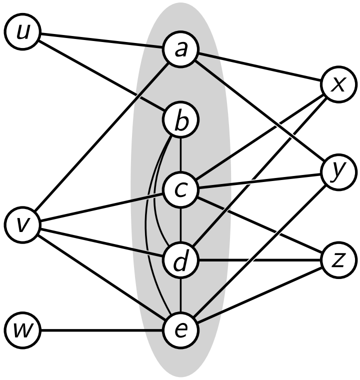
- Model vertices in IS by constraints
- Use Theorem to find subset L' of relevant constraints
- Keep only vertices and edges used for relevant constraints



u, w: $\emptyset$
v:
$C(a, c, d)$
$C(a, d, e)$
$C(c, d, e)$
$C(a, c, e)$
x:
$C(a, c, d)$
y:
$C(a, c, e)$
z:
$C(c, d, e)$

Kernel for q-Coloring

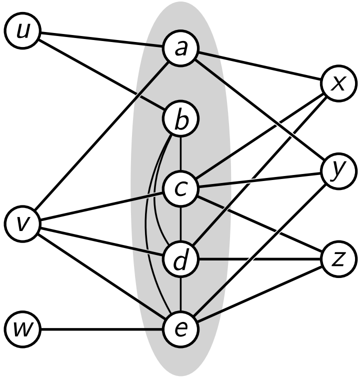
- Model vertices in IS by constraints
- Use Theorem to find subset L' of relevant constraints
- Keep only vertices and edges used for relevant constraints



u, w: $\emptyset$
v:
~~$C(a, c, d)$~~
$C(a, d, e)$
~~$C(c, d, e)$~~
~~$C(a, c, e)$~~
x:
$C(a, c, d)$
y:
$C(a, c, e)$
z:
~~$C(c, d, e)$~~

Kernel for q-Coloring

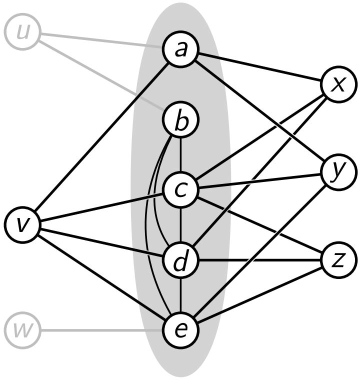
- Model vertices in IS by constraints
- Use Theorem to find subset L' of relevant constraints
- Keep only vertices and edges used for relevant constraints



u, w: $\emptyset$
v:
~~$C(a, c, d)$~~
$C(a, d, e)$
~~$C(c, d, e)$~~
~~$C(a, c, e)$~~
x:
$C(a, c, d)$
y:
$C(a, c, e)$
z:
~~$C(c, d, e)$~~

Kernel for q-Coloring

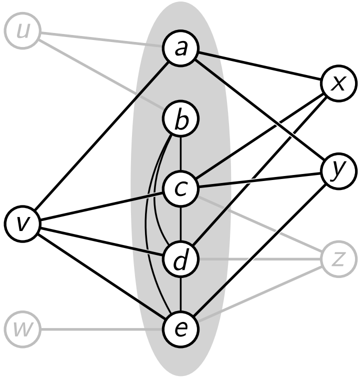
- Model vertices in IS by constraints
- Use Theorem to find subset L' of relevant constraints
- Keep only vertices and edges used for relevant constraints



u, w: $\emptyset$
v:
~~$C(a, c, d)$~~
$C(a, d, e)$
~~$C(c, d, e)$~~
~~$C(a, c, e)$~~
x:
$C(a, c, d)$
y:
$C(a, c, e)$
z:
~~$C(c, d, e)$~~

Kernel for q-Coloring

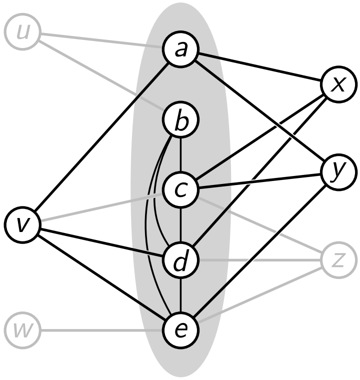
- Model vertices in IS by constraints
- Use Theorem to find subset L' of relevant constraints
- Keep only vertices and edges used for relevant constraints



u, w: $\emptyset$
v:
~~$C(a, c, d)$~~
$C(a, d, e)$
~~$C(c, d, e)$~~
~~$C(a, c, e)$~~
x:
$C(a, c, d)$
y:
$C(a, c, e)$
z:
~~$C(c, d, e)$~~

Kernel for q-Coloring

- Model vertices in IS by constraints
- Use Theorem to find subset L' of relevant constraints
- Keep only vertices and edges used for relevant constraints



u, w: $\emptyset$
v:
~~$C(a, c, d)$~~
$C(a, d, e)$
~~$C(c, d, e)$~~
~~$C(a, c, e)$~~
x:
$C(a, c, d)$
y:
$C(a, c, e)$
z:
~~$C(c, d, e)$~~

Kernel size

- Number of constraints $O(\#vars^{degree})$
  - $q \cdot k$ variables
  - degree $q - 1$
- Number of constraints $O((qk)^{q-1})$
- Constraint corresponds to ≤ 1 vertex and $\leq q$ edges
- Encode graph in $O(k^{q-1} \log k)$ bits for q fixed

Theorem

q-Coloring parameterized by Vertex Cover has a kernel with $O(k^{q-1})$ vertices and bitsize $O(k^{q-1} \log k)$.

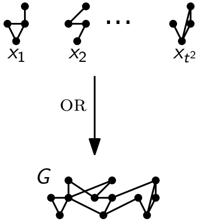
Sparsification lower bound

Lower bound

We show that 3-Coloring has no kernel of size $O(|V(G)|^{2-\varepsilon})$

Degree-2 cross-composition

- Start from NP-hard problem Q
  - Choose Q carefully
- Give a polynomial-time algorithm:
- Input: t^2 similar instances of Q
- Output: Instance G s.t.
  - $|V(G)| = O(t \cdot \max |x_i|^c)$
  - G is 3-colorable if and only if at least one of the inputs a yes-instance



Starting problem

Restricted Coloring with Triangle Split Decomposition

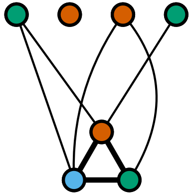
[Jansen and P. Algorithmica 2017]

Input: graph G with partitioning of the vertices $V(G) = S \cup T$

- S is an independent set in G
- $G[T]$ is a disjoint union of triangles

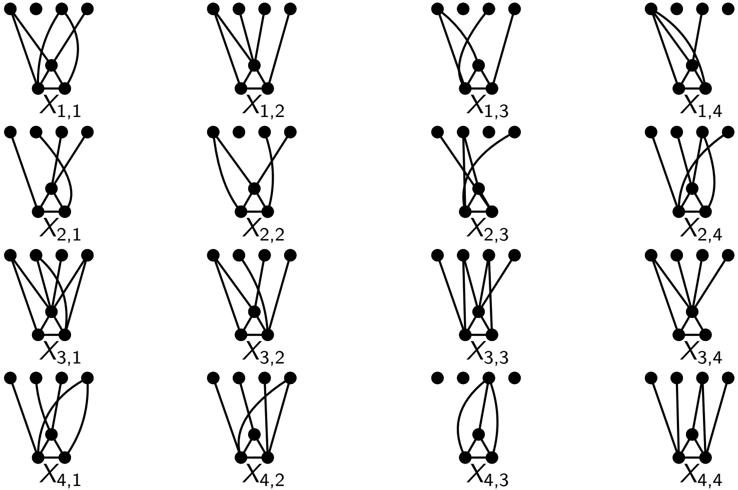
Question: Is there a proper 3-coloring of G (using red,green,blue), such that no vertex in S is colored blue?

We call this a restricted coloring



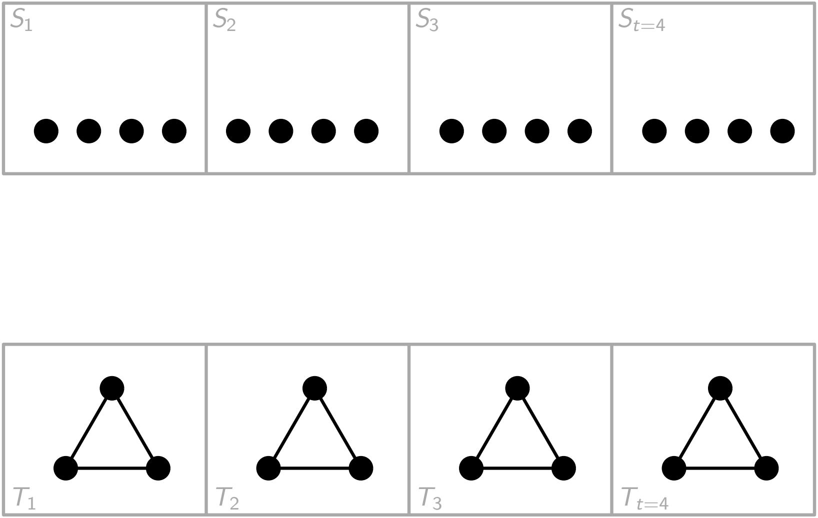
Cross-composition: General strategy

Combine $t^2 = 16$ instances of size n into one of size $O(t \cdot n^{O(1)})$



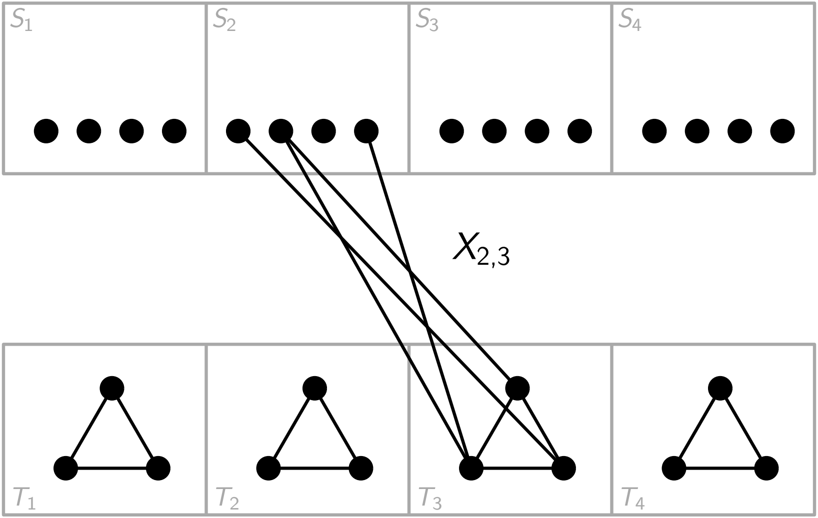
Cross-composition: General strategy

Combine $t^2 = 16$ instances of size n into one of size $O(t \cdot n^{O(1)})$



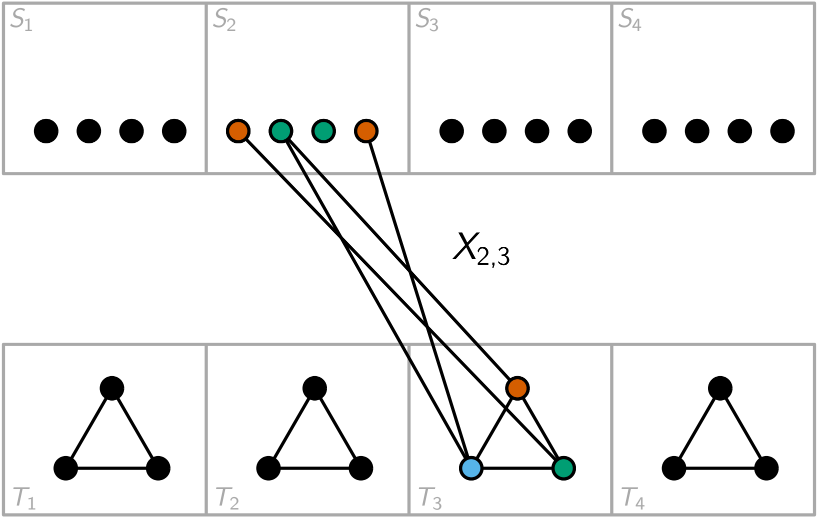
Cross-composition: General strategy

Combine $t^2 = 16$ instances of size n into one of size $O(t \cdot n^{O(1)})$



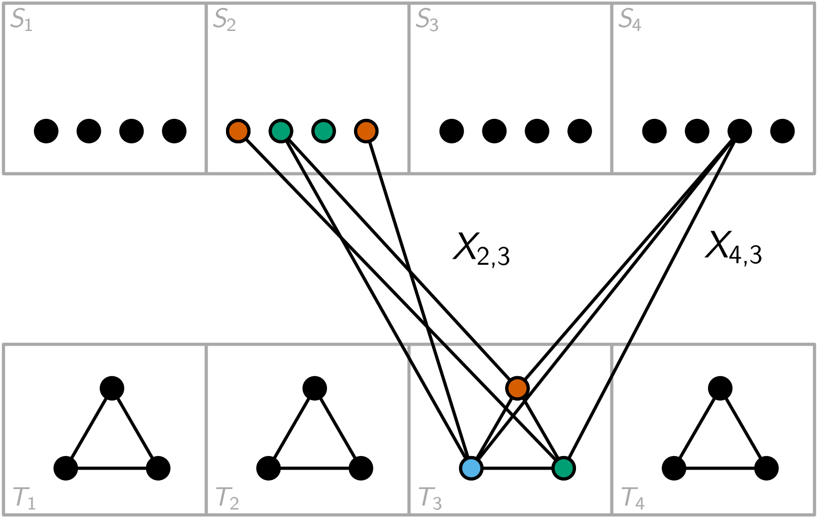
Cross-composition: General strategy

Combine $t^2 = 16$ instances of size n into one of size $O(t \cdot n^{O(1)})$



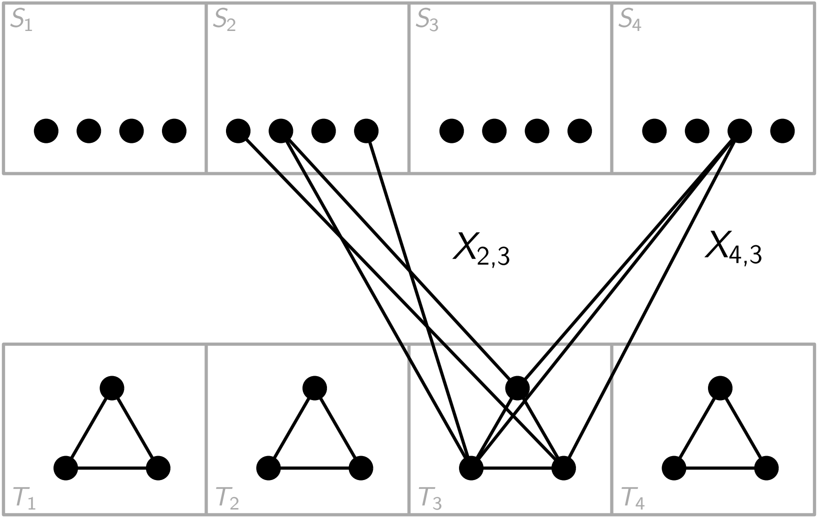
Cross-composition: General strategy

Combine $t^2 = 16$ instances of size n into one of size $O(t \cdot n^{O(1)})$



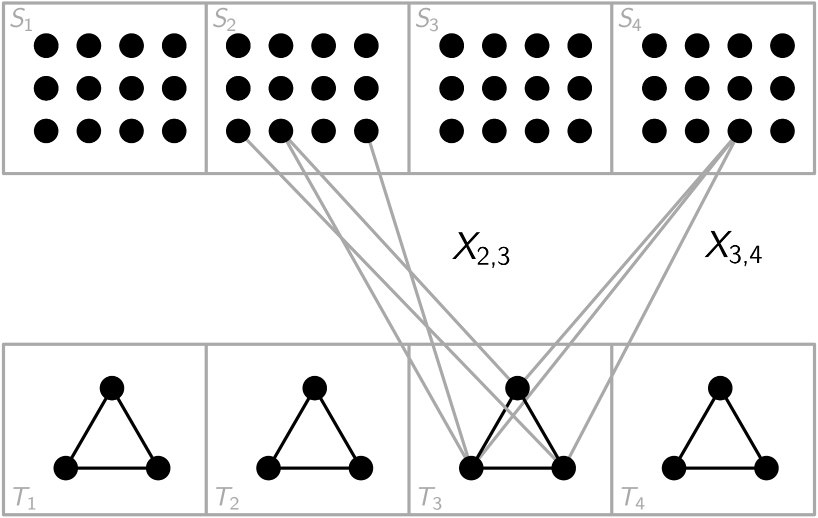
Cross-composition for 3-Coloring

Ensuring OR-property



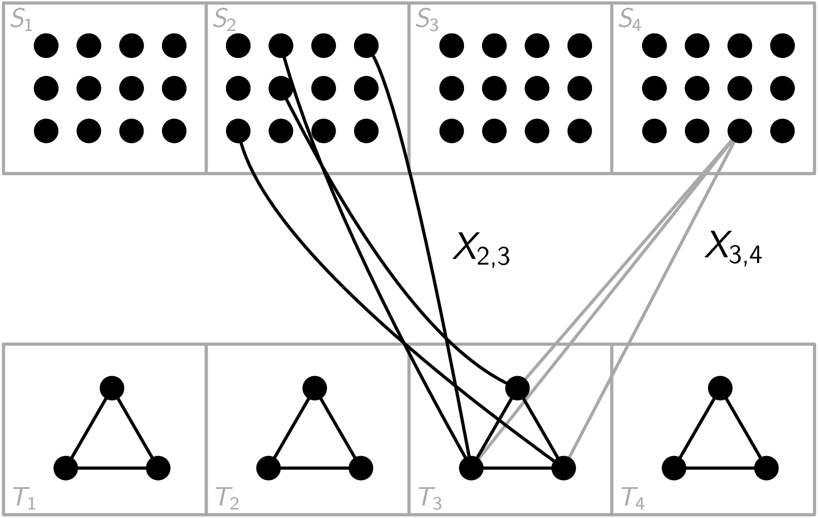
Cross-composition for 3-Coloring

Ensuring OR-property



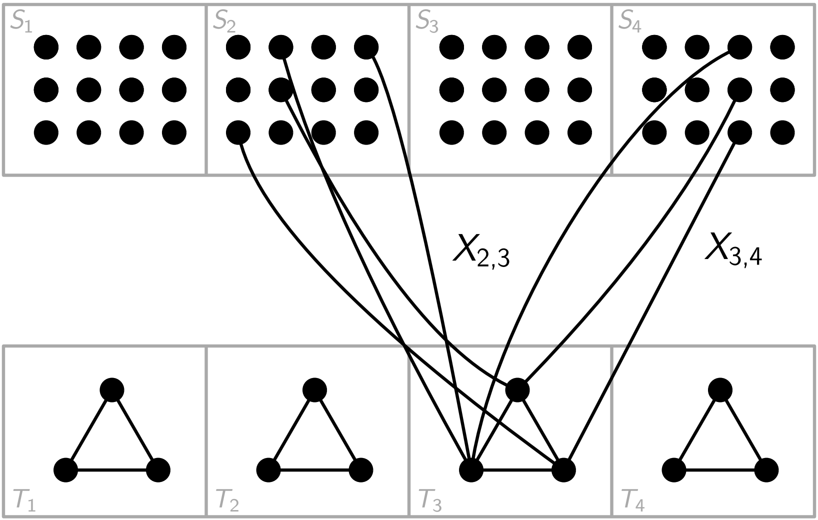
Cross-composition for 3-Coloring

Ensuring OR-property



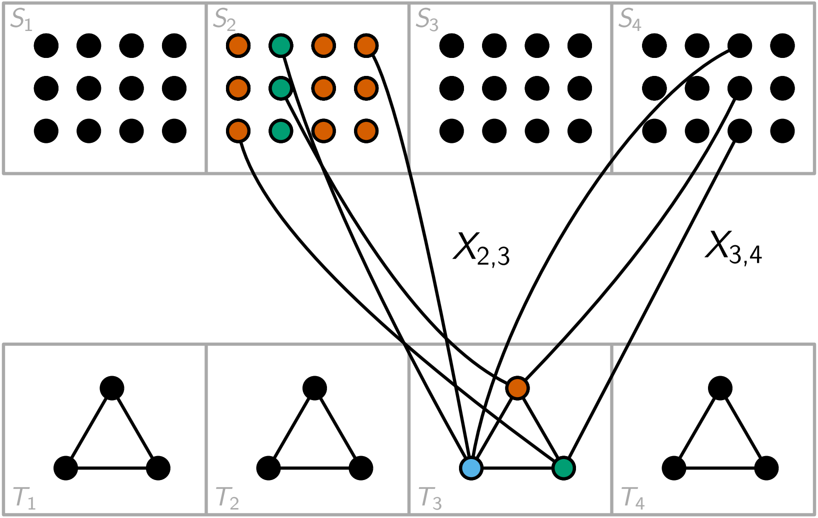
Cross-composition for 3-Coloring

Ensuring OR-property



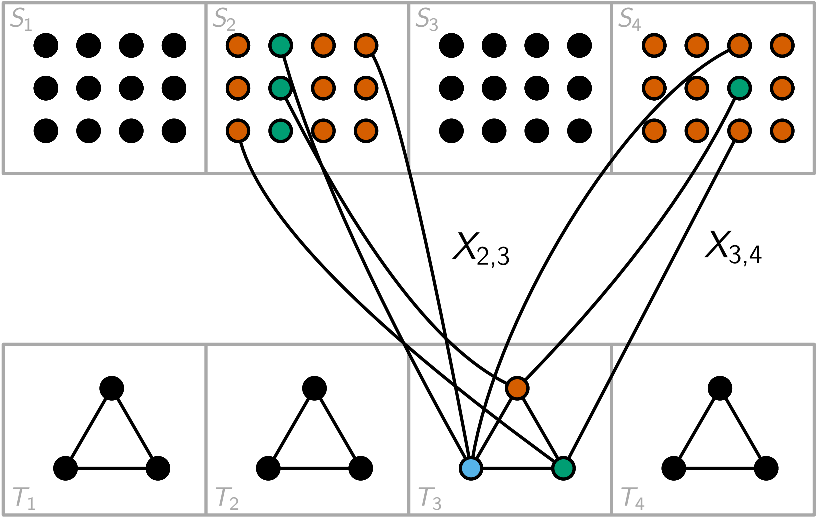
Cross-composition for 3-Coloring

Ensuring OR-property



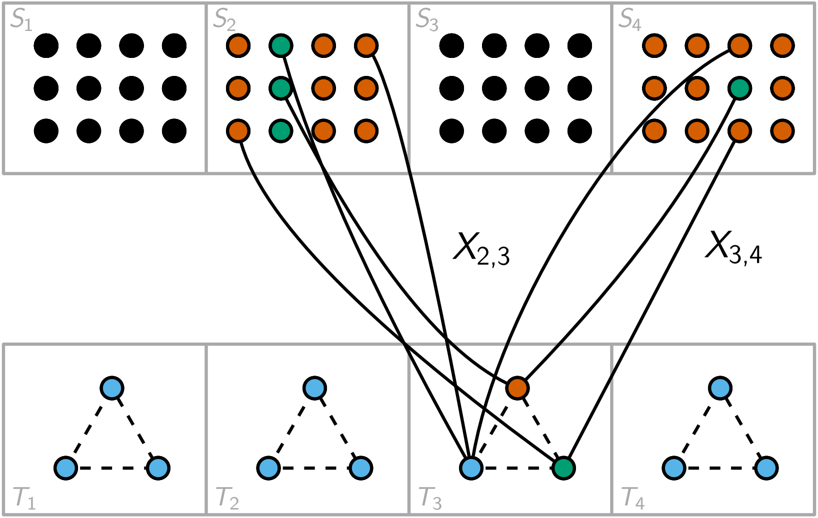
Cross-composition for 3-Coloring

Ensuring OR-property



Cross-composition for 3-Coloring

Ensuring OR-property



Conclusion

q-Coloring has

- a kernel with bitsize $O(k^{q-1} \log k)$
- no kernel with bitsize $O(k^{q-1-\varepsilon})$ unless NP $\subseteq$ coNP/poly

3-Coloring

- has no kernel with bitsize $O(n^{2-\varepsilon})$ unless NP $\subseteq$ coNP/poly

Techniques

- Finding redundant constraints applied to graph problem
- Method of copying vertices for cross-composition

Future work

- Exact kernel bounds for q-Coloring with other parameters
  - Modulator to cograph

Conclusion

q-Coloring has

- a kernel with bitsize $O(k^{q-1} \log k)$
- no kernel with bitsize $O(k^{q-1-\varepsilon})$ unless NP $\subseteq$ coNP/poly

3-Coloring

- has no kernel with bitsize $O(n^{2-\varepsilon})$ unless NP $\subseteq$ coNP/poly

Techniques

- Finding redundant constraints applied to graph problem
- Method of copying vertices for cross-composition

Future work

- Exact kernel bounds for q-Coloring with other parameters
  - Modulator to cograph

Conclusion

q-Coloring has

- a kernel with bitsize $O(k^{q-1} \log k)$
- no kernel with bitsize $O(k^{q-1-\varepsilon})$ unless NP $\subseteq$ coNP/poly

3-Coloring

- has no kernel with bitsize $O(n^{2-\varepsilon})$ unless NP $\subseteq$ coNP/poly

Techniques

- Finding redundant constraints applied to graph problem
- Method of copying vertices for cross-composition

Future work

- Exact kernel bounds for q-Coloring with other parameters
  - Modulator to cograph

Conclusion

q-Coloring has

- a kernel with bitsize $O(k^{q-1} \log k)$
- no kernel with bitsize $O(k^{q-1-\varepsilon})$ unless NP $\subseteq$ coNP/poly

3-Coloring

- has no kernel with bitsize $O(n^{2-\varepsilon})$ unless NP $\subseteq$ coNP/poly

Techniques

- Finding redundant constraints applied to graph problem
- Method of copying vertices for cross-composition

Future work

- Exact kernel bounds for q-Coloring with other parameters
  - Modulator to cograph

Thank you!

Conclusion

q-Coloring has

- a kernel with bitsize $O(k^{q-1} \log k)$
- no kernel with bitsize $O(k^{q-1-\varepsilon})$ unless NP $\subseteq$ coNP/poly

3-Coloring

- has no kernel with bitsize $O(n^{2-\varepsilon})$ unless NP $\subseteq$ coNP/poly

Techniques

- Finding redundant constraints applied to graph problem
- Method of copying vertices for cross-composition

Future work

- Exact kernel bounds for q-Coloring with other parameters
  - Modulator to cograph

Thank you!

Polynomial for q-Coloring

For 3-Coloring:

$$a \wedge d + a \wedge e + d \wedge e +$$
$$d \wedge a + e \wedge a + e \wedge d \equiv_2 0$$

Formula:

$$y_{1,1} \cdot y_{2,2} + y_{1,1} \cdot y_{3,2} + y_{2,1} \cdot y_{1,2} + y_{2,1} \cdot y_{3,2} + y_{3,1} \cdot y_{1,2} + y_{3,1} \cdot y_{2,2} \equiv_2 0$$

In general:

$$\sum_{\substack{i_1,\dots,i_{q-1} \in [q] \\ \text{distinct}}} \prod_{k=1}^{q-1} y_{i_k,k} \equiv_2 0$$