Parameterized Complexity of Generalized Domination Problems

Petr A. Golovach[1], Jan Kratochvíl[2] a Ondřej Suchý[2]

[1]Department of Informatics, University of Bergen, Norway
email: petrg@ii.uib.no,

[2]Department of Applied Mathematics and Institute for Theoretical Computer Science (ITI),
Charles University, Prague, Czech Republic
email: {honza|suchy}@kam.mff.cuni.cz

35th International Workshop
on Graph-Theoretic Concepts in Computer Science,
Montpellier, France
25th June 2009

(σ, ρ)-domination - definition

Given two sets σ, ρ of nonnegative integers

Definice (Telle 94)

A set S of vertices of a graph G is (σ, ρ)-*dominating* iff

- for each vertex $v \in S$ is $|S \cap N(v)| \in \sigma$, and
- for each vertex $v \notin S$ is $|S \cap N(v)| \in \rho$.

(σ, ρ)-domination - examples

σ	ρ	Problem
$\{0, 1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Dominating Set
$\{1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Total Dominating Set
$\{0, 1, 2, \ldots\}$	$\{1\}$	Efective Dominating Set
$\{0\}$	$\{1, 2, \ldots\}$	Indep. Dominating Set
$\{0\}$	$\{0, 1, 2, \ldots\}$	Independent Set
$\{0\}$	$\{1\}$	Perfect Code
$\{r\}$	$\{0, 1, 2, \ldots\}$	Induced r-Regular subgraph

(σ, ρ)-domination - examples - Complexity

σ	ρ	Problem	Complexity
$\{0, 1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Dominating Set	NP-complete
$\{1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Total Dominating Set	NP-complete
$\{0, 1, 2, \ldots\}$	$\{1\}$	Efective Dominating Set	NP-complete
$\{0\}$	$\{1, 2, \ldots\}$	Indep. Dominating Set	NP-complete
$\{0\}$	$\{0, 1, 2, \ldots\}$	Independent Set	NP-complete
$\{0\}$	$\{1\}$	Perfect Code	NP-complete
$\{r\}$	$\{0, 1, 2, \ldots\}$	Induced r-Regular subgraph	NP-complete

(σ, ρ)-domination - examples - Complexity

σ	ρ	Problem	Complexity
$\{0, 1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Dominating Set	NP-complete
$\{1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Total Dominating Set	NP-complete
$\{0, 1, 2, \ldots\}$	$\{1\}$	Efective Dominating Set	NP-complete
$\{0\}$	$\{1, 2, \ldots\}$	Indep. Dominating Set	NP-complete
$\{0\}$	$\{0, 1, 2, \ldots\}$	Independent Set	NP-complete
$\{0\}$	$\{1\}$	Perfect Code	NP-complete
$\{r\}$	$\{0, 1, 2, \ldots\}$	Induced r-Regular subgraph	NP-complete

Theorem(Telle 1994)

(σ, ρ)-DOMINATING SET is NP-complete for any σ and ρ finite, $0 \notin \rho$.

(σ, ρ)-domination - examples - Complexity

σ	ρ	Problem	Complexity
$\{0, 1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Dominating Set	NP-complete
$\{1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Total Dominating Set	NP-complete
$\{0, 1, 2, \ldots\}$	$\{1\}$	Efective Dominating Set	NP-complete
$\{0\}$	$\{1, 2, \ldots\}$	Indep. Dominating Set	NP-complete
$\{0\}$	$\{0, 1, 2, \ldots\}$	Independent Set	NP-complete
$\{0\}$	$\{1\}$	Perfect Code	NP-complete
$\{r\}$	$\{0, 1, 2, \ldots\}$	Induced r-Regular subgraph	NP-complete

Theorem(Telle 1994)

(σ, ρ)-DOMINATING SET is NP-complete for any σ and ρ finite, $0 \notin \rho$.

- on interval graphs - in P for any σ, ρ finite
  [Kratochvíl, Manuel, Miller 1995]
- Chordal graphs, Degenerated graphs - P/NP-c. dichotomy
  [Golovach, Kratochvíl WG 2007, TAMC 2008]

(σ, ρ)-domination - examples - Parameterized complexity

σ	ρ	Problem	Par. Compl.
$\{0, 1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Dominating Set	W[2]-complete
$\{1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Total Dominating Set	W[2]-hard
$\{0, 1, 2, \ldots\}$	$\{1\}$	Efective Dominating Set	W[1]-hard
$\{0\}$	$\{1, 2, \ldots\}$	Indep. Dominating Set	W[2]-complete
$\{0\}$	$\{0, 1, 2, \ldots\}$	Independent Set	W[1]-complete
$\{0\}$	$\{1\}$	Perfect Code	W[1]-complete
$\{r\}$	$\{0, 1, 2, \ldots\}$	Induced r-Regular subgraph	W[1]-hard

Theorem(Telle 1994)

(σ, ρ)-Dominating Set is NP-complete for any σ and ρ finite, $0 \notin \rho$.

(σ, ρ)-domination - examples - Parameterized complexity

σ	ρ	Problem	Par. Compl.
$\{0, 1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Dominating Set	W[2]-complete
$\{1, 2, \ldots\}$	$\{1, 2, \ldots\}$	Total Dominating Set	W[2]-hard
$\{0, 1, 2, \ldots\}$	$\{1\}$	Efective Dominating Set	W[1]-hard
$\{0\}$	$\{1, 2, \ldots\}$	Indep. Dominating Set	W[2]-complete
$\{0\}$	$\{0, 1, 2, \ldots\}$	Independent Set	W[1]-complete
$\{0\}$	$\{1\}$	Perfect Code	W[1]-complete
$\{r\}$	$\{0, 1, 2, \ldots\}$	Induced r-Regular subgraph	W[1]-hard

Theorem(Telle 1994)

(σ, ρ)-DOMINATING SET is NP-complete for any σ and ρ finite, $0 \notin \rho$.

Theorem

(σ, ρ)-DOMINATING SET OF SIZE k is W[1]-complete for any σ and ρ finite, $0 \notin \rho$.

Overview of our results

σ	ρ	size	graphs	parameterized complexity
finite	finite, $0 \notin$	$= k, \leq k$		W[1]-hard
recursive	finite	$= k, \leq k$		in W[1]
finite or co-finite	finite or co-finite	$\geq n - k$		FPT
EVEN or **ODD**	**EVEN** or **ODD**	$= n - k$, $\geq n - k$		W[1]-hard
finite or co-finite	finite or co-finite	$= k, \leq k$	locally minor excluding	FPT
$\max \sigma < \min \rho$		$= k, \leq k$	degenerated	FPT

Even/Odd Set of size (at least) $|R| - k$ is W[1]-hard

W[1]-hardness - starting problem

EXACT SATISFIABILITY

Instance: A Boolean formula ϕ in conjunctive normal form, without negations.
Parameter: k.
Question: Does ϕ allow a satisfying truth assignment such that at most k variables have value *true*, and each clause of ϕ contains exactly one variable which evaluate to *true*?

EXACT SATISAFIABILITY is known W[1]-complete problem.

W[1]-hardness - starting problem

At Most α-Exact Satisfiability

Instance: A Boolean formula ϕ in conjunctive normal form, without negations.
Parameter: k.
Question: Does ϕ allow a satisfying truth assignment such that at most k variables have value *true*, and each clause of ϕ contains at most α variables which evaluate to *true*?

Exact Satisafiability is known W[1]-complete problem.

Lemma

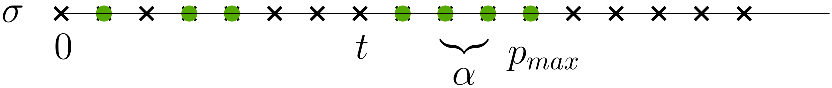
For any fixed α: At Most α-Exact Satisfiability *is W[1]-hard.*

Proof

Easy reduction from Exact Satisafiability.

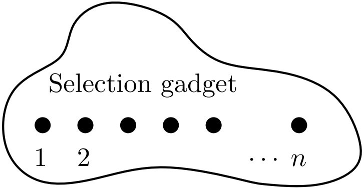
W[1]-hardness - idea of the reduction I

Denote $p_{max} := \max \sigma$, $q_{max} := \max \rho$,
$t := \max\{i \mid i \notin \sigma, i+1 \in \sigma\}$, $\alpha := p_{max} - t$

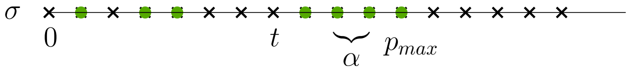


We reduce At Most α-Exact Satisfiability with n variables to (σ, ρ)-dominating set using two gadgets:

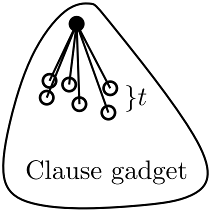
- Selection gadget is a graph with n roots such that
  - in each (σ, ρ)-dominating set is exactly one root, and
  - for each root there is a (σ, ρ)-dominating set containing it.



W[1]-hardness - idea of the reduction II

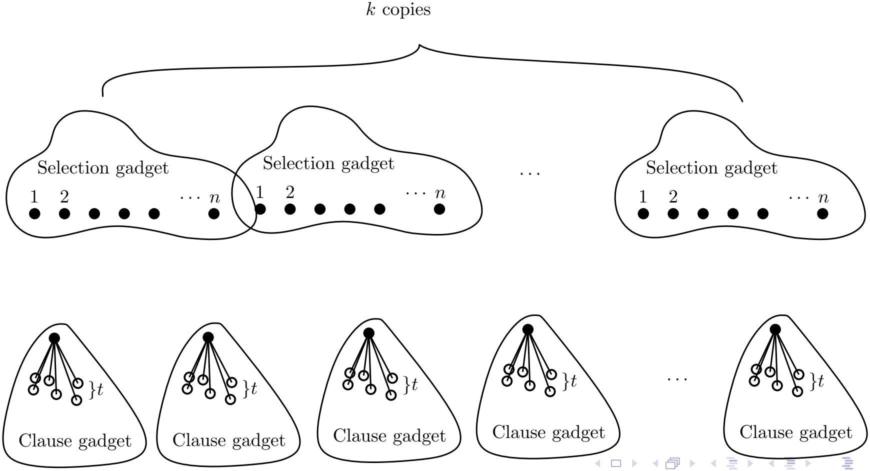


- Clause gadget is a graph with one root, which
  - is in any (σ, ρ)-dominating set, and
  - has exactly t neighbors in any (σ, ρ)-dominating set, and hence
  - needs at least one, but at most α further neighbors.



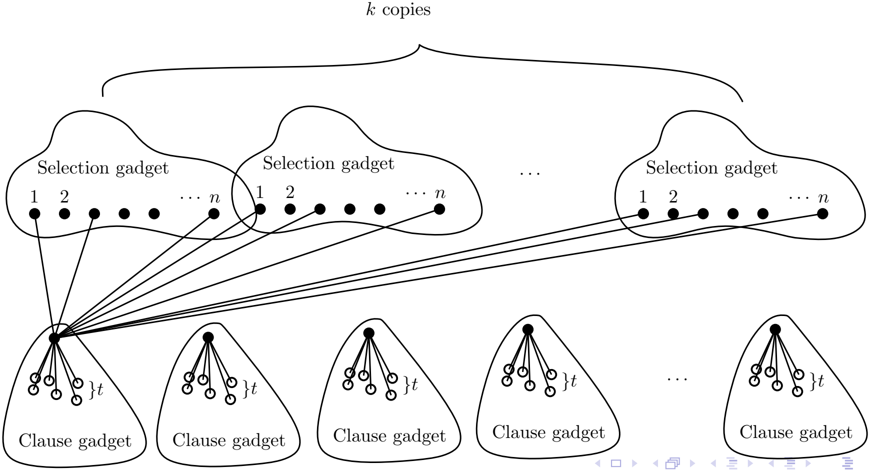
W[1]-hardness - idea of the reduction III

- take k copies of the selection gadget and
- a copy of the clause gadget for each clause and
- connect the root of the clause gadget to the roots of selection gadget corresponding to the variables that appear in the clause



W[1]-hardness - idea of the reduction III

- take k copies of the selection gadget and
- a copy of the clause gadget for each clause and
- connect the root of the clause gadget to the roots of selection gadget corresponding to the variables that appear in the clause



W[1]-membership - machine characterisation

Non-deterministic RAM (NRAM) is RAM with a single non-deterministic instruction "GUESS(n)" added, which guesses a number between 1 and n.

Definition

An NRAM program $\mathbb{P}$ is *tail-nondeterministic k-restricted* if there are computable functions f and g and a polynomial p such that on every run with input $(x, k) \in \Sigma^* \times \mathbb{N}, n := |x|$ the program $\mathbb{P}$

- performs at most $f(k) \cdot p(n)$ steps;
- uses at most the first $f(k) \cdot p(n)$ registers;
- contains numbers $\leq f(k) \cdot p(n)$ in any register at any time; and
- all nondeterministic steps are among the last $g(k)$ steps of the computation.

W[1]-membership - machine characterisation

Non-deterministic RAM (NRAM) is RAM with a single non-deterministic instruction "GUESS(n)" added, which guesses a number between 1 and n.

Definition

An NRAM program $\mathbb{P}$ is *tail-nondeterministic k-restricted* if there are computable functions f and g and a polynomial p such that on every run with input $(x, k) \in \Sigma^* \times \mathbb{N}, n := |x|$ the program $\mathbb{P}$

- performs at most $f(k) \cdot p(n)$ steps;
- uses at most the first $f(k) \cdot p(n)$ registers;
- contains numbers $\leq f(k) \cdot p(n)$ in any register at any time; and
- all nondeterministic steps are among the last $g(k)$ steps of the computation.

Theorem (Flum and Grohe 2006)

A parameterized problem P is in W[1] if and only if there is a tail-nondeterministic k-restricted NRAM program deciding P.

Algorithm

1. For $r := 1$ To $q_{max} + 1$ do ForAll $R \in \binom{V}{r}$ do

$$B(R) := |\bigcap_{u \in R} N_G(u)| = |\{v | v \in V, \forall u \in R : uv \in E\}|;$$

Algorithm

1. For $r := 1$ To $q_{max} + 1$ do ForAll $R \in \binom{V}{r}$ do
$$B(R) := |\bigcap_{u \in R} N_G(u)| = |\{v | v \in V, \forall u \in R : uv \in E\}|;$$
2. Guess k (distinct) vertices $v_1, \ldots, v_k$, denote $S = \{v_1, \ldots, v_k\}$;
3. For $i := 1$ To k do If $|\{v_j | v_i v_j \in E\}| \notin \sigma$ then REJECT;

Algorithm

1. For $r := 1$ To $q_{max} + 1$ do ForAll $R \in \binom{V}{r}$ do
$$B(R) := |\bigcap_{u \in R} N_G(u)| = |\{v | v \in V, \forall u \in R : uv \in E\}|;$$
2. Guess k (distinct) vertices $v_1, \ldots, v_k$, denote $S = \{v_1, \ldots, v_k\}$;
3. For $i := 1$ To k do If $|\{v_j | v_i v_j \in E\}| \notin \sigma$ then REJECT;
4. For $r := q_{max} + 1$ Downto 1 do
$$D(r) := \sum_{R \in \binom{S}{r}} (B(R) - |\bigcap_{u \in R} N_G(u) \cap S|) = \sum_{R \in \binom{S}{r}} |\{v \in V \setminus S \mid R \subseteq N_G(v)\}|$$

Algorithm

1. For $r := 1$ To $q_{max} + 1$ do ForAll $R \in \binom{V}{r}$ do
$$B(R) := |\bigcap_{u \in R} N_G(u)| = |\{v | v \in V, \forall u \in R : uv \in E\}|;$$
2. Guess k (distinct) vertices $v_1, \ldots, v_k$, denote $S = \{v_1, \ldots, v_k\}$;
3. For $i := 1$ To k do If $|\{v_j | v_i v_j \in E\}| \notin \sigma$ then REJECT;
4. For $r := q_{max} + 1$ Downto 1 do
$$D(r) := \sum_{R \in \binom{S}{r}} (B(R) - |\bigcap_{u \in R} N_G(u) \cap S|) = \sum_{R \in \binom{S}{r}} |\{v \in V \setminus S \mid R \subseteq N_G(v)\}|$$
$$C(r) := D(r) - \sum_{t=r+1}^{q_{max}} \left(\binom{t}{r} \cdot C(t) \right)$$
If $r \notin \rho$ and $C(r) \neq 0$ then REJECT;

Algorithm

1. For $r := 1$ To $q_{max} + 1$ do ForAll $R \in \binom{V}{r}$ do
$$B(R) := |\bigcap_{u \in R} N_G(u)| = |\{v | v \in V, \forall u \in R : uv \in E\}|;$$
2. Guess k (distinct) vertices $v_1, \ldots, v_k$, denote $S = \{v_1, \ldots, v_k\}$;
3. For $i := 1$ To k do If $|\{v_j | v_i v_j \in E\}| \notin \sigma$ then REJECT;
4. For $r := q_{max} + 1$ Downto 1 do
$$D(r) := \sum_{R \in \binom{S}{r}} (B(R) - |\bigcap_{u \in R} N_G(u) \cap S|) = \sum_{R \in \binom{S}{r}} |\{v \in V \setminus S \mid R \subseteq N_G(v)\}|$$
$$C(r) := D(r) - \sum_{t=r+1}^{q_{max}} \left(\binom{t}{r} \cdot C(t) \right)$$
If $r \notin \rho$ and $C(r) \neq 0$ then REJECT;
5. If $\sum_{r \in \rho} C(r) = n - k$ then ACCEPT;
Else REJECT;

Dual parameterization

Question: Is there a (σ, ρ)-dominating set of size at least $n - k$?

Dual parameterization

Question: Is there a (σ, ρ)-dominating set of size at least $n - k$?

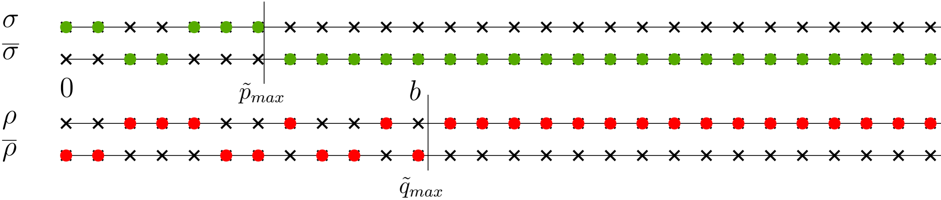
For $X \subset \mathbb{N}_0$ denote $\overline{X} := \mathbb{N}_0 \setminus X$ the complement of X.

Theorem

Let either σ or $\overline{\sigma}$ be finite, and similarly either ρ or $\overline{\rho}$ finite.
Then (σ, ρ)-DOMINATING SET OF SIZE AT LEAST $n - k$ is FPT.

Dual in FPT - proof I

Denote $\tilde{p}_{max}$ the maximum of σ or $\overline{\sigma}$ (of the finite one of them). Similarly denote $\tilde{q}_{max}$ the maximum of ρ or $\overline{\rho}$.
Finally b is $\max\{\tilde{p}_{max}, \tilde{q}_{max}\}$.

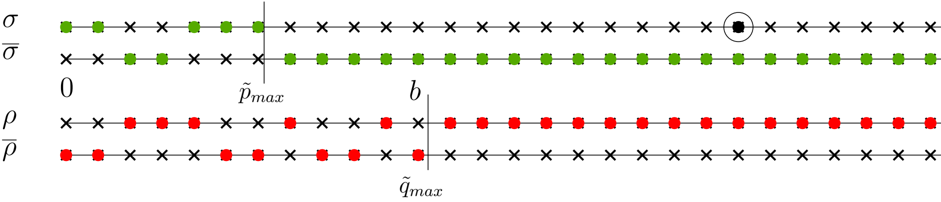


We say that a vertex is

- Unsatisfied - if it has a wrong number of neighbors in S
- Big - if it is of degree at least $b + k$
- Small, Satisfied - ...

Dual in FPT - proof I

Denote $\tilde{p}_{max}$ the maximum of σ or $\overline{\sigma}$ (of the finite one of them). Similarly denote $\tilde{q}_{max}$ the maximum of ρ or $\overline{\rho}$.
Finally b is $\max\{\tilde{p}_{max}, \tilde{q}_{max}\}$.



We say that a vertex is

- Unsatisfied - if it has a wrong number of neighbors in S
- Big - if it is of degree at least $b + k$
- Small, Satisfied - ...

Algorithm for Dual

Procedure `Exclude`(S)

- If *all vertices satisfied* then Return(S);Exit;
- If $|S| = n - k$ then Halt;
- Let v be an unsatisfied vertex;
- If v *big* then
  - If $v \in S$ *and* ρ *infinite* then `Exclude`$(S \setminus v)$;
  - Else Halt;

  Else
  - If $v \in S$ then `Exclude`$(S \setminus v)$;
  - Let $\{u_1, \ldots, u_r\} = S \cap N(v)$ be the neighbors of v in S;
  - If $r = 0$ then Halt;
  - For $i := 1$ to $\min\{b+1, r\}$ do `Exclude`$(S \setminus \{u_i\})$;

Usage: Call `Exclude`(V);

Parity sets

Denote **EVEN** the set of even nonnegative integers and
ODD the set of odd nonnegative integers
The following is a well-known W[1]-hard problem

Odd Set

Input: Red-Blue bipartite graph $G = (R, B, E)$
Parameter: k.
Question: Is there a set of $S \subseteq R$ of exactly k red vertices,
in which every blue vertex has odd number of neighbors?

Parity sets

Denote **EVEN** the set of even nonnegative integers and
ODD the set of odd nonnegative integers
The following is a well-known W[1]-hard problem

Odd Set

Input: Red-Blue bipartite graph $G = (R, B, E)$
Parameter: k.
Question: Is there a set of $S \subseteq R$ of exactly k red vertices,
in which every blue vertex has odd number of neighbors?

Lemma

Odd Set is W[1]-hard, even if every blue vertex has odd degree.

Parity sets

Denote **EVEN** the set of even nonnegative integers and
ODD the set of odd nonnegative integers
The following is a well-known W[1]-hard problem

Odd Set

Input: Red-Blue bipartite graph $G = (R, B, E)$
Parameter: k.
Question: Is there a set of $S \subseteq R$ of exactly k red vertices,
in which every blue vertex has odd number of neighbors?

Lemma

Odd Set is W[1]-hard, even if every blue vertex has odd degree.

Theorem

Odd Set of size (at least) $|R| - k$, and
Even Set of size (at least) $|R| - k$ are W[1]-hard problems.

$\sigma, \rho \in \{\textbf{EVEN}, \textbf{ODD}\}$

Theorem

Let $\sigma, \rho \in \{\textbf{EVEN}, \textbf{ODD}\}$. Then (σ, ρ)-Dominating Set of Size (at least) $n - k$ is W[1]-hard problem.

$\sigma, \rho \in \{\textbf{EVEN}, \textbf{ODD}\}$

Theorem

Let $\sigma, \rho \in \{\textbf{EVEN}, \textbf{ODD}\}$. Then (σ, ρ)-Dominating Set of Size (at least) $n - k$ is W[1]-hard problem.

Lemma

Even Set of Size $|R| - k$ is W[1]-hard, even if all red vertices have even degree.

Proof of Lemma

Replace each vertex in B by two copies with the same neighborhood.

$\sigma, \rho \in \{\textbf{EVEN}, \textbf{ODD}\}$

Theorem

Let $\sigma, \rho \in \{\textbf{EVEN}, \textbf{ODD}\}$. Then (σ, ρ)-Dominating Set of Size (at least) $n - k$ is W[1]-hard problem.

Lemma

Even Set of Size $|R| - k$ is W[1]-hard, even if all red vertices have even degree.

Proof of Lemma

Replace each vertex in B by two copies with the same neighborhood.

Proof of Theorem for the case $\sigma = \rho = \textbf{EVEN}$

If S is an even set of size $|R| - k$ and all the degrees in R are even, then $S \cup B$ is (**EVEN**,**EVEN**)-dominating set of size $n - k$.

Overview of our results

σ	ρ	size	graphs	parameterized complexity
finite	finite, $0 \notin$	$= k, \leq k$		W[1]-hard
recursive	finite	$= k, \leq k$		in W[1]
finite or co-finite	finite or co-finite	$\geq n - k$		FPT
EVEN or **ODD**	**EVEN** or **ODD**	$= n - k$, $\geq n - k$		W[1]-hard
finite or co-finite	finite or co-finite	$= k, \leq k$	locally minor excluding	FPT
$\max \sigma < \min \rho$		$= k, \leq k$	degenerated	FPT

Even/Odd Set of size (at least) $|R| - k$ is W[1]-hard

Open problems / Research directions

- Clasification of (reasonable) infinite σ, ρ
- r-Regular Induced Subgraph of Size $n-k$
  has a kernel of size $O(kr(r+k)^2)$[Moser and Thilikos 2006].
  Do our FPT problems admit polynomial kernel?
- Restricted graph classes
- ...

Thank you for your attention!