Domination-like problems parameterized by tree-width

Mathieu Chapelle

LIGM, Université Paris-Est Marne-la-Vallée

JFRO 2012
LIP6, Université Paris 6

In brief

- Very few problems are known to be **W**-hard (*i.e.* not **FPT**) when parameterized by tree-width;

In brief

- Very few problems are known to be **W**-hard (*i.e.* not **FPT**) when parameterized by tree-width;
- All the studied cases of GENERALIZED DOMINATION are **FPT** when parameterized by tree-width;

In brief

- Very few problems are known to be **W**-hard (*i.e.* not **FPT**) when parameterized by tree-width;
- All the studied cases of Generalized domination are **FPT** when parameterized by tree-width;

1. We extend known results of "**FPT**ness" to more cases;

In brief

- Very few problems are known to be **W**-hard (*i.e.* not **FPT**) when parameterized by tree-width;
- All the studied cases of Generalized domination are **FPT** when parameterized by tree-width;

1. We extend known results of "**FPT**ness" to more cases;
2. We prove that there exists (many) cases for which Generalized domination become **W**[1]-hard when parameterized by tree-width.

Parameterized complexity

$$\mathbf{FPT} \subset \mathbf{W}[1] \subset \mathbf{W}[2] \subset \ldots \subset \mathbf{XP}$$

Parameterized complexity

$$\textbf{FPT} \subset \textbf{W}[1] \subset \textbf{W}[2] \subset \ldots \subset \textbf{XP}$$

Definition

A problem $\mathcal{P}$ is in **FPT** parameterized by k if it can be solved in time $\mathcal{O}\big(f(k) \cdot \text{poly}(n)\big)$.

Example

k-VERTEX COVER can be solved in time $\mathcal{O}\big(1.2738^k \cdot k \cdot n\big)$.

[Chen, Kanj, Xia, 2010]

Parameterized complexity

$$\mathbf{FPT} \subset \mathbf{W}[1] \subset \mathbf{W}[2] \subset \ldots \subset \mathbf{XP}$$

Definition

A problem $\mathcal{P}$ is in **XP** parameterized by k if it can be solved in time $\mathcal{O}\big(poly(n)^{f(k)}\big)$.

Example

k-Coloration is not in **XP**.

Parameterized complexity

$$\mathbf{FPT} \subset \mathbf{W}[1] \subset \mathbf{W}[2] \subset \ldots \subset \mathbf{XP}$$

Definition

A problem $\mathcal{P}$ is W[t]-hard parameterized by k if there exists an *fpt*-reduction from any known W[1]-hard problem $\mathcal{Q}$ to $\mathcal{P}$, that is:

$$\mathcal{Q} \preceq_{\text{fpt}} \mathcal{P}$$

Examples

k-Independent set is W[1]-hard.
k-Dominating set is W[2]-hard.

Some definitions

FPT cases

W[1]-hardness

Conclusion

Generalized domination

Definition

$D \subseteq V$ is a dominating set if, for all $v \in V$:

- $v \in D$; or
- $\exists u \in V : u \in D \cap N(v)$.

Generalized domination

Definition

$D \subseteq V$ is a dominating set if, for all $v \in V$:

- $v \in D \Rightarrow |D \cap N(v)| \geq 0$;
- $v \notin D \Rightarrow |D \cap N(v)| \geq 1$.

Generalized domination

Definition

$D \subseteq V$ is a dominating set if, for all $v \in V$:

- $v \in D \Rightarrow |D \cap N(v)| \in \mathbb{N}$;
- $v \notin D \Rightarrow |D \cap N(v)| \in \mathbb{N}^*$.

Generalized domination

Definition

$D \subseteq V$ is a $[\sigma, \varrho]$-dominating set if, for all $v \in V$:

- $v \in D \Rightarrow |D \cap N(v)| \in \sigma$; $\qquad \sigma \subseteq \mathbb{N}$
- $v \notin D \Rightarrow |D \cap N(v)| \in \varrho$. $\qquad \varrho \subseteq \mathbb{N}$

σ and ϱ fix some constraints on the neighborhood of each vertex:

- σ fixes constraints on the neighborhood of vertices in D;
- ϱ fixes constraints on the neighborhood of vertices not in D.

Generalized domination

Definition

$D \subseteq V$ is a $[\sigma, \varrho]$-dominating set if, for all $v \in V$:

- $v \in D \Rightarrow |D \cap N(v)| \in \sigma$; $\quad \sigma \subseteq \mathbb{N}$
- $v \notin D \Rightarrow |D \cap N(v)| \in \varrho$. $\quad \varrho \subseteq \mathbb{N}$

σ and ϱ fix some constraints on the neighborhood of each vertex:

- σ fixes constraints on the neighborhood of vertices in D;
- ϱ fixes constraints on the neighborhood of vertices not in D.

Remark

A graph G does not always admit a $[\sigma, \varrho]$-dominating set.

Known domination-like problems

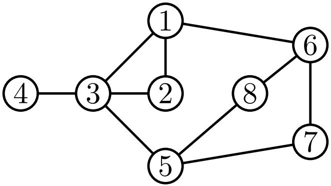
Problem	σ	ϱ
Dominating set	$\mathbb{N}$	$\mathbb{N}^*$
Independent set	$\{0\}$	$\mathbb{N}$
Perfect code	$\{0\}$	$\{1\}$
Independent dominating set	$\{0\}$	$\mathbb{N}^*$
Total dominating set	$\mathbb{N}^*$	$\mathbb{N}^*$
Induced matching	$\{1\}$	$\mathbb{N}$

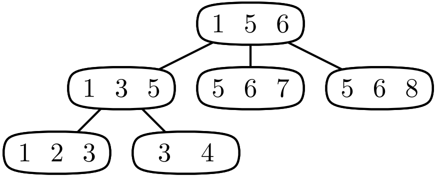
...

Definition

A tree decomposition $(T, \{X_i \subseteq V\})$ of a graph $G = (V, E)$ is such that:

- $\forall v \in V$, $\exists i : v \in X_i$;
- $\forall uv \in E$, $\exists i : u, v \in X_i$;
- $\forall v \in V$, the *bags* containing v induce a subtree of T.

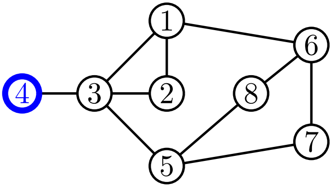


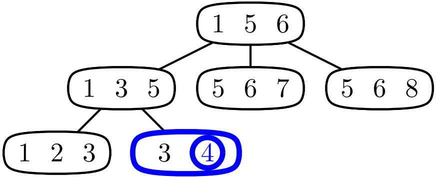


Definition

A tree decomposition $\left(T, \{X_i \subseteq V\}\right)$ of a graph $G = (V, E)$ is such that:

- $\forall v \in V, \exists i : v \in X_i$;
- $\forall uv \in E, \exists i : u, v \in X_i$;
- $\forall v \in V$, the *bags* containing v induce a subtree of T.

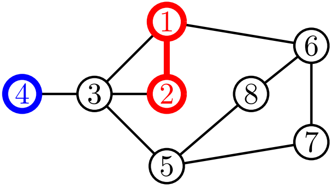


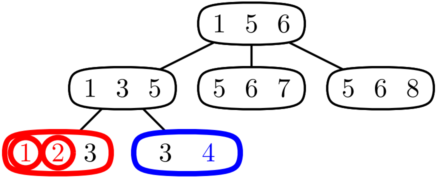


Definition

A tree decomposition $\big(T, \{X_i \subseteq V\}\big)$ of a graph $G = (V, E)$ is such that:

- $\forall v \in V$, $\exists i : v \in X_i$;
- $\forall uv \in E$, $\exists i : u, v \in X_i$;
- $\forall v \in V$, the *bags* containing v induce a subtree of T.

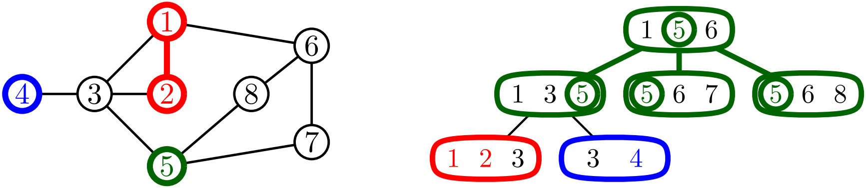




Definition

A tree decomposition $(T, \{X_i \subseteq V\})$ of a graph $G = (V, E)$ is such that:

- $\forall v \in V, \exists i : v \in X_i$;
- $\forall uv \in E, \exists i : u, v \in X_i$;
- $\forall v \in V$, the *bags* containing v induce a subtree of T.



Tree-width

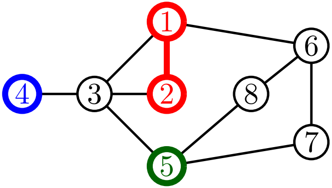
Definition

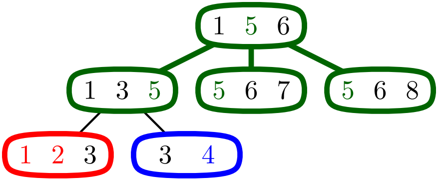
A tree decomposition $(T, \{X_i \subseteq V\})$ of a graph $G = (V, E)$ is such that:

- $\forall v \in V$, $\exists i : v \in X_i$;
- $\forall uv \in E$, $\exists i : u, v \in X_i$;
- $\forall v \in V$, the *bags* containing v induce a subtree of T.

Width of a decomposition $= \max |X_i| - 1$.

Tree-width of G, $\text{tw}(G)$ = smallest width over all decompositions of G.





Some definitions

FPT cases

W[1]-hardness

Conclusion

Known results

Theorem (van Rooij, Bodlaender, Rossmanith, 2009)

$[\sigma, \varrho]$-Dominating set can be solved in time $\mathcal{O}^*(s^{\text{tw}})$, if σ and ϱ are both finite or cofinite, where s is the minimum number of states needed to represent σ and ϱ.

Known results

Theorem (van Rooij, Bodlaender, Rossmanith, 2009)
$[\sigma, \varrho]$-DOMINATING SET can be solved in time $\mathcal{O}^*(s^{\text{tw}})$, if σ and ϱ are both finite or cofinite, where s is the minimum number of states needed to represent σ and ϱ.

Corollary
INDEPENDENT SET can be solved in time $\mathcal{O}^*(2^{\text{tw}})$.

[Niedermeier, 2006]

DOMINATING SET can be solved in time $\mathcal{O}^*(3^{\text{tw}})$.

[van Rooij, Bodlaender, Rossmanith, 2009]

Known results

Theorem (van Rooij, Bodlaender, Rossmanith, 2009)
$[\sigma, \varrho]$-DOMINATING SET can be solved in time $\mathcal{O}^*(s^{\mathrm{tw}})$, if σ and ϱ are both finite or cofinite, where s is the minimum number of states needed to represent σ and ϱ.

Corollary
INDEPENDENT SET can be solved in time $\mathcal{O}^*(2^{\mathrm{tw}})$.

[Niedermeier, 2006]

DOMINATING SET can be solved in time $\mathcal{O}^*(3^{\mathrm{tw}})$.

[van Rooij, Bodlaender, Rossmanith, 2009]

Under **SETH** hypothesis, this time complexity is optimal.

[Lokshtanov, Marx, Saurabh, 2010]

Extension of **FPT** cases

Using the famous Courcelle's theorem: [Courcelle, 1997]

If σ and ϱ are finite or cofinite, then $[\sigma, \varrho]$-DOMINATING SET is expressible in **$MSOL_2$**.
$\rightarrow$ **FPT** when parameterized by tree-width.

Extension of **FPT** cases

Using an extension of the famous Courcelle's theorem:

[Courcelle, Makowsky, Rotics, 2001]

If σ and ϱ are ultimately periodics, then $[\sigma, \varrho]$-DOMINATING SET is expressible in **CMSOL**.

$\rightarrow$ **FPT** when parameterized by tree-width.

Extension of FPT cases

Using an extension of the famous Courcelle's theorem:

[Courcelle, Makowsky, Rotics, 2001]

If σ and ϱ are ultimately periodics, then $[\sigma, \varrho]$-DOMINATING SET is expressible in **CMSOL**.
$\rightarrow$ **FPT** when parameterized by tree-width.

$$\begin{array}{rcl} \exists S, \bar{S} \;\; \forall v \in V : & & (v \in S \wedge v \notin \bar{S}) \vee (v \notin S \wedge v \in \bar{S}) \\ & \wedge & v \in S \Rightarrow |N(v) \cap S| \in \sigma \\ & \wedge & v \in \bar{S} \Rightarrow |N(v) \cap S| \in \varrho \end{array}$$

$$\begin{array}{rcl} |N(v) \cap S| \in \sigma & \equiv & \bigvee_{i \in \{1,\dots,k_\sigma\}} \exists Y_S \left(\mathrm{Card}_{t_i^\sigma}(Y_S) \wedge \bigvee_{p \in \sigma, p \leq p_\sigma} \exists u_1, \dots, u_p \;\; \zeta \right) \\ \text{avec} \;\; \zeta & \equiv & \left[\left(u_i \in (N(v) \cap S) \wedge u_i \notin Y_S \right) \wedge \forall u \;\; (u \neq u_i) \right] \Rightarrow \left(u \in Y_S \Leftrightarrow u \in (N(v) \cap S) \right) \end{array}$$

Extension of **FPT** cases

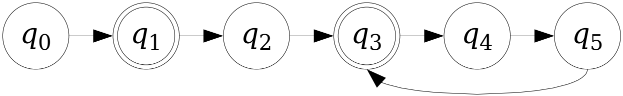
Theorem

$[\sigma, \varrho]$-DOMINATING SET can be solved in time $\mathcal{O}^*(s^{\text{tw}})$, if σ and ϱ are both ultimately periodics, where s is a *small* function on the minimum number of states needed to represent σ and ϱ by two automata.

$$s = |\sigma_0| + |\varrho_0| + \text{maxperiod}(\sigma)^2 + \text{maxperiod}(\varrho)^2$$

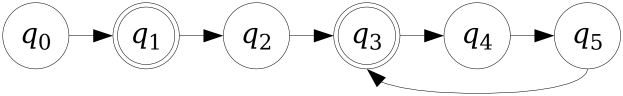
Algorithm idea

We use two finite deterministic unary-language automata to enumerate σ and ϱ.



Algorithm idea

We use two finite deterministic unary-language automata to enumerate σ and ϱ.



The state associated to a given vertex $v \in V$ encode:

- whether v is in D (state in σ) or not in D (state in ϱ);
- the number of neighbors it has in D.

Algorithm idea

- Represent σ and ϱ by two finite unary-language automata;

Algorithm idea

Algorithm idea

- Represent σ and ϱ by two finite unary-language automata;
- Apply a classical dynamic programming scheme on a (nice) tree decomposition of the input graph;

Algorithm idea

- Represent σ and ϱ by two finite unary-language automata;
- Apply a classical dynamic programming scheme on a (nice) tree decomposition of the input graph;
- Encode the number of selected neighbors of each vertex using the corresponding state in one of the two automata;

Algorithm idea

- Represent σ and ϱ by two finite unary-language automata;
- Apply a classical dynamic programming scheme on a (nice) tree decomposition of the input graph;
- Encode the number of selected neighbors of each vertex using the corresponding state in one of the two automata;
- Use fast subset convolution to fasten the `join` operation.

Algorithm idea

- Represent σ and ϱ by two finite unary-language automata;
- Apply a classical dynamic programming scheme on a (nice) tree decomposition of the input graph;
- Encode the number of selected neighbors of each vertex using the corresponding state in one of the two automata;
- Use fast subset convolution to fasten the `join` operation.

Theorem

$[\sigma, \varrho]$-Dominating set can be efficiently solved in **FPT** time if σ and ϱ are both ultimately periodics.

Some definitions

FPT cases

$W[1]$-hardness

Conclusion

Motivation

Question

Is $[\sigma, \varrho]$-DOMINATING SET always **FPT** when parameterized by tree-width?

Remark

Very few *parameterized* graph problems are known not to be **FPT** when parameterized by tree-width.

Motivation

Question

Is $[\sigma, \varrho]$-Dominating set always **FPT** when parameterized by tree-width?

Remark

Very few *parameterized* graph problems are known not to be **FPT** when parameterized by tree-width.

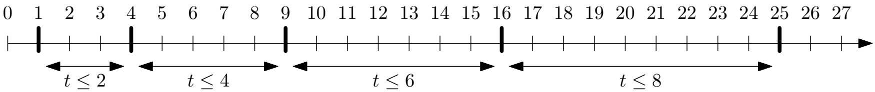
Lemma

For any polytime decidable sets σ and ϱ, $[\sigma, \varrho]$-Dominating set is in **XP** when parameterized by tree-width.

Some **W**[1]-hard cases

Theorem

If σ exclude arbitrary long intervals and ϱ is cofinite, then $[\sigma, \varrho]$-Dominating set is W[1]-hard when parameterized by tw.



Technical condition on σ:

We require that an excluded interval of length t can be found at distance *poly*(t).

Some W[1]-hard cases

Theorem

If σ exclude arbitrary long intervals and ϱ is cofinite, then $[\sigma, \varrho]$-Dominating set is W[1]-hard when parameterized by tw.

Given σ and ϱ, we will reduce k-Capacitated dominating set to $[\sigma, \varrho]$-Dominating set.

k-Capacitated dominating set is W[1]-hard when parameterized by the tree-width of the input graph + the size k of the expected solution.

[Dom, Lokshtanov, Saurabh, Villanger, 2008]

Two-steps reduction

Step 1

k-Capacitated dominating set

$\preceq_{fpt}$

$[\sigma, \varrho]$-Dominating set with preselected vertices

Step 2

$[\sigma, \varrho]$-Dominating set with preselected vertices

$\preceq_{fpt}$

$[\sigma, \varrho]$-Dominating set

Two-steps reduction

Step 1

k-Capacitated dominating set

$\preceq_{fpt}$

$[\sigma, \varrho]$-Dominating set with preselected vertices

Step 2

$[\sigma, \varrho]$-Dominating set with preselected vertices

$\preceq_{fpt}$

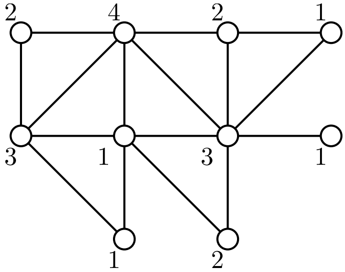
$[\sigma, \varrho]$-Dominating set

Capacitated domination

Definition

Let $G = (V, E)$, $\text{cap} : V \to \mathbb{N}$.
(C, dom) is a capacitated dominating set of G, with $C \subseteq V$ and dom(v) a function which associates to each vertex $v \in C$ a subset of its vertices, if:

- $\forall v \in C,\ |\text{dom}(v)| \leq \text{cap}(v)$;
- $\forall u \notin C,\ \exists v \in C : u \in \text{dom}(v)$.

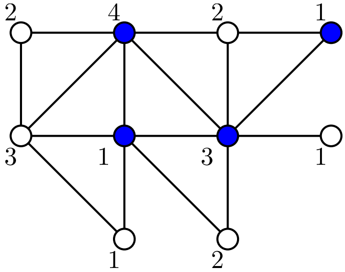


Capacitated domination

Definition

Let $G = (V, E)$, $\text{cap} : V \to \mathbb{N}$.
(C, dom) is a capacitated dominating set of G, with $C \subseteq V$ and dom(v) a function which associates to each vertex $v \in C$ a subset of its vertices, if:

- $\forall v \in C,\ |\text{dom}(v)| \leq \text{cap}(v)$;
- $\forall u \notin C,\ \exists v \in C : u \in \text{dom}(v)$.

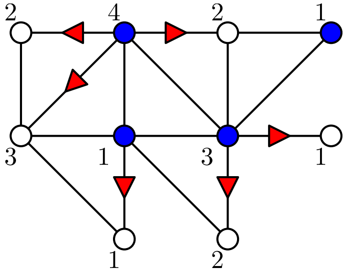


Capacitated domination

Definition

Let $G = (V, E)$, $\text{cap} : V \to \mathbb{N}$.
(C, dom) is a capacitated dominating set of G, with $C \subseteq V$ and dom(v) a function which associates to each vertex $v \in C$ a subset of its vertices, if:

- $\forall v \in C,\ |\text{dom}(v)| \leq \text{cap}(v)$;
- $\forall u \notin C,\ \exists v \in C : u \in \text{dom}(v)$.



Capacitated domination

Definition

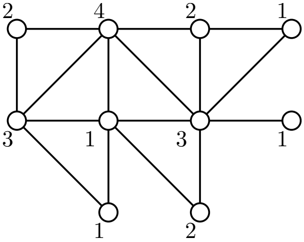
Let $G = (V, E)$, $\mathsf{cap} : V \to \mathbb{N}$.
(C, dom) is a capacitated dominating set of G, with $C \subseteq V$ and dom(v) a function which associates to each vertex $v \in C$ a subset of its vertices, if:

- $\forall v \in C,\ |\mathsf{dom}(v)| \leq \mathsf{cap}(v)$;
- $\forall u \notin C,\ \exists v \in C : u \in \mathsf{dom}(v)$.

k-CAPACITATED DOMINATING SET
Input: $G = (V, E)$ of tree-width tw, and $\mathsf{cap} : V \to \mathbb{N}$.
Parameter: $k + \mathsf{tw}$.
Question: Decide whether G admits a capacitated dominating set (C, dom) such that $|C| \leq k$.

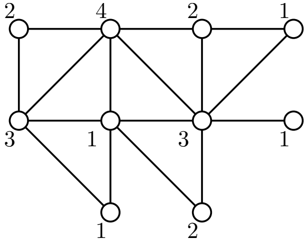
Reduction (step 1)

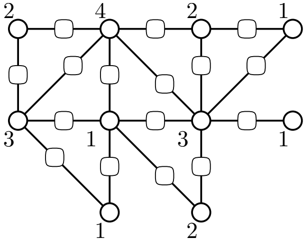
Let G be an instance of k-Capacitated dominating set.



Reduction (step 1)

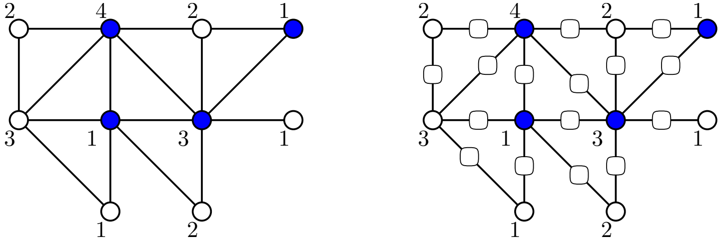
Let G be an instance of k-Capacitated dominating set.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:





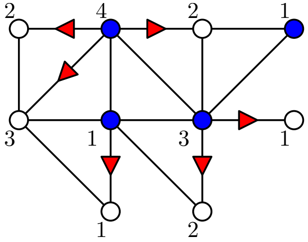
Reduction (step 1)

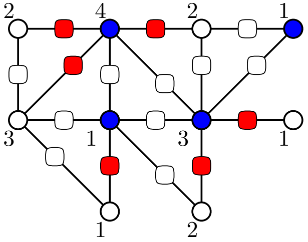
Let G be an instance of k-Capacitated dominating set.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:



Reduction (step 1)

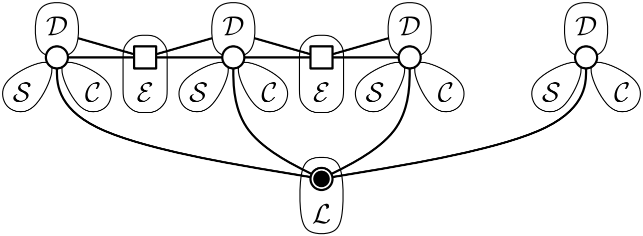
Let G be an instance of k-Capacitated dominating set.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:





Reduction (step 1)

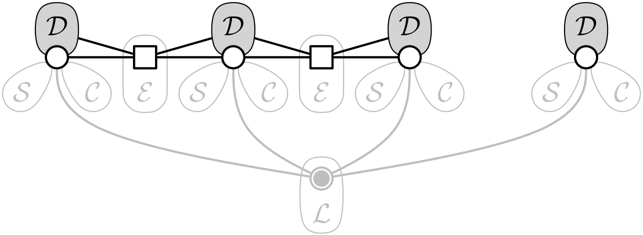
Let G be an instance of k-Capacitated dominating set.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:



Reduction (step 1)

Let G be an instance of k-Capacitated dominating set.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:

domination ($\mathcal{D}$) : force G to admit a (classical) dominating set;

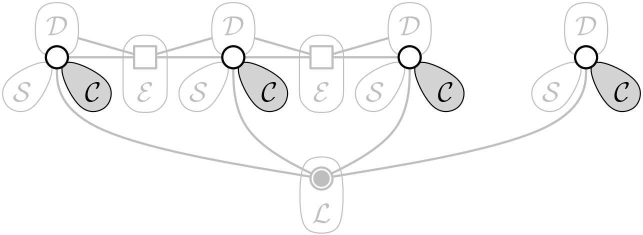


Reduction (step 1)

Let G be an instance of k-CAPACITATED DOMINATING SET.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:

domination ($\mathcal{D}$) : force G to admit a (classical) dominating set;

capacity ($\mathcal{C}$) : encode the capacity function cap, and allows a selected vertex to be satisfied;



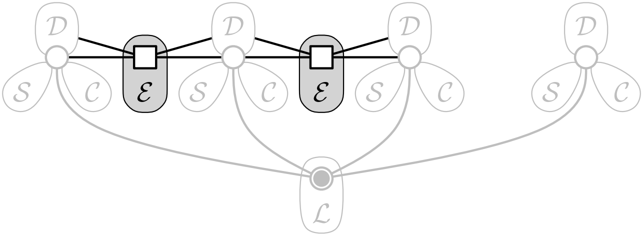
Reduction (step 1)

Let G be an instance of k-Capacitated dominating set.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:

domination ($\mathcal{D}$) : force G to admit a (classical) dominating set;

capacity ($\mathcal{C}$) : encode the capacity function cap, and allows a selected vertex to be satisfied;

edge-selection ($\mathcal{E}$) : encode the domination function dom;



Reduction (step 1)

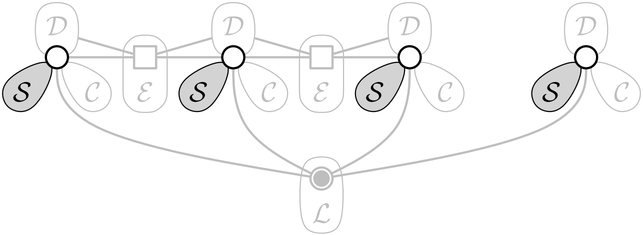
Let G be an instance of k-Capacitated dominating set.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:

domination ($\mathcal{D}$) : force G to admit a (classical) dominating set;

capacity ($\mathcal{C}$) : encode the capacity function cap, and allows a selected vertex to be satisfied;

edge-selection ($\mathcal{E}$) : encode the domination function dom;

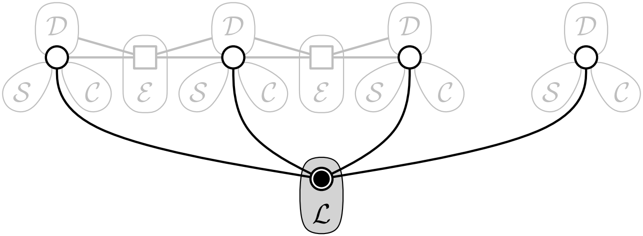
satisfiability ($\mathcal{S}$) : allow a non-selected vertex to be satisfied;



Reduction (step 1)

Let G be an instance of k-Capacitated dominating set.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:

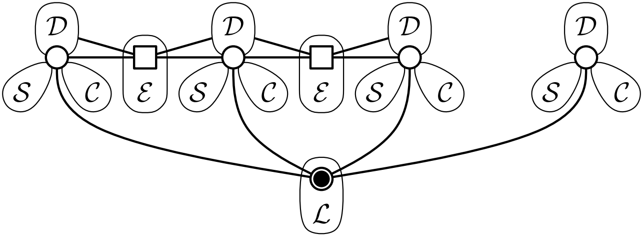
- *domination* ($\mathcal{D}$) : force G to admit a (classical) dominating set;
- *capacity* ($\mathcal{C}$) : encode the capacity function cap, and allows a selected vertex to be satisfied;
- *edge-selection* ($\mathcal{E}$) : encode the domination function dom;
- *satisfiability* ($\mathcal{S}$) : allow a non-selected vertex to be satisfied;
- *limitation* ($\mathcal{L}$) : encode the parameter k.



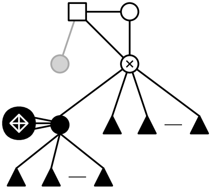
Reduction (step 1)

Let G be an instance of k-Capacitated dominating set.
Starting with $H = I(G)$ (incidence graph), we add several gadgets:

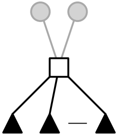
- *domination* ($\mathcal{D}$) : force G to admit a (classical) dominating set;
- *capacity* ($\mathcal{C}$) : encode the capacity function cap, and allows a selected vertex to be satisfied;
- *edge-selection* ($\mathcal{E}$) : encode the domination function dom;
- *satisfiability* ($\mathcal{S}$) : allow a non-selected vertex to be satisfied;
- *limitation* ($\mathcal{L}$) : encode the parameter k.



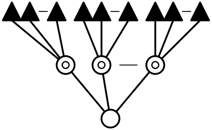
Gadgets (step 1)



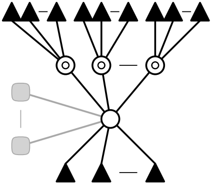
domination ($\mathcal{D}$)



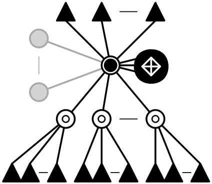
edge-selection ($\mathcal{E}$)



satisfiability ($\mathcal{S}$)



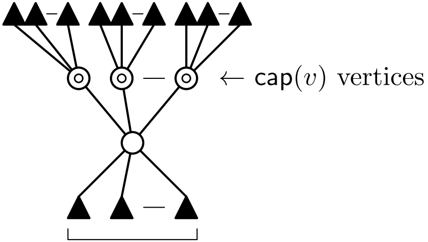
capacity ($\mathcal{C}$)



limitation ($\mathcal{L}$)

▲ = preselected vertex

Use of arbitrary long excluded intervals



$\min_\sigma\{p \mid \forall i \leq d(v),\ p+i \notin \sigma\} - \mathsf{cap}(v) - 1$ vertices

gadget *capacity* ($\mathcal{C}$)

Two-steps reduction

Step 1

k-CAPACITATED DOMINATING SET

$\preceq_{fpt}$

$[\sigma, \varrho]$-DOMINATING SET WITH PRESELECTED VERTICES

Step 2

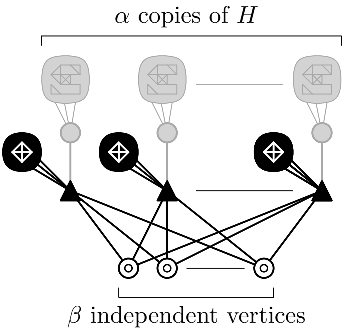
$[\sigma, \varrho]$-DOMINATING SET WITH PRESELECTED VERTICES

$\preceq_{fpt}$

$[\sigma, \varrho]$-DOMINATING SET

Reduction (second step)

Let H be an instance of $[\sigma, \varrho]$-DOMINATING SET WITH PRESELECTED VERTICES. We construct H' as follows:



$$\alpha = \min\{q \mid q \in \sigma \cap \varrho\}$$
$$\beta = \min\{p - 1 \mid \sigma_{\min} + p \in \sigma\}$$

Two-steps reduction

Step 1

k-Capacitated dominating set

$\preceq_{fpt}$

$[\sigma, \varrho]$-Dominating set with preselected vertices

Step 2

$[\sigma, \varrho]$-Dominating set with preselected vertices

$\preceq_{fpt}$

$[\sigma, \varrho]$-Dominating set

Some definitions

FPT cases

W[1]-hardness

Conclusion

Conclusion

Results

- If σ and ϱ are ultimately periodics, then $[\sigma, \varrho]$-DOMINATING SET is **FPT** when parameterized by tree-width.

Conclusion

Results

- If σ and ϱ are ultimately periodics, then $[\sigma, \varrho]$-Dominating set is **FPT** when parameterized by tree-width.
- If σ exclude arbitrary long intervals and ϱ is cofinite, then $[\sigma, \varrho]$-Dominating set is W[1]-hard when parameterized by tree-width.

Conclusion

Results

- If σ and ϱ are ultimately periodics, then $[\sigma, \varrho]$-Dominating set is **FPT** when parameterized by tree-width.
- If σ exclude arbitrary long intervals and ϱ is cofinite, then $[\sigma, \varrho]$-Dominating set is **W**[1]-hard when parameterized by tree-width.

And now?

- In which **W**$[t]$ class does this problem fall?
- What about other cases of $[\sigma, \varrho]$ (*e.g.* with bounded intervals)?
- Other **W**[1]-hard problems parameterized by tree-width?

Conclusion

Results

- If σ and ϱ are ultimately periodics, then $[\sigma, \varrho]$-DOMINATING SET is **FPT** when parameterized by tree-width.
- If σ exclude arbitrary long intervals and ϱ is cofinite, then $[\sigma, \varrho]$-DOMINATING SET is **W**[1]-hard when parameterized by tree-width.

And now?

- In which **W**$[t]$ class does this problem fall?
- What about other cases of $[\sigma, \varrho]$ (*e.g.* with bounded intervals)?
- Other **W**[1]-hard problems parameterized by tree-width?

And voilà!