On some graph modification problems

BERNARD RIES

Decision Support & Operations Research
Department of Informatics
Université de Fribourg, Switzerland

ECCO 2017

Results presented here were obtained in joint work with:

- Oznur Dinner
  Kadir Has University, Turkey
- Daniel Paulusma
  Durham University, United Kingdom
- Christophe Picouleau
  Conservatoire National des Arts et Métiers, France

TYPICAL GRAPH MODIFICATION PROBLEM

Usually, in a graph modification problem we aim to

- modify a given graph G via a small number of operations

TYPICAL GRAPH MODIFICATION PROBLEM

Usually, in a graph modification problem we aim to

- modify a given graph G via a small number of operations
- into some other graph H that has some desired property,

TYPICAL GRAPH MODIFICATION PROBLEM

Usually, in a graph modification problem we aim to

- modify a given graph G via a small number of operations
- into some other graph H that has some desired property,
- often describing a graph class to which H must belong.

TYPICAL GRAPH MODIFICATION PROBLEM

Usually, in a graph modification problem we aim to

- modify a given graph G via a small number of operations
- into some other graph H that has some desired property,
- often describing a graph class to which H must belong.

Examples:

- What is the fewest number of edges to be deleted from a graph G to make it become a permutation graph? [Burzyn et al., 2006]

TYPICAL GRAPH MODIFICATION PROBLEM

Usually, in a graph modification problem we aim to

- modify a given graph G via a small number of operations
- into some other graph H that has some desired property,
- often describing a graph class to which H must belong.

Examples:

- What is the fewest number of edges to be deleted from a graph G to make it become a permutation graph? [Burzyn et al., 2006]
- What is the fewest number of edges to be added to a graph G to make it become chordal bipartite? [Sritharan, 2016]

TYPICAL GRAPH MODIFICATION PROBLEM

Usually, in a graph modification problem we aim to

- modify a given graph G via a small number of operations
- into some other graph H that has some desired property,
- often describing a graph class to which H must belong.

Examples:

- What is the fewest number of edges to be deleted from a graph G to make it become a permutation graph? [Burzyn et al., 2006]
- What is the fewest number of edges to be added to a graph G to make it become chordal bipartite? [Sritharan, 2016]
- Is it possible to make a graph bipartite with at most k edge contractions? [Heggernes et al., 2013]

TYPICAL GRAPH MODIFICATION PROBLEM

A variety of classical graph-theoretic problems can be captured that way.

TYPICAL GRAPH MODIFICATION PROBLEM

A variety of classical graph-theoretic problems can be captured that way.

- If only k vertex deletions are allowed and H must be an independent set

$$\Rightarrow \texttt{INDEPENDENT SET}$$

TYPICAL GRAPH MODIFICATION PROBLEM

A variety of classical graph-theoretic problems can be captured that way.

- If only k vertex deletions are allowed and H must be an independent set

  $\Rightarrow$ INDEPENDENT SET

- If only k vertex deletions are allowed and H must be a clique

  $\Rightarrow$ CLIQUE

TYPICAL GRAPH MODIFICATION PROBLEM

A variety of classical graph-theoretic problems can be captured that way.

- If only k vertex deletions are allowed and H must be an independent set

  $$\Rightarrow \texttt{INDEPENDENT SET}$$

- If only k vertex deletions are allowed and H must be a clique

  $$\Rightarrow \texttt{CLIQUE}$$

- $\cdots$

BLOCKER PROBLEMS

Instead of specifying a graph class to which H must belong, we may specify a graph parameter π and then,

BLOCKER PROBLEMS

Instead of specifying a graph class to which H must belong, we may specify a graph parameter π and then,

- given a graph G, a set S of one or more graph operations, an integer k, we aim to

BLOCKER PROBLEMS

Instead of specifying a graph class to which H must belong, we may specify a graph parameter π and then,

- given a graph G, a set S of one or more graph operations, an integer k, we aim to
  - transform G into a graph G' using at most k operations from S,

BLOCKER PROBLEMS

Instead of specifying a graph class to which H must belong, we may specify a graph parameter π and then,

- given a graph G, a set S of one or more graph operations, an integer k, we aim to
  - transform G into a graph G' using at most k operations from S,
  - such that $\pi(G') \leq \pi(G) - d$ for some threshold $d \geq 0$.

BLOCKER PROBLEMS

Instead of specifying a graph class to which H must belong, we may specify a graph parameter π and then,

- given a graph G, a set S of one or more graph operations, an integer k, we aim to
  - transform G into a graph G' using at most k operations from S,
  - such that $\pi(G') \leq \pi(G) - d$ for some threshold $d \geq 0$.

Such problems are called blocker problems.

BLOCKER PROBLEMS

Instead of specifying a graph class to which H must belong, we may specify a graph parameter π and then,

- given a graph G, a set S of one or more graph operations, an integer k, we aim to
  - transform G into a graph G' using at most k operations from S,
  - such that $\pi(G') \leq \pi(G) - d$ for some threshold $d \geq 0$.

Such problems are called blocker problems.

- The edges/vertices involved can be viewed as "blocking" the parameter π.

BLOCKER PROBLEMS

Instead of specifying a graph class to which H must belong, we may specify a graph parameter π and then,

- given a graph G, a set S of one or more graph operations, an integer k, we aim to
  - transform G into a graph G' using at most k operations from S,
  - such that $\pi(G') \leq \pi(G) - d$ for some threshold $d \geq 0$.

Such problems are called blocker problems.

- The edges/vertices involved can be viewed as "blocking" the parameter π.
- Identifying such sets may have some nice applications.

BLOCKER PROBLEMS - APPLICATIONS[1]

- Consider for instance the blocker problem with
  - $\pi = \omega$ and $S = \{\text{vertex deletion}\}$.

[1] from [Boginski et al., 2014]

BLOCKER PROBLEMS - APPLICATIONS[1]

- Consider for instance the blocker problem with
  - $\pi = \omega$ and $S = \{\text{vertex deletion}\}$.

- So, we are interested in finding a subset of k vertices to be deleted, in order to make the clique number decrease by at least d units.

[1] from [Boginski et al., 2014]

BLOCKER PROBLEMS - APPLICATIONS[1]

- Consider for instance the blocker problem with
  - $\pi = \omega$ and $S = \{\text{vertex deletion}\}$.

- So, we are interested in finding a subset of k vertices to be deleted, in order to make the clique number decrease by at least d units.

- In a (social or biological or communication or ...) network, a clique represents a cohesive cluster.

[1] from [Boginski et al., 2014]

BLOCKER PROBLEMS - APPLICATIONS[1]

- Consider for instance the blocker problem with
  - $\pi = \omega$ and $S = \{\text{vertex deletion}\}$.
- So, we are interested in finding a subset of k vertices to be deleted, in order to make the clique number decrease by at least d units.
- In a (social or biological or communication or ...) network, a clique represents a cohesive cluster.
  - Attacker's objective: delete a small set of vertices to restrict the size of a largest cohesive cluster in the remaining graph.

[1] from [Boginski et al., 2014]

BLOCKER PROBLEMS - APPLICATIONS[1]

- Consider for instance the blocker problem with
  - $\pi = \omega$ and $S = \{\text{vertex deletion}\}$.
- So, we are interested in finding a subset of k vertices to be deleted, in order to make the clique number decrease by at least d units.
- In a (social or biological or communication or ...) network, a clique represents a cohesive cluster.
  - Attacker's objective: delete a small set of vertices to restrict the size of a largest cohesive cluster in the remaining graph.
  - Defender's objective: identify a set of vertices whose deletion would substantially decrease the size of largest cohesive cluster, in order to protect these vertices.

[1] from [Boginski et al., 2014]

BLOCKER PROBLEMS

There exist several relations to other graph problems:

BLOCKER PROBLEMS

There exist several relations to other graph problems:

- HADWIGER NUMBER: given a graph G and an integer r, does G contain K_r as a minor? [Golovach et al., 2014]

BLOCKER PROBLEMS

There exist several relations to other graph problems:

- HADWIGER NUMBER: given a graph G and an integer r, does G contain K_r as a minor? [Golovach et al., 2014]
  - $\pi = \alpha$, $S = \{\text{edge contraction}\}$, $k = n - r$, $d = \alpha(G) - 1$.

BLOCKER PROBLEMS

There exist several relations to other graph problems:

- HADWIGER NUMBER: given a graph G and an integer r, does G contain K_r as a minor? [Golovach et al., 2014]
  - $\pi = \alpha$, $S = \{\text{edge contraction}\}$, $k = n - r$, $d = \alpha(G) - 1$.
- s-CLUB CONTRACTION: given a graph G, an integer k and a fixed integer s, can G be k-contracted into a graph with diameter at most s? [Golovach et al., 2014]

BLOCKER PROBLEMS

There exist several relations to other graph problems:

- HADWIGER NUMBER: given a graph G and an integer r, does G contain K_r as a minor? [Golovach et al., 2014]
  - $\pi = \alpha$, $S = \{\text{edge contraction}\}$, $k = n - r$, $d = \alpha(G) - 1$.
- s-CLUB CONTRACTION: given a graph G, an integer k and a fixed integer s, can G be k-contracted into a graph with diameter at most s? [Golovach et al., 2014]
  - for $s = 1$: $\pi = \alpha$, $S = \{\text{edge contraction}\}$, $d = \alpha(G) - 1$.

BLOCKER PROBLEMS

There exist several relations to other graph problems:

- HADWIGER NUMBER: given a graph G and an integer r, does G contain K_r as a minor? [Golovach et al., 2014]
  - $\pi = \alpha$, $S = \{\text{edge contraction}\}$, $k = n - r$, $d = \alpha(G) - 1$.
- s-CLUB CONTRACTION: given a graph G, an integer k and a fixed integer s, can G be k-contracted into a graph with diameter at most s? [Golovach et al., 2014]
  - for $s = 1$: $\pi = \alpha$, $S = \{\text{edge contraction}\}$, $d = \alpha(G) - 1$.
- VERTEX COVER: given a graph G and an integer k, does G contain a subset V' of at most k vertices such that each edge has at least one endvertex in V'?

BLOCKER PROBLEMS

There exist several relations to other graph problems:

- HADWIGER NUMBER: given a graph G and an integer r, does G contain K_r as a minor? [Golovach et al., 2014]
  - $\pi = \alpha$, $S = \{\text{edge contraction}\}$, $k = n - r$, $d = \alpha(G) - 1$.
- s-CLUB CONTRACTION: given a graph G, an integer k and a fixed integer s, can G be k-contracted into a graph with diameter at most s? [Golovach et al., 2014]
  - for $s = 1$: $\pi = \alpha$, $S = \{\text{edge contraction}\}$, $d = \alpha(G) - 1$.
- VERTEX COVER: given a graph G and an integer k, does G contain a subset V' of at most k vertices such that each edge has at least one endvertex in V'?
  - for triangle-free graphs: $\pi = \omega$, $S = \{\text{vertex deletion}\}$, $d = 1$.

π and S

- Graph parameters considered so far:
  - independence number α;
  - chromatic number χ;
  - clique number ω;
  - vertex cover number β;
  - matching number μ.

π and S

- Graph parameters considered so far:
  - independence number α;
  - chromatic number χ;
  - clique number ω;
  - vertex cover number β;
  - matching number μ.
- Set S always consisted of a single operation:
  - vertex deletion;
  - edge deletion;
  - edge addition.

MATCHING NUMBER μ

- $\pi = \mu$;
- S={edge deletion}

MATCHING NUMBER μ

- $\pi = \mu$;
- S={edge deletion}

Given a graph G and an integer k, is it possible to find k edges to delete such that the resulting graph G' satisfies $\mu(G') \leq \mu(G) - d$?

MATCHING NUMBER μ

- $\pi = \mu$;
- S={edge deletion}

Given a graph G and an integer k, is it possible to find k edges to delete such that the resulting graph G' satisfies $\mu(G') \leq \mu(G) - d$?

Theorem [Bentz et al., 2010]

Even if d is fixed, the problem is NP-complete in bipartite graphs. But it is polynomial-time solvable in grid graphs and graphs of bounded treewidth.

INDEPENDENCE NUMBER α

- $\pi = \alpha$;
- S={vertex deletion}

INDEPENDENCE NUMBER α

- $\pi = \alpha$;
- S={vertex deletion}

Given a graph G and an integer k, is it possible to find k vertices to delete such that the resulting graph G' satisfies $\alpha(G') \leq \alpha(G) - d$?

INDEPENDENCE NUMBER α

- $\pi = \alpha$;
- S={vertex deletion}

Given a graph G and an integer k, is it possible to find k vertices to delete such that the resulting graph G' satisfies $\alpha(G') \leq \alpha(G) - d$?

Theorem [Costa et al., 2010]

The problem is polynomial-time solvable in bipartite graphs.

CHROMATIC NUMBER χ

- $\pi = \chi$;
- S={edge deletion}

CHROMATIC NUMBER χ

- $\pi = \chi$;
- S={edge deletion}

Given a graph G and an integer k, is it possible to find k edges to delete such that the resulting graph G' satisfies $\chi(G') \leq \chi(G) - d$?

CHROMATIC NUMBER χ

- $\pi = \chi$;
- S={edge deletion}

Given a graph G and an integer k, is it possible to find k edges to delete such that the resulting graph G' satisfies $\chi(G') \leq \chi(G) - d$?

Theorem [Bazgan et al., 2015]

The problem is polynomial-time solvable in threshold graphs. If d is fixed it is polynomial-time solvable in split graphs.

EDGE CONTRACTION

We will focus on two operations:

- vertex deletion and

EDGE CONTRACTION

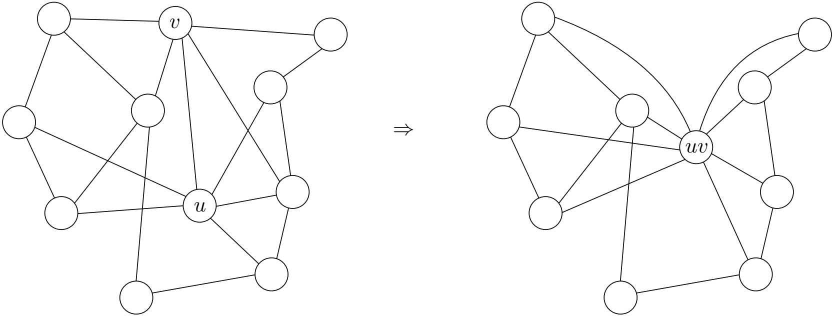
We will focus on two operations:

- vertex deletion and
- edge contraction.

EDGE CONTRACTION

We will focus on two operations:

- vertex deletion and
- edge contraction.



k-CONTRACTED, k-VERTEX-DELETED

k-contracted

Let G, G' be two graphs. We say that G can be k-contracted into the graph G' if G can be modified into G' by a sequence of at most k edge contractions.

k-CONTRACTED, k-VERTEX-DELETED

k-contracted

Let G, G' be two graphs. We say that G can be k-contracted into the graph G' if G can be modified into G' by a sequence of at most k edge contractions.

k-vertex-deleted

Let G, G' be two graphs. We say that G can be k-vertex-deleted into the graph G' if G can be modified into G' by a sequence of at most k vertex deletions.

PROBLEMS WITH $\pi \in \{\alpha, \omega, \chi\}$

CONTRACTION BLOCKER(π)
Instance: a graph G and two integers $d, k \geq 0$
Question: can G be k-contracted into G' with $\pi(G') \leq \pi(G) - d$?

PROBLEMS WITH $\pi \in \{\alpha, \omega, \chi\}$

CONTRACTION BLOCKER(π)

Instance: a graph G and two integers $d, k \geq 0$

Question: can G be k-contracted into G' with $\pi(G') \leq \pi(G) - d$?

DELETION BLOCKER(π)

Instance: a graph G and two integers $d, k \geq 0$

Question: can G be k-vertex-deleted in G' with $\pi(G') \leq \pi(G) - d$?

PROBLEMS WITH $\pi \in \{\alpha, \omega, \chi\}$

CONTRACTION BLOCKER(π)

Instance: a graph G and two integers $d, k \geq 0$

Question: can G be k-contracted into G' with $\pi(G') \leq \pi(G) - d$?

DELETION BLOCKER(π)

Instance: a graph G and two integers $d, k \geq 0$

Question: can G be k-vertex-deleted in G' with $\pi(G') \leq \pi(G) - d$?

- If d is not part of the input, but fixed instead, the corresponding problems will be denoted by:
  - d-CONTRACTION BLOCKER(π)
  - d-DELETION BLOCKER(π)

BIPARTITE GRAPHS

- If $\pi \in \{\omega, \chi\}$, CONTRACTION BLOCKER(π) is trivial in bipartite graphs.

BIPARTITE GRAPHS

- If $\pi \in \{\omega, \chi\}$, CONTRACTION BLOCKER(π) is trivial in bipartite graphs.
- For $\pi = \alpha$, it turns out that CONTRACTION BLOCKER(α) is NP-hard for bipartite graphs.

Theorem [Paulusma, Picouleau, R., 2017]

CONTRACTION BLOCKER(α) is NP-hard for bipartite graphs.

BIPARTITE GRAPHS

- If $\pi \in \{\omega, \chi\}$, CONTRACTION BLOCKER(π) is trivial in bipartite graphs.
- For $\pi = \alpha$, it turns out that CONTRACTION BLOCKER(α) is NP-hard for bipartite graphs.

Theorem [Paulusma, Picouleau, R., 2017]

CONTRACTION BLOCKER(α) is NP-hard for bipartite graphs.

- We use a reduction from 1-CONTRACTION BLOCKER(α), which is NP-complete on cobipartite graphs.

BIPARTITE GRAPHS

- If $\pi \in \{\omega, \chi\}$, CONTRACTION BLOCKER(π) is trivial in bipartite graphs.
- For $\pi = \alpha$, it turns out that CONTRACTION BLOCKER(α) is NP-hard for bipartite graphs.

Theorem [Paulusma, Picouleau, R., 2017]

CONTRACTION BLOCKER(α) is NP-hard for bipartite graphs.

- We use a reduction from 1-CONTRACTION BLOCKER(α), which is NP-complete on cobipartite graphs.
- Notice that bipartite graphs are not closed under edge contraction; therefore membership to NP cannot be established.

TREES

Theorem [Paulusma, Picouleau, R., 2017]

CONTRACTION BLOCKER(α) is linear-time solvable on trees.

TREES

Theorem [Paulusma, Picouleau, R., 2017]

CONTRACTION BLOCKER(α) is linear-time solvable on trees.

Idea of the proof:

- Notice the following:
  - $\alpha(T) + \mu(T) = n$, where n is the number of vertices in T; thus, if $d > n - \mu(T)$, the answer is NO.

TREES

Theorem [Paulusma, Picouleau, R., 2017]

CONTRACTION BLOCKER(α) is linear-time solvable on trees.

Idea of the proof:

- Notice the following:
  - $\alpha(T) + \mu(T) = n$, where n is the number of vertices in T; thus, if $d > n - \mu(T)$, the answer is NO.
  - Trees are closed under edge contraction!

TREES

Theorem [Paulusma, Picouleau, R., 2017]

CONTRACTION BLOCKER(α) is linear-time solvable on trees.

Idea of the proof:

- Notice the following:
  - $\alpha(T) + \mu(T) = n$, where n is the number of vertices in T; thus, if $d > n - \mu(T)$, the answer is NO.
  - Trees are closed under edge contraction!
- Contracting an edge: $T \Rightarrow T'$.

TREES

Theorem [Paulusma, Picouleau, R., 2017]

CONTRACTION BLOCKER(α) is linear-time solvable on trees.

Idea of the proof:

- Notice the following:
  - $\alpha(T) + \mu(T) = n$, where n is the number of vertices in T; thus, if $d > n - \mu(T)$, the answer is NO.
  - Trees are closed under edge contraction!
- Contracting an edge: $T \Rightarrow T'$.
- An edge contraction does not increase α or μ.

TREES

Theorem [Paulusma, Picouleau, R., 2017]

CONTRACTION BLOCKER(α) is linear-time solvable on trees.

Idea of the proof:

- Notice the following:
  - $\alpha(T) + \mu(T) = n$, where n is the number of vertices in T; thus, if $d > n - \mu(T)$, the answer is NO.
  - Trees are closed under edge contraction!
- Contracting an edge: $T \Rightarrow T'$.
- An edge contraction does not increase α or μ.
- Since $\alpha(T) + \mu(T) = n$ and $\alpha(T') + \mu(T') = n - 1$, it follows that:
  - either $\alpha(T') = \alpha(T) - 1$,
  - or $\mu(T') = \mu(T) - 1$.

TREES

- Suppose that $d \leq n - 2\mu(T)$.

TREES

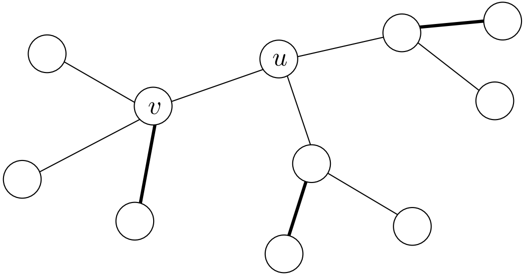
- Suppose that $d \leq n - 2\mu(T)$.
  - $n - 2\mu(T)$ vertices are unsaturated by a maximum matching M.

TREES

- Suppose that $d \leq n - 2\mu(T)$.
  - $n - 2\mu(T)$ vertices are unsaturated by a maximum matching M.
  - For any edge uv, such that u is unsaturated, v must be saturated since M is maximum.

TREES

- Suppose that $d \leq n - 2\mu(T)$.
  - $n - 2\mu(T)$ vertices are unsaturated by a maximum matching M.
  - For any edge uv, such that u is unsaturated, v must be saturated since M is maximum.
  - Contracting such an edge $\Rightarrow$ T' such that $\mu(T') = \mu(T)$.



TREES

- Suppose that $d \leq n - 2\mu(T)$.
  - $n - 2\mu(T)$ vertices are unsaturated by a maximum matching M.
  - For any edge uv, such that u is unsaturated, v must be saturated since M is maximum.
  - Contracting such an edge $\Rightarrow$ T' such that $\mu(T') = \mu(T)$.
  - Hence, $\alpha(T') = \alpha(T) - 1$.

TREES

- Suppose that $d \leq n - 2\mu(T)$.
  - $n - 2\mu(T)$ vertices are unsaturated by a maximum matching M.
  - For any edge uv, such that u is unsaturated, v must be saturated since M is maximum.
  - Contracting such an edge $\Rightarrow$ T' such that $\mu(T') = \mu(T)$.
  - Hence, $\alpha(T') = \alpha(T) - 1$.
  - $\Rightarrow$ if $d \leq n - 2\mu(T)$, contracting d such edges yields a tree T' with $\alpha(T') = \alpha(T) - d$.

TREES

- Suppose that $d \leq n - 2\mu(T)$.
  - $n - 2\mu(T)$ vertices are unsaturated by a maximum matching M.
  - For any edge uv, such that u is unsaturated, v must be saturated since M is maximum.
  - Contracting such an edge $\Rightarrow$ T' such that $\mu(T') = \mu(T)$.
  - Hence, $\alpha(T') = \alpha(T) - 1$.
  - $\Rightarrow$ if $d \leq n - 2\mu(T)$, contracting d such edges yields a tree T' with $\alpha(T') = \alpha(T) - d$.
  - $\Rightarrow$ if $k \geq d$, the answer is YES, otherwise the answer is NO.

TREES

- Suppose that $d > n - 2\mu(T)$.

TREES

- Suppose that $d > n - 2\mu(T)$.
  - First contract the $n - 2\mu(T)$ edges with exactly one endvertex that is unsaturated by M.

TREES

- Suppose that $d > n - 2\mu(T)$.
  - First contract the $n - 2\mu(T)$ edges with exactly one endvertex that is unsaturated by M.
  - $\Rightarrow$ tree T' with $\mu(T') = \mu(T)$ and $\alpha(T') = \alpha(T) - (n - 2\mu(T))$.

TREES

- Suppose that $d > n - 2\mu(T)$.
  - First contract the $n - 2\mu(T)$ edges with exactly one endvertex that is unsaturated by M.
  - $\Rightarrow$ tree T' with $\mu(T') = \mu(T)$ and $\alpha(T') = \alpha(T) - (n - 2\mu(T))$.
  - M is now a perfect matching in T'.

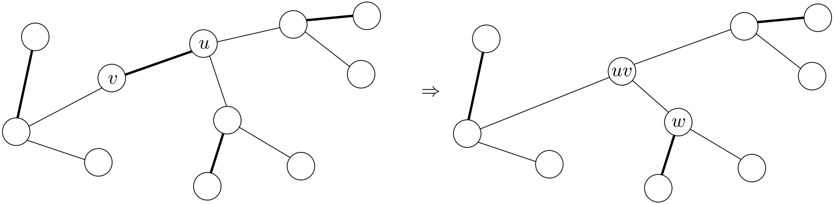
TREES

- Suppose that $d > n - 2\mu(T)$.
  - First contract the $n - 2\mu(T)$ edges with exactly one endvertex that is unsaturated by M.
  - $\Rightarrow$ tree T' with $\mu(T') = \mu(T)$ and $\alpha(T') = \alpha(T) - (n - 2\mu(T))$.
  - M is now a perfect matching in T'.
  - Hence, contracting any edge in T' will result in a tree T'' with $\mu(T'') = \mu(T') - 1$ and $\alpha(T'') = \alpha(T')$.

TREES

- Suppose that $d > n - 2\mu(T)$.
  - First contract the $n - 2\mu(T)$ edges with exactly one endvertex that is unsaturated by M.
  - $\Rightarrow$ tree T' with $\mu(T') = \mu(T)$ and $\alpha(T') = \alpha(T) - (n - 2\mu(T))$.
  - M is now a perfect matching in T'.
  - Hence, contracting any edge in T' will result in a tree T'' with $\mu(T'') = \mu(T') - 1$ and $\alpha(T'') = \alpha(T')$.
  - But if we contract an edge $uv \in M$, the new vertex becomes unsaturated by $M' = M \setminus \{uv\}$.

TREES



TREES

- Suppose that $d > n - 2\mu(T)$.
  - First contract the $n - 2\mu(T)$ edges with exactly one endvertex that is unsaturated by M.
  - $\Rightarrow$ tree T' with $\mu(T') = \mu(T)$ and $\alpha(T') = \alpha(T) - (n - 2\mu(T))$.
  - M is now a perfect matching in T'.
  - Hence, contracting any edge in T' will result in a tree T'' with $\mu(T'') = \mu(T') - 1$ and $\alpha(T'') = \alpha(T')$.
  - But if we contract an edge $uv \in M$, the new vertex becomes unsaturated by $M' = M \setminus \{uv\}$.
  - Now we can contract an edge $(uv)w$ and obtain T''' with $\mu(T''') = \mu(T'')$ and $\alpha(T''') = \alpha(T'') - 1$.

TREES

- Suppose that $d > n - 2\mu(T)$.
  - First contract the $n - 2\mu(T)$ edges with exactly one endvertex that is unsaturated by M.
  - $\Rightarrow$ tree T' with $\mu(T') = \mu(T)$ and $\alpha(T') = \alpha(T) - (n - 2\mu(T))$.
  - M is now a perfect matching in T'.
  - Hence, contracting any edge in T' will result in a tree T'' with $\mu(T'') = \mu(T') - 1$ and $\alpha(T'') = \alpha(T')$.
  - But if we contract an edge $uv \in M$, the new vertex becomes unsaturated by $M' = M \setminus \{uv\}$.
  - Now we can contract an edge $(uv)w$ and obtain T''' with $\mu(T''') = \mu(T'')$ and $\alpha(T''') = \alpha(T'') - 1$.
  - We can show that this is optimal.

TREES

- Suppose that $d > n - 2\mu(T)$.
  - First contract the $n - 2\mu(T)$ edges with exactly one endvertex that is unsaturated by M.
  - $\Rightarrow$ tree T' with $\mu(T') = \mu(T)$ and $\alpha(T') = \alpha(T) - (n - 2\mu(T))$.
  - M is now a perfect matching in T'.
  - Hence, contracting any edge in T' will result in a tree T'' with $\mu(T'') = \mu(T') - 1$ and $\alpha(T'') = \alpha(T')$.
  - But if we contract an edge $uv \in M$, the new vertex becomes unsaturated by $M' = M \setminus \{uv\}$.
  - Now we can contract an edge $(uv)w$ and obtain T''' with $\mu(T''') = \mu(T'')$ and $\alpha(T''') = \alpha(T'') - 1$.
  - We can show that this is optimal.
  - $\Rightarrow$ if $k \geq 2(d + \mu(T)) - n$, the answer is YES, otherwise it's NO.

TREES - BIPARTITE GRAPHS

- We have

$$\mu(G) + \alpha(G) = n$$

for any bipartite graph G!

TREES - BIPARTITE GRAPHS

- We have

$$\mu(G) + \alpha(G) = n$$

for any bipartite graph G!

- But bipartite graphs are not closed under edge contraction!

TREES - BIPARTITE GRAPHS

- We have

$$\mu(G) + \alpha(G) = n$$

for any bipartite graph G!

- But bipartite graphs are not closed under edge contraction!
- Being closed under edge contraction plays an important role in our proof!

TREES - BIPARTITE GRAPHS

- We have

$$\mu(G) + \alpha(G) = n$$

for any bipartite graph G!

- But bipartite graphs are not closed under edge contraction!

- Being closed under edge contraction plays an important role in our proof!

Theorem [Costa et al., 2011], [Bazgan et al., 2011]

For $\pi \in \{\alpha, \omega, \chi\}$, $\texttt{DELETION BLOCKER}(\pi)$ is polynomial-time solvable in bipartite graphs.

INTERVAL GRAPHS

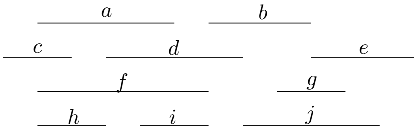
Interval graph

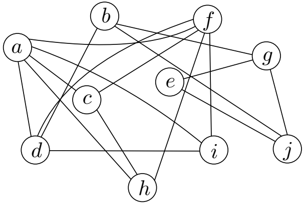
A graph G is an interval graph if one can associate with each vertex in G an interval on the real line such that two vertices are adjacent in G if and only the corresponding intervals intersect.

INTERVAL GRAPHS

Interval graph

A graph G is an interval graph if one can associate with each vertex in G an interval on the real line such that two vertices are adjacent in G if and only the corresponding intervals intersect.





PROPERTIES OF INTERVAL GRAPHS

- Interval graphs do not contain any induced cycle of length 4.

PROPERTIES OF INTERVAL GRAPHS

- Interval graphs do not contain any induced cycle of length 4.
- An interval graph G on n vertices contains at most n maximal cliques.

PROPERTIES OF INTERVAL GRAPHS

- Interval graphs do not contain any induced cycle of length 4.
- An interval graph G on n vertices contains at most n maximal cliques.
- All maximal cliques in an interval graph G can be found in polynomial time.

PROPERTIES OF INTERVAL GRAPHS

- Interval graphs do not contain any induced cycle of length 4.
- An interval graph G on n vertices contains at most n maximal cliques.
- All maximal cliques in an interval graph G can be found in polynomial time.
- Two technical Lemmas are needed as well.

INTERVAL GRAPHS

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

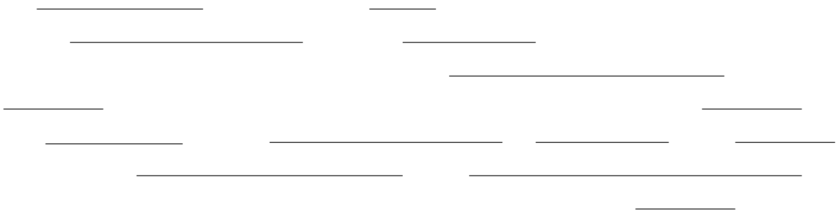
Let $\pi \in \{\omega, \chi\}$. Then $\texttt{CONTRACTION BLOCKER}(\pi)$ can be solved in polynomial time on interval graphs.

INTERVAL GRAPHS

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

Let $\pi \in \{\omega, \chi\}$. Then `CONTRACTION BLOCKER`(π) can be solved in polynomial time on interval graphs.

Idea of the proof:

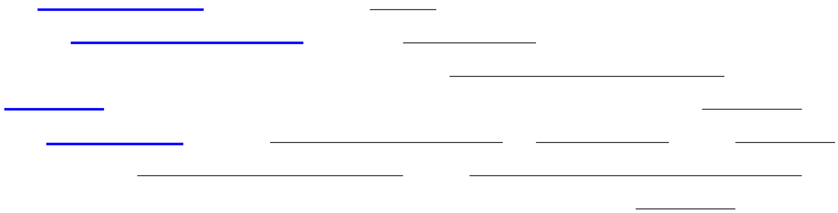


INTERVAL GRAPHS

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

Let $\pi \in \{\omega, \chi\}$. Then $\texttt{CONTRACTION BLOCKER}(\pi)$ can be solved in polynomial time on interval graphs.

Idea of the proof:

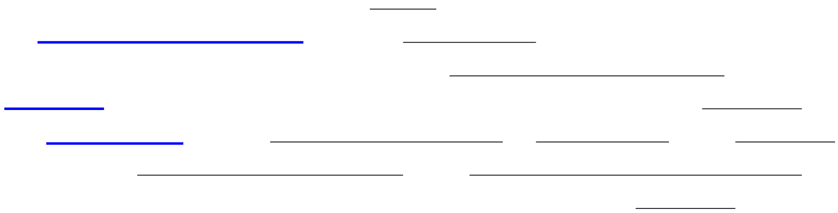


INTERVAL GRAPHS

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

Let $\pi \in \{\omega, \chi\}$. Then $\texttt{CONTRACTION BLOCKER}(\pi)$ can be solved in polynomial time on interval graphs.

Idea of the proof:

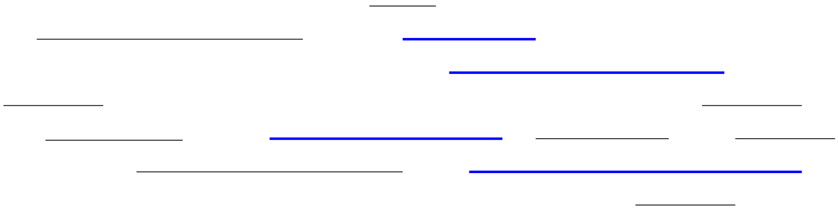


INTERVAL GRAPHS

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

Let $\pi \in \{\omega, \chi\}$. Then $\texttt{CONTRACTION BLOCKER}(\pi)$ can be solved in polynomial time on interval graphs.

Idea of the proof:

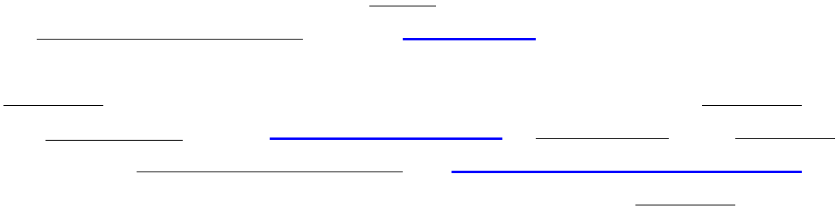


INTERVAL GRAPHS

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

Let $\pi \in \{\omega, \chi\}$. Then $\texttt{CONTRACTION BLOCKER}(\pi)$ can be solved in polynomial time on interval graphs.

Idea of the proof:

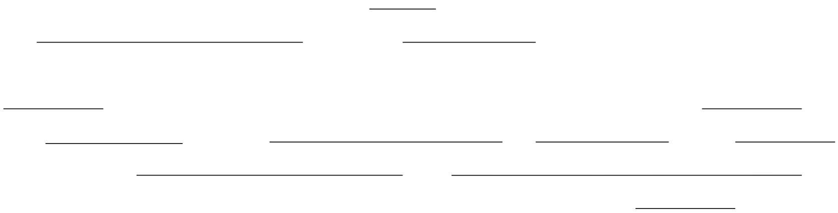


INTERVAL GRAPHS

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

Let $\pi \in \{\omega, \chi\}$. Then `CONTRACTION BLOCKER(`π`)` can be solved in polynomial time on interval graphs.

Idea of the proof:



INTERVAL GRAPHS

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

Let $\pi \in \{\omega, \chi\}$. Then `CONTRACTION BLOCKER`(π) can be solved in polynomial time on interval graphs.

Using similar arguments, we can also show the following:

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

Let $\pi \in \{\omega, \chi\}$. Then `DELETION BLOCKER`(π) can be solved in polynomial time on interval graphs.

CHORDAL GRAPHS

The previous results cannot be generalized to chordal graphs:

Theorem [Paulusma, Picouleau, R., 2016]

Let $\pi \in \{\omega, \chi\}$. Then 1-CONTRACTION BLOCKER(π) and 1-DELETION BLOCKER(π) are NP-complete for chordal graphs.

CHORDAL GRAPHS

The previous results cannot be generalized to chordal graphs:

Theorem [Paulusma, Picouleau, R., 2016]

Let $\pi \in \{\omega, \chi\}$. Then 1-CONTRACTION BLOCKER(π) and 1-DELETION BLOCKER(π) are NP-complete for chordal graphs.

- For $\pi = \alpha$, the complexity of both problems, CONTRACTION BLOCKER(α) and DELETION BLOCKER(α), is unknown in interval graphs.
- For $\pi = \alpha$, the complexity of both problems, 1-CONTRACTION BLOCKER(α) and 1-DELETION BLOCKER(α), is unknown in chordal graphs.

COGRAPHS

Cograph

A graph G is a cograph if it does not contain any path on 4 vertices as an induced subgraph.

COGRAPHS

Cograph

A graph G is a cograph if it does not contain any path on 4 vertices as an induced subgraph.

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

For $\pi \in \{\alpha, \omega, \chi\}$, both CONTRACTION BLOCKER(π) and DELETION BLOCKER(π) can be solved in polynomial time for cographs.

SPLIT GRAPHS

Split graph

A graph G is a split graph if its vertex set can be partitioned into a clique and a stable set.

SPLIT GRAPHS

Split graph

A graph G is a split graph if its vertex set can be partitioned into a clique and a stable set.

Theorem [Costa et al., 2011]

For $\pi \in \{\alpha, \omega, \chi\}$, DELETION BLOCKER($\pi$) is NP-complete and d-DELETION BLOCKER(π) is polynomial time solvable for split graphs.

SPLIT GRAPHS

Split graph

A graph G is a split graph if its vertex set can be partitioned into a clique and a stable set.

Theorem [Costa et al., 2011]

For $\pi \in \{\alpha, \omega, \chi\}$, $\texttt{DELETION BLOCKER}(\pi)$ is NP-complete and d-$\texttt{DELETION BLOCKER}(\pi)$ is polynomial time solvable for split graphs.

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

For $\pi \in \{\alpha, \omega, \chi\}$, $\texttt{CONTRACTION BLOCKER}(\pi)$ is NP-complete and d-$\texttt{CONTRACTION BLOCKER}(\pi)$ is polynomial time solvable for split graphs.

SUBCLASSES OF PERECT GRAPHS

	CONTRACTION BLOCKER(π)		DELETION BLOCKER(π)	
Class	$\pi = \alpha$	$\pi = \omega = \chi$	$\pi = \alpha$	$\pi = \omega = \chi$
Trees	P	P	P*	P
Bipartite	NP-h	P	P*	P
Cobipartite	$d = 1$: NP-c	NP-c d fixed: P	P	P*
Cograph	P	P	P	P
Split	NP-c d fixed: P	NP-c d fixed: P	NP-c* d fixed: P*	NP-c* d fixed: P*
Interval		P		P
Chordal	NP-c	$d = 1$: NP-c	NP-c	$d = 1$: NP-c
Perfect	$d = 1$: NP-h	$d = 1$: NP-h	NP-c	$d = 1$: NP-c

P_ℓ-FREE GRAPHS

- Cographs are equivalent to P_4-free graphs, i.e. graphs not containing an induced path on 4 vertices.

P_ℓ-FREE GRAPHS

- Cographs are equivalent to P_4-free graphs, i.e. graphs not containing an induced path on 4 vertices.
- Split graphs do not contain any induced path on 5 vertices, i.e. they are P_5-free.

P_ℓ-FREE GRAPHS

- Cographs are equivalent to P_4-free graphs, i.e. graphs not containing an induced path on 4 vertices.
- Split graphs do not contain any induced path on 5 vertices, i.e. they are P_5-free.
- So we obtain the following dichotomy from the previous results for P_ℓ-free graphs, i.e. not containing any induced path on ℓ vertices.

P_ℓ-FREE GRAPHS

- Cographs are equivalent to P_4-free graphs, i.e. graphs not containing an induced path on 4 vertices.
- Split graphs do not contain any induced path on 5 vertices, i.e. they are P_5-free.
- So we obtain the following dichotomy from the previous results for P_ℓ-free graphs, i.e. not containing any induced path on ℓ vertices.

Theorem [Dinner, Paulusma, Picouleau, R., 2015]

Let $\pi \in \{\alpha, \omega, \chi\}$. Then CONTRACTION BLOCKER(π) and DELETION BLOCKER(π) can be solved in polynomial time in P_ℓ-free graphs if $\ell \leq 4$ and NP-hard if $\ell \geq 5$.

1. INTRODUCTION

2. PREVIOUS WORK

3. VERTEX DELETION, EDGE CONTRACTION, $\pi \in \{\alpha, \omega, \chi\}$
   - SUBCLASSES OF PERFECT GRAPHS
   - *H*-FREE GRAPHS

4. CONCLUSION

DEFINITIONS

H-free graph

Let G, H be two graph. Then G is said to be H-free, if it does not contain H as an induced subgraph.

DEFINITIONS

H-free graph

Let G, H be two graph. Then G is said to be H-free, if it does not contain H as an induced subgraph.

- If H is an induced subgraph of G, we write $H \subseteq_i G$.

DEFINITIONS

H-free graph

Let G, H be two graph. Then G is said to be H-free, if it does not contain H as an induced subgraph.

- If H is an induced subgraph of G, we write $H \subseteq_i G$.
- Let G_1, G_2 be two vertex-disjoint graphs. The union $G_1 \oplus G_2$ creates the disjoint union of G_1 and G_2.

DEFINITIONS

H-free graph

Let G, H be two graph. Then G is said to be H-free, if it does not contain H as an induced subgraph.

- If H is an induced subgraph of G, we write $H \subseteq_i G$.
- Let G_1, G_2 be two vertex-disjoint graphs. The union $G_1 \oplus G_2$ creates the disjoint union of G_1 and G_2.

clique-proof

We say that a graph class $\mathcal{G}$ is clique-proof if

$$G \in \mathcal{G} \implies G \oplus G \in \mathcal{G}, \; G \oplus K_s \in \mathcal{G}$$

SOME USEFUL RESULTS

Theorem [Paulusma, Picouleau, R., 2016]

If `CLIQUE` is NP-complete for a clique-proof graph class $\mathcal{G}$, then `CONTRACTION BLOCKER`(ω) is co-NP-hard for $\mathcal{G}$, even if $d = k = 1$.

SOME USEFUL RESULTS

Theorem [Paulusma, Picouleau, R., 2016]

If CLIQUE is NP-complete for a clique-proof graph class $\mathcal{G}$, then CONTRACTION BLOCKER(ω) is co-NP-hard for $\mathcal{G}$, even if $d = k = 1$.

Theorem [Král et al., 2011]

Let H be a graph. If $H \subseteq_i P_4$ or $H \subseteq_i P_3 \oplus K_1$, then the COLORING problem is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

SOME USEFUL RESULTS

Theorem [Paulusma, Picouleau, R., 2016]

If CLIQUE is NP-complete for a clique-proof graph class $\mathcal{G}$, then CONTRACTION BLOCKER(ω) is co-NP-hard for $\mathcal{G}$, even if $d = k = 1$.

Theorem [Král et al., 2011]

Let H be a graph. If $H \subseteq_i P_4$ or $H \subseteq_i P_3 \oplus K_1$, then the COLORING problem is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Theorem [Paulusma, Picouleau, R., 2017]

1-DELETION BLOCKER(α), with $k = 1$, is NP-hard even for graphs of girth at least g, for every fixed $g \geq 3$.

H-FREE GRAPHS, $\pi \in \{\omega, \alpha\}$

Theorem [Dinner, Paulusma, Picouleau, R., 2015-2017]

Let H be a graph. Then the following holds:

- If $H \subseteq_i P_4$, then DELETION BLOCKER(α) and CONTRACTION BLOCKER(α) are polynomial-time solvable for H-free graphs; otherwise both are NP-hard or co-NP-hard for H-free graphs.
- If $H \subseteq_i P_4$, then DELETION BLOCKER(ω) is polynomial-time solvable for H-free graphs; otherwise it is NP-hard or co-NP-hard for H-free graphs.
- Let $H \neq K_3 \oplus K_1$. If $H \subseteq_i P_4$ or $H \subseteq_i$ paw, then CONTRACTION BLOCKER(ω) is polynomial-time solvable for H-free graphs; otherwise it is NP-hard or co-NP-hard for H-free graphs.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then CONTRACTION BLOCKER(α) is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then $\texttt{CONTRACTION BLOCKER}(\alpha)$ is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then CONTRACTION BLOCKER(α) is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.
- Otherwise, suppose that H contains a cycle C_r.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then CONTRACTION BLOCKER(α) is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.
- Otherwise, suppose that H contains a cycle C_r.
  - If r is odd, use the result on bipartite graphs.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then $\texttt{CONTRACTION BLOCKER}(\alpha)$ is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.
- Otherwise, suppose that H contains a cycle C_r.
  - If r is odd, use the result on bipartite graphs.
  - If r is even, H either contains C_4 or $2P_2$; use previous result on split graphs.

H-FREE GRAPHS

Idea of the proof (continued):

- So we may assume that H contains no cycle, hence is a forest.

H-FREE GRAPHS

Idea of the proof (continued):

- So we may assume that H contains no cycle, hence is a forest.
  - If H contains $3P_1$, use previous result on cobipartite graphs.

H-FREE GRAPHS

Idea of the proof (continued):

- So we may assume that H contains no cycle, hence is a forest.
  - If H contains $3P_1$, use previous result on cobipartite graphs.
  - So suppose that H is $3P_1$-free; hence it must contain $2P_2$ otherwise $H \subseteq_i P_4$; use previous result on split graphs.

□

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then $\texttt{DELETION BLOCKER}(\alpha)$ is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then DELETION BLOCKER(α) is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then DELETION BLOCKER(α) is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.
- Otherwise, suppose that H contains a cycle C_r.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then DELETION BLOCKER(α) is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.
- Otherwise, suppose that H contains a cycle C_r.
  - If $r = 3$, use previous result on graphs of girth at least $g = 4$.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then DELETION BLOCKER(α) is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.
- Otherwise, suppose that H contains a cycle C_r.
  - If $r = 3$, use previous result on graphs of girth at least $g = 4$.
  - If $r \geq 4$, use previous result on split graphs.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then DELETION BLOCKER(α) is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.
- Otherwise, suppose that H contains a cycle C_r.
  - If $r = 3$, use previous result on graphs of girth at least $g = 4$.
  - If $r \geq 4$, use previous result on split graphs.
- So we may assume now that H is a forest.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then $\texttt{DELETION BLOCKER}(\alpha)$ is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.
- Otherwise, suppose that H contains a cycle C_r.
  - If $r = 3$, use previous result on graphs of girth at least $g = 4$.
  - If $r \geq 4$, use previous result on split graphs.
- So we may assume now that H is a forest.
  - If $2P_2 \subseteq_i H$, use previous result on split graphs.

H-FREE GRAPHS

Theorem [Paulusma, Picouleau, R., 2017]

Let H be a graph. If $H \subseteq_i P_4$, then DELETION BLOCKER(α) is polynomial-time solvable for H-free graphs, otherwise it is NP-hard for H-free graphs.

Idea of the proof:

- If $H \subseteq_i P_4$, use previous result on cographs.
- Otherwise, suppose that H contains a cycle C_r.
  - If $r = 3$, use previous result on graphs of girth at least $g = 4$.
  - If $r \geq 4$, use previous result on split graphs.
- So we may assume now that H is a forest.
  - If $2P_2 \subseteq_i H$, use previous result on split graphs.
  - If $3P_1 \subseteq_i H$, we use the fact the DELETION BLOCKER(ω) is NP-hard for triangle-free graphs. ($\Rightarrow$ VERTEX COVER)

H-FREE GRAPHS, $\pi = \chi$

Theorem [Dinner, Paulusma, Picouleau, R., 2015-2017]

Let H be a graph. Then the following holds:

- If $H \subseteq_i P_4$, then CONTRACTION BLOCKER(χ) is polynomial-time solvable for H-free graphs; otherwise it is NP-hard for H-free graphs.
- If $H \subseteq_i P_4$ or $H \subseteq_i P_3 \oplus K_1$, then DELETION BLOCKER($\chi$) is polynomial-time solvable for H-free graphs; otherwise it is NP-hard or co-NP-hard for H-free graphs.

1. INTRODUCTION

2. PREVIOUS WORK

3. VERTEX DELETION, EDGE CONTRACTION, $\pi \in \{\alpha, \omega, \chi\}$
   - SUBCLASSES OF PERFECT GRAPHS
   - *H*-FREE GRAPHS

4. CONCLUSION

CONCLUSION

- We presented blocker problems and their relations with other graph problems.

CONCLUSION

- We presented blocker problems and their relations with other graph problems.
- In particular we focused on the following two problems and presented results for subclasses of perfect graphs and for H-free graphs.

CONTRACTION BLOCKER(π)

Instance: a graph G and two integers $d, k \geq 0$

Question: can G be k-contracted into G' with $\pi(G') \leq \pi(G) - d$?

DELETION BLOCKER(π)

Instance: a graph G and two integers $d, k \geq 0$

Question: can G be k-vertex-deleted in G' with $\pi(G') \leq \pi(G) - d$?

OPEN PROBLEMS

What is the complexity of

- (1-)CONTRACTION BLOCKER(α) and (1-)DELETION BLOCKER(α) for interval graphs?
- 1-CONTRACTION BLOCKER(α) and 1-DELETION BLOCKER(α) for chordal graphs?
- CONTRACTION BLOCKER(ω) for $(K_3 \oplus K_1)$-free graphs?
- 1-CONTRACTION BLOCKER(π) and 1-DELETION BLOCKER(π) for H-free graphs with $k = 1$ and $\pi \in \{\omega, \alpha\}$?

Thank you for your attention!