Well-Structured Modulators: FPT Algorithms and Kernels

Eduard Eiben Robert Ganian Stefan Szeider

Algorithms & Complexity Group
TU Wien

GROW 2015, Aussios, France,
October 13, 2015

Introduction

- General goal: identify conditions which allow solution of NP-hard graph problems
- We want to get results for a wide range of problems
- Many problems can be captured by Monadic Second Order (MSO) logic

MSO (MS_1) logic

- used in Courcelle's Theorem (no edge quantification)
- Quantify over vertices and vertex sets
- Atoms: edges between vertices, set inclusion, equality
- Example: $\exists x \forall y : edge(x, y) \vee x = y$.

MSO (MS_1) logic

- used in Courcelle's Theorem (no edge quantification)
- Quantify over vertices and vertex sets
- Atoms: edges between vertices, set inclusion, equality
- Example: $\exists x \forall y : edge(x, y) \vee x = y$.

Expressible problems

- 3-colorability, 3-clique cover, 3-partition into trees ...
- $\exists A, B, C\ \forall x, y$: *partition* and *neighborhood* conditions

MSO (MS_1) logic

- used in Courcelle's Theorem (no edge quantification)
- Quantify over vertices and vertex sets
- Atoms: edges between vertices, set inclusion, equality
- Example: $\exists x \forall y : edge(x, y) \vee x = y$.

Expressible problems

- 3-colorability, 3-clique cover, 3-partition into trees ...
- $\exists A, B, C\ \forall x, y$: *partition* and *neighborhood* conditions

Definition ($\mathrm{MSO\text{-}MC}_\varphi$)

Instance: A graph G.
Question: Does $G \models \varphi$ hold?

MSO (MS_1) logic

- used in Courcelle's Theorem (no edge quantification)
- Quantify over vertices and vertex sets
- Atoms: edges between vertices, set inclusion, equality
- Example: $\exists x \forall y : edge(x, y) \vee x = y$.

Expressible problems

- 3-colorability, 3-clique cover, 3-partition into trees ...
- $\exists A, B, C \; \forall x, y$: *partition* and *neighborhood* conditions

Definition ($\textsc{MSO-Opt}_{\varphi}$)

Instance: A graph G and an integer r.
Question: Is there $X \subseteq V(G)$ s.t. $G \models \varphi(X)$ and $|X| \leq r$?

Structural approach

- MSO model checking NP-hard in general
  ...but efficiently solvable by using the structure of inputs

Structural approach

- MSO model checking NP-hard in general
  ...but efficiently solvable by using the structure of inputs
- How to measure the "structure" of inputs? $\rightarrow$ Parameterized Complexity
- Idea: use a parameter k to measure how "well-structured" the graph is
  - smaller k = more structured
- Develop algorithms which run well if the parameter is small
  - FPT algorithms: $f(k) \cdot n^{\mathcal{O}(1)}$ runtime
  - Kernelization: see later

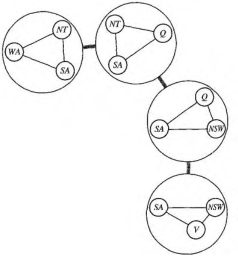
Choice of parameter

- A good parameter should
  - be *small* for as many inputs as possible, but
  - allow the design of FPT algorithms for many problems
- Two different approaches to parameter design:
  - Decompositions
  - Modulators

Decomposition approach

Parameters are associated with a *decomposition* which can be used to solve problems.

- Treewidth



Theorem (Courcelle)

For an n-vertex graph G with treewidth k and an MSO sentence φ, we can solve $\mathrm{MSO\text{-}MC}_{\varphi}$ in time $f(k, \varphi) \cdot n$.

- treewidth large on dense graphs

Decomposition approach

- Clique-width
  - more general than treewidth, low on some dense graphs
  - allows FPT-time MSO model checking if decompositions are provided
  - cannot compute decompositions

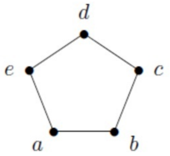
Decomposition approach

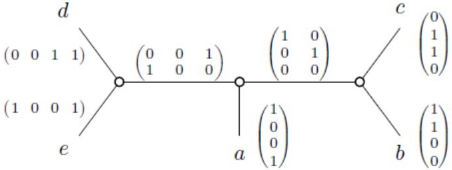
- Rank-width

Theorem (Ganian, Hliněný)

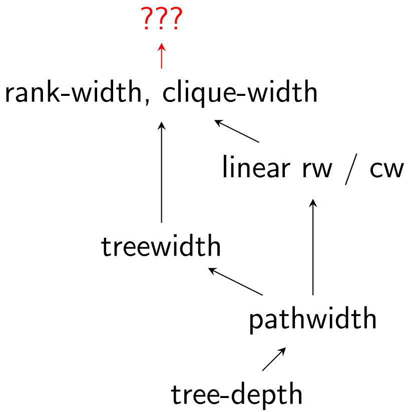
For an n-vertex graph G with rank-width k and an MSO sentence φ, we can solve $\mathrm{MSO\text{-}MC}_\varphi$ in time $f(k, \varphi) \cdot n^3$.

  - as general as clique-width—bounded from below and above by a function of clique-width
  - can compute decompositions [Hliněný, Oum]





Decomposition-based width measures



Modulator approach

Parameters measure how “close” a graph is to a graph class $\mathcal{H}$

Also very successful in related fields (backdoors in SAT, CSP...)

Modulator approach

Parameters measure how "close" a graph is to a graph class $\mathcal{H}$

Also very successful in related fields (backdoors in SAT, CSP...)

- k is the number of vertices that need to be deleted to get to $\mathcal{H}$
- Vertex Cover and Feedback Vertex Set are special cases of modulators
- allow the use of a vast range of work on specific graph classes

This talk

Combine the 2 approaches to introduce a family of "hybrid" parameters.

Goal for FPT algorithms:

- more general than rank-width and modulators

This talk

Combine the 2 approaches to introduce a family of “hybrid” parameters.

Goal for FPT algorithms:

- more general than rank-width and modulators
- computable in FPT time

This talk

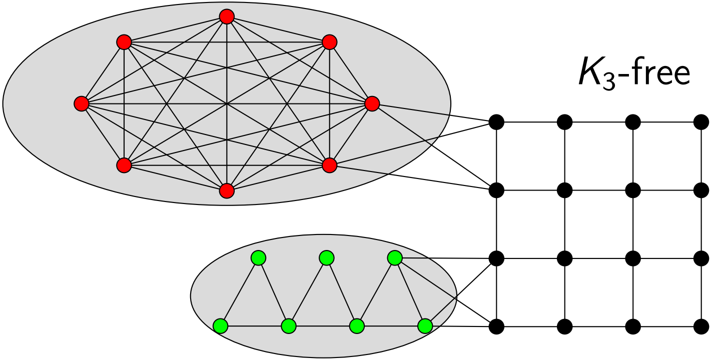
Combine the 2 approaches to introduce a family of “hybrid” parameters.

Goal for FPT algorithms:

- more general than rank-width and modulators
- computable in FPT time
- allow FPT-time MSO model checking
  - under certain conditions...

Well-structured modulators

Basic idea: what if the graph has a large but well-structured modulator to $\mathcal{H}$?



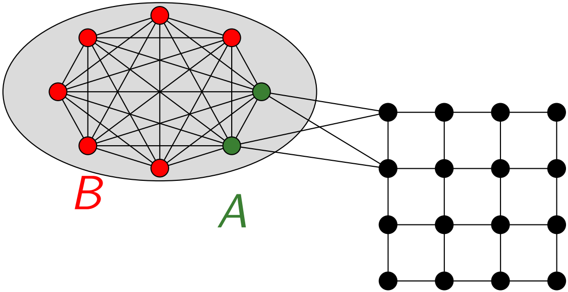
A graph with a 2-well-structured modulator to K_3-free graphs

!!!Edges going into modulator must be controlled!!!

Well-structured modulators

We use splits to control edges going into the modulator.

A set of vertices $X \subseteq V(G)$ is a split-module if it can be partitioned into $\{A, B\}$ such that A is completely connected to neighborhood of X and B does not have neighbors outside X.

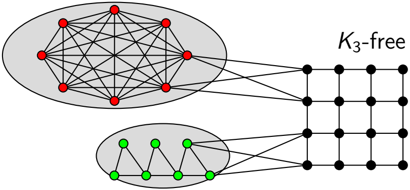


Well-structured modulators

Definition

A set $\mathbf{X}$ of pairwise-disjoint split-modules of a graph G is called a k-well-structured modulator to $\mathcal{H}$ if

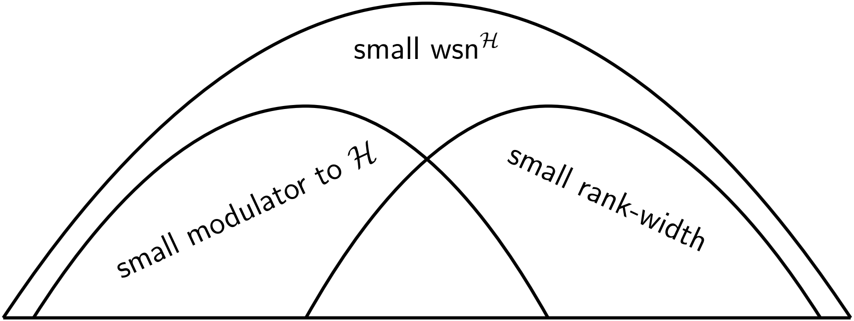
- $|\mathbf{X}| \leq k$, and
- $\bigcup_{X_i \in \mathbf{X}} X_i$ is a modulator to $\mathcal{H}$, and
- rank-width of $G[X_i] \leq k$ for each $X_i \in \mathbf{X}$.



A graph with a 2-well-structured modulator to K_3-free graphs

Well-structured modulators vs. other parameters

Well-structure number ($\mathrm{wsn}^{\mathcal{H}}$): minimum k such that G has a k-well-structured modulator to $\mathcal{H}$



Finding well-structured modulators

Theorem

Finding a k-well-structured modulator to any $\mathcal{H}$ characterizable by a finite set of forbidden induced subgraphs is FPT.

Examples for $\mathcal{H}$: split graphs, P_5-free graphs, graphs of bounded degree, triangle-free graphs, claw-free graphs...

Finding well-structured modulators

Theorem

Finding a k-well-structured modulator to any $\mathcal{H}$ characterizable by a finite set of forbidden induced subgraphs is FPT.

Examples for $\mathcal{H}$: split graphs, P_5-free graphs, graphs of bounded degree, triangle-free graphs, claw-free graphs...

How this works:

- We use [Cunningham, 1982] and algorithm of [Gioan, Paul, Tedder, Corneil, 2014] to partition vertices into maximal split-modules of rank-width at most k.
- Reduce to d-hitting set.

Solving $\mathrm{MSO\text{-}MC}_\varphi$

Theorem

For any MSO formula φ such that $\mathrm{MSO\text{-}MC}_\varphi$ is FPT parameterized by modulator-size to $\mathcal{H}$, $\mathrm{MSO\text{-}MC}_\varphi$ is FPT parameterized by wsn$^{\mathcal{H}}(G)$.

Blue part is a necessary condition.

Solving $\mathrm{MSO\text{-}MC}_\varphi$

Theorem

For any MSO formula φ such that $\mathrm{MSO\text{-}MC}_\varphi$ is FPT parameterized by modulator-size to $\mathcal{H}$, $\mathrm{MSO\text{-}MC}_\varphi$ is FPT parameterized by $wsn^{\mathcal{H}}(G)$.

Blue part is a necessary condition.

Note that this captures not only the generality of $\mathrm{MSO\text{-}MC}_\varphi$, but also applies to many choices of $\mathcal{H}$.

$\mathrm{MSO\text{-}MC}_\varphi$ – Example

c-Coloring is FPT parameterized by the size of a modulator to P_5-free graphs. [Golovach, Paulusma, Song 2014 + Cai 2003]

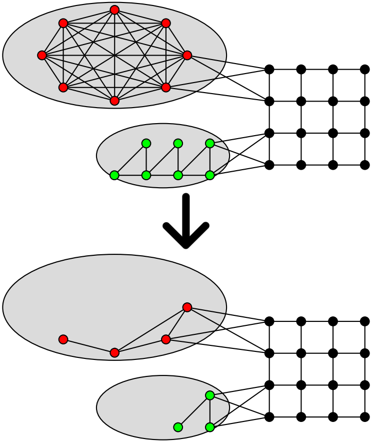
Hence:

Corollary

c-Coloring is FPT parameterized by $\mathit{wsn}^{P_5\text{-free}}$.

Solving $\mathrm{MSO\text{-}MC}_\varphi$ – Proof

Idea: replace big split-modules by small ones



Now use necessary condition

What about $\text{MSO-}\textsc{Opt}_\varphi$?

Surprisingly, there is a (provable) difference in difficulty between $\text{MSO-}\textsc{Opt}_\varphi$ and MSO-MC_φ par. by wsn.

Theorem

There exists an MSO formula φ and graph class $\mathcal{H}$ (satisfying same conditions as for previous theorem) such that $\text{MSO-}\textsc{Opt}_\varphi$ is paraNP-hard parameterized by wsn$^\mathcal{H}$.

Solving other problems

Theorem

Minimum Vertex Cover and Maximum Clique are FPT parameterized by $wsn^{\mathcal{H}}$ iff they are polytime tractable on $\mathcal{H}$.

Weaker necessary condition – not sufficient for $\mathrm{MSO\text{-}MC}_{\varphi}$!
Choices of $\mathcal{H}$ for Minimum Vertex Cover:

- $(2K_2, C_4, C_5)$-free graphs (split graphs);
- P_5-free graphs [Lokshtanov, Vatshelle, Villanger, 2014];
- fork-free graphs [Alekseev, 2004];
- (banner, $T_{2,2,2}$)-free graphs and (banner, $K_{3,3}$-e, twin-house)-free graphs [Gerber, Brandstadt, Lozin, 2001-2003].

Kernelization

Kernelization studies the efficient preprocessing and compression of inputs.
Basic idea:

- A *kernelization algorithm* $\mathbb{A}$ takes an instance (I, p) and outputs an instance (I', p') (the *kernel*) such that:
  - $\mathbb{A}$ runs in polynomial time,
  - $(I, p) \in P$ iff $(I', p') \in P$, and
  - $|I'| + p' \leq f(p)$ for a (fixed) computable function f.

Kernelization

Kernelization studies the efficient preprocessing and compression of inputs.
Basic idea:

- A *kernelization algorithm* $\mathbb{A}$ takes an instance (I, p) and outputs an instance (I', p') (the *kernel*) such that:
  - $\mathbb{A}$ runs in polynomial time,
  - $(I, p) \in P$ iff $(I', p') \in P$, and
  - $|I'| + p' \leq f(p)$ for a (fixed) computable function f.

If f is a polynomial function, then we speak of *polynomial kernels* – these are of particular interest.

Well-Structured Modulators for Kernelization

No polykernels when using rank-width.

!BUT! Modulators often give polykernels for various problems.

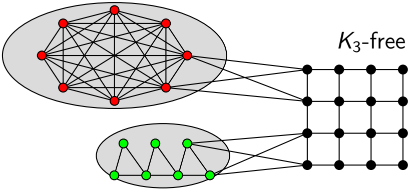
Can we exploit structure of modulators also for kernels?

(k, c)-Well-Structured Modulators

Definition

A set $\mathbf{X}$ of pairwise-disjoint split-modules of a graph G is called a (k, c)-well-structured modulator (wsm) to $\mathcal{H}$ if

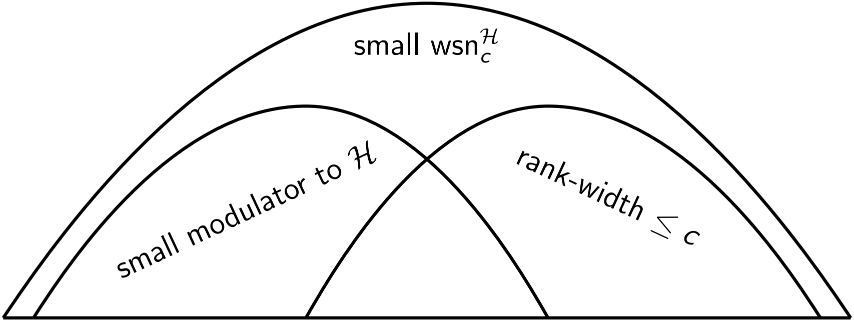
- $|\mathbf{X}| \leq k$, and
- $\bigcup_{X_i \in \mathbf{X}} X_i$ is a modulator to $\mathcal{H}$, and
- rank-width of $G[X_i] \leq c$ for each $X_i \in \mathbf{X}$.



A graph with a $(2, 1)$-wsm to K_3-free graphs

Well-structured modulators vs. other parameters

Well-structure number ($\text{wsn}_c^{\mathcal{H}}$): minimum k such that G has a (k, c)-well-structured modulator to $\mathcal{H}$



- [MFCS 2013]: special case of $\text{wsn}_c^{\mathcal{H}}$, where $\mathcal{H}$ is empty and there are additional restrictions on split-modules.
- [Brno+Aachen, ESA 2013]: special case of $\text{wsn}_c^{\mathcal{H}}$, where bags have size 1 and $\mathcal{H}$ is the class of constant treedepth.

WSMs for kernels

Differences compared to FPT algorithms:

- Finding WSMs for kernels
  - need to approximate in polytime (can't afford FPT time)

WSMs for kernels

Differences compared to FPT algorithms:

- Finding WSMs for kernels
  - need to approximate in polytime (can't afford FPT time)
- Using WSMs for kernels
  - Need a replacement procedure which only takes polytime (using constant rank-width)

WSMs for kernels

Differences compared to FPT algorithms:

- Finding WSMs for kernels
  - need to approximate in polytime (can't afford FPT time)
- Using WSMs for kernels
  - Need a replacement procedure which only takes polytime (using constant rank-width)

Theorem

For every MSO sentence φ and every graph class $\mathcal{H}$ such that

1. $\mathrm{MSO\text{-}MC}_\varphi$ *admits a polykernel par. by modulator-size to $\mathcal{H}$, and*
2. *a (k, c)-wsm to $\mathcal{H}$ can be (poly)-approximated in polytime,*

$\mathrm{MSO\text{-}MC}_\varphi$ *admits a polykernel.*

WSMs for kernels – applications

- If $\mathcal{H}$ is the class of empty graphs, we lift our previous kernelization results of [MFCS 2013] and our main theorem also holds for $\textsc{MSO-Opt}_\varphi$.

WSMs for kernels – applications

- If $\mathcal{H}$ is the class of empty graphs, we lift our previous kernelization results of [MFCS 2013] and our main theorem also holds for $\mathrm{MSO}\text{-}\mathrm{O{\scriptstyle PT}}_\varphi$.
- If $\mathcal{H}$ is the class of edgeless graphs, we lift vertex cover as a parameter. On graphs of bounded expansion, **every** $\mathrm{MSO}\text{-}\mathrm{MC}_\varphi$ and $\mathrm{MSO}\text{-}\mathrm{O{\scriptstyle PT}}_\varphi$ problem admits a polykernel for $\mathsf{wsn}_c^{\mathcal{H}}$.

WSMs for kernels – applications

- If $\mathcal{H}$ is the class of empty graphs, we lift our previous kernelization results of [MFCS 2013] and our main theorem also holds for $\text{MSO-}\textsc{Opt}_\varphi$.
- If $\mathcal{H}$ is the class of edgeless graphs, we lift vertex cover as a parameter. On graphs of bounded expansion, **every** MSO-MC_φ and $\text{MSO-}\textsc{Opt}_\varphi$ problem admits a polykernel for $\mathsf{wsn}_c^{\mathcal{H}}$.
- If $\mathcal{H}$ is the class of forests, we lift feedback vertex set as a parameter. **Every** MSO-MC_φ problem admits a linear kernel parameterized by $\mathsf{wsn}_c^{\mathcal{H}}$ on graphs of **bounded degree**.

WSMs for kernels – applications

- If $\mathcal{H}$ is the class of empty graphs, we lift our previous kernelization results of [MFCS 2013] and our main theorem also holds for $\mathrm{MSO\text{-}O{\scriptsize PT}}_\varphi$.
- If $\mathcal{H}$ is the class of edgeless graphs, we lift vertex cover as a parameter. On graphs of bounded expansion, **every** $\mathrm{MSO\text{-}MC}_\varphi$ and $\mathrm{MSO\text{-}O{\scriptsize PT}}_\varphi$ problem admits a polykernel for $\mathsf{wsn}_c^{\mathcal{H}}$.
- If $\mathcal{H}$ is the class of forests, we lift feedback vertex set as a parameter. **Every** $\mathrm{MSO\text{-}MC}_\varphi$ problem admits a linear kernel parameterized by $\mathsf{wsn}_c^{\mathcal{H}}$ on graphs of **bounded degree**.
- If $\mathcal{H}$ is a class of bounded treedepth, we lift previous results of [Aachen+Brno, ESA2013]. **Every** $\mathrm{MSO\text{-}MC}_\varphi$ problem admits a linear kernel parameterized by $\mathsf{wsn}_c^{\mathcal{H}}$ on graphs of **bounded expansion**.

Final remarks

We introduced a family of "hybrid" parameters that:

- are more general than modulator-size (and rank-width)
- are approximable in polytime (computable in FPT time)
- allow kernelization (FPT algorithms) for MSO decision problems
  - under certain conditions

What else lies above rank-width?

Can we replace splits with less restrictive conditions?

Thank you for your attention!