On the Parameterized Complexity of Graph Modification towards First-Order Logic Properties

Theorietag 2025
Workshop on Algorithms, Complexity and Logic

Marlene Gründel

FRIEDRICH-SCHILLER-UNIVERSITÄT JENA

Graph Modification Problems

Let P be a graph property (e.g. being clique, having diameter at most two, not containing any triangle, ...).

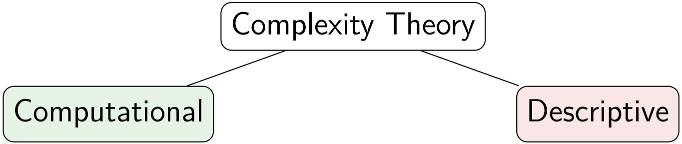
GRAPH MODIFICATION TOWARDS PROPERTY P

Input: A graph G and $k \in \mathbb{N}$.
Question: Do k modifications suffice to make G satisfy P?

Examples for modifications:

- VERTEX REMOVAL TOWARDS PROPERTY P
- EDGE REMOVAL TOWARDS PROPERTY P
- EDGE COMPLETION TOWARDS PROPERTY P
- EDGE EDITING TOWARDS PROPERTY P (Edge Removal + Completion)

Two Kinds of Complexity



Computational studies the computational resources that are necessary to decide a problem

Descriptive studies the logical resources that are necessary to define a problem

Graph Modification Problems

GRAPH MODIFICATION TOWARDS FOL-formula Φ

Input: A graph G and $k \in \mathbb{N}$.
Question: Do k modifications suffice to make G satisfy Φ?

Here, an FOL-formula Φ

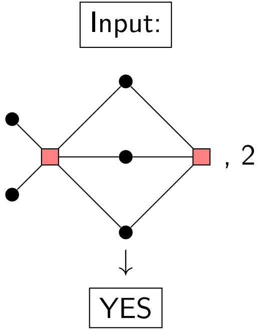
- contains variables that correspond to vertices,
- may consist of adjacency ($\sim$) and identity ($=$) predicates,
- is assumed to be in Prenex Normal Form, i.e. all quantifiers ($\forall, \exists$) are pushed to the left.

Example: Vertex Cover

Let $\Phi := \forall x \forall y\, (x \not\sim y)$.

Vertex Removal towards Φ

Input: A graph G and $k \in \mathbb{N}$.
Question: Does there exist a set F of at most k vertices such that $G \setminus F$ satisfies Φ?



This problem is known as Vertex Cover.
It is NP-complete [Karp '72] and can be decided in time $1.25284^k \cdot n^{\mathcal{O}(1)}$ [Harris, Narayanaswamy '24].

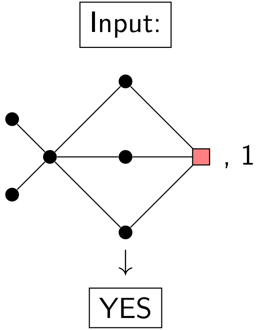
Example: 2-Club

Let $\Phi := \forall x \forall y \exists z \, (x = y) \vee (x \sim y) \vee ((x \sim z) \wedge (z \sim y))$.

Vertex Removal towards Φ

Input: A graph G and $k \in \mathbb{N}$.

Question: Does there exist a set F of at most k vertices such that $G \setminus F$ satisfies Φ?



This problem is known as 2-Club.

It is NP-complete [Balasundaram, Butenko, Trukhanov '05] and cannot be decided in time $(2-\epsilon)^k \cdot n^{\mathcal{O}(1)}$, unless SETH fails [Hartung, Komusiewicz, Nichterlein '15].

Previous Work

Complexity of graph modification problems is very well-studied [Lewis, Yannakakis '80; Crespelle, Drange, Fomin, Golovach '23], but rarely from the descriptive perspective.

[Fomin, Golovach, Thilikos '20] identify FOL-formulas Φ for which the problems

- Vertex Removal towards Φ,
- Edge Removal towards Φ,
- Edge Completion towards Φ and
- Edge Editing towards Φ

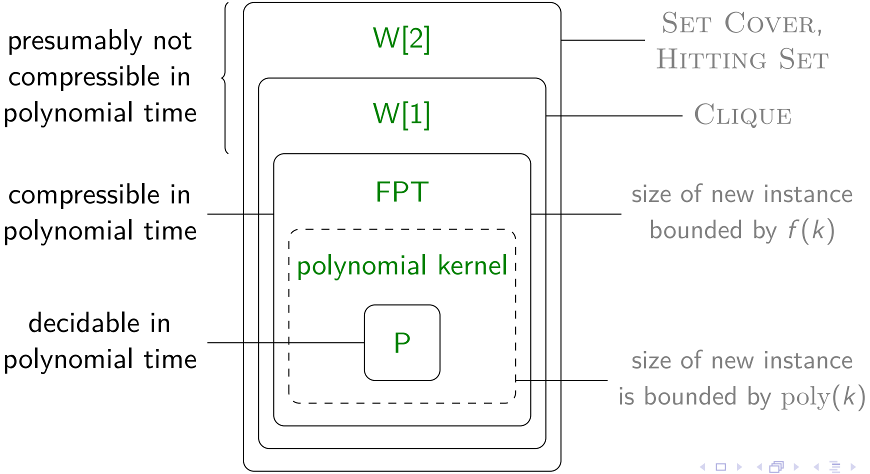
are fixed-parameter tractable (FPT) or presumably not FPT, based on the prefix class of Φ.

This Work

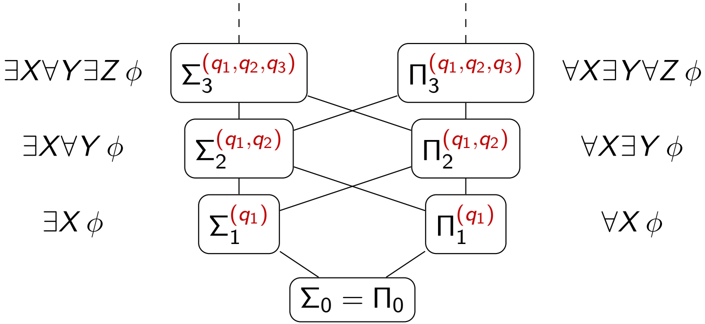
More fine-grained analysis in terms of

- Computational Complexity:
  distinguish various classes of (parameterized) complexity.
- Descriptive Complexity:
  define new prefix classes based on the number of variables within quantifier blocks.

This Work: Classes of Parameterized Complexity



This Work: ‘Fine-Grained’ Arithmetical hierarchy



where ϕ is a quantifier-free FOL-formula and
$|X| = q_1, |Y| = q_2, |Z| = q_3$ with $q_1, q_2, q_3 \in \mathbb{N}^+$.

Example: If $\Phi := \forall x \forall y \exists z\, (x = y) \vee (x \sim y) \vee ((x \sim z) \wedge (y \sim z))$, then $\Phi \in \Pi_2^{(2,1)}$.

Results — Vertex Removal towards Φ

		Parameterized Complexity of Vertex Removal towards Φ parameterized by deletion budget k			
		Polynomial-time algorithm	Polynomial Kernel no $2^{o(n+m)}$ algorithm, unless ETH fails	FPT no $2^{o(n+m)}$ algorithm, unless ETH fails	W[2]-hard no $n^{o(k)}\|G\|^{\mathcal{O}(1)}$ algorithm, unless W[1] = FPT
#quantifier blocks in Φ	1	Σ_1, $\Pi_1^{(1)}$	$\Pi_1^{(\geq 2)}$		
	2	$\Sigma_2^{(\geq 1,1)}$, $\Pi_2^{(1,\geq 1)}$		$\Sigma_2^{(\geq 1,\geq 2)}$, $\Pi_2^{(\geq 2,\geq 1)}$	
	3	$\Sigma_3^{(\geq 1,1,\geq 1)}$		$\Sigma_3^{(\geq 1,\geq 2,\geq 1)}$	Π_3
	≥ 4				all

Results — VERTEX REMOVAL TOWARDS Φ

#quantifier blocks in Φ	Polynomial-time algorithm	Polynomial Kernel no $2^{o(n+m)}$ algorithm, unless ETH fails	FPT no $2^{o(n+m)}$ algorithm, unless ETH fails	W[2]-hard no $n^{o(k)}\|G\|^{\mathcal{O}(1)}$ algorithm, unless W[1] = FPT
1	Σ_1, $\Pi_1^{(1)}$	$\Pi_1^{(\geq 2)}$		
2	$\Sigma_2^{(\geq 1,1)}$, $\Pi_2^{(1,\geq 1)}$		$\Sigma_2^{(\geq 1,\geq 2)}$, $\Pi_2^{(\geq 2,\geq 1)}$	
3	$\Sigma_3^{(\geq 1,1,\geq 1)}$		$\Sigma_3^{(\geq 1,\geq 2,\geq 1)}$	Π_3
≥ 4				all

Parameterized Complexity of VERTEX REMOVAL TOWARDS Φ parameterized by deletion budget k

Results — Vertex Removal towards Φ

		Parameterized Complexity of Vertex Removal towards Φ parameterized by deletion budget k			
		Polynomial-time algorithm	Polynomial Kernel no $2^{o(n+m)}$ algorithm, unless ETH fails	FPT no $2^{o(n+m)}$ algorithm, unless ETH fails	W[2]-hard no $n^{o(k)}\|G\|^{\mathcal{O}(1)}$ algorithm, unless W[1] = FPT
#quantifier blocks in Φ	1	Σ_1, $\Pi_1^{(1)}$	$\Pi_1^{(\geq 2)}$		
	2	$\Sigma_2^{(\geq 1, 1)}$, $\Pi_2^{(1, \geq 1)}$		$\Sigma_2^{(\geq 1, \geq 2)}$, $\Pi_2^{(\geq 2, \geq 1)}$	
	3	$\Sigma_3^{(\geq 1, 1, \geq 1)}$		$\Sigma_3^{(\geq 1, \geq 2, \geq 1)}$	Π_3
	≥ 4				all

Results — Vertex Removal towards Φ

		Parameterized Complexity of Vertex Removal towards Φ parameterized by deletion budget k			
		Polynomial-time algorithm	Polynomial Kernel no $2^{o(n+m)}$ algorithm, unless ETH fails	FPT no $2^{o(n+m)}$ algorithm, unless ETH fails	W[2]-hard no $n^{o(k)}\|G\|^{\mathcal{O}(1)}$ algorithm, unless W[1] = FPT
#quantifier blocks in Φ	1	Σ_1, $\Pi_1^{(1)}$	$\Pi_1^{(\geq 2)}$		
	2	$\Sigma_2^{(\geq 1, 1)}$, $\Pi_2^{(1, \geq 1)}$		$\Sigma_2^{(\geq 1, \geq 2)}$, $\Pi_2^{(\geq 2, \geq 1)}$	
	3	$\Sigma_3^{(\geq 1, 1, \geq 1)}$		$\Sigma_3^{(\geq 1, \geq 2, \geq 1)}$	Π_3
	≥ 4				all

Results — Vertex Removal towards Φ

		Parameterized Complexity of Vertex Removal towards Φ parameterized by deletion budget k			
		Polynomial-time algorithm	Polynomial Kernel no $2^{o(n+m)}$ algorithm, unless ETH fails	FPT no $2^{o(n+m)}$ algorithm, unless ETH fails	W[2]-hard no $n^{o(k)}\|G\|^{\mathcal{O}(1)}$ algorithm, unless W[1] = FPT
#quantifier blocks in Φ	1	Σ_1, $\Pi_1^{(1)}$	$\Pi_1^{(\geq 2)}$		
	2	$\Sigma_2^{(\geq 1, 1)}$, $\Pi_2^{(1, \geq 1)}$		$\Sigma_2^{(\geq 1, \geq 2)}$, $\Pi_2^{(\geq 2, \geq 1)}$	
	3	$\Sigma_3^{(\geq 1, 1, \geq 1)}$		$\Sigma_3^{(\geq 1, \geq 2, \geq 1)}$	Π_3
	≥ 4				all

Results — Vertex Removal towards Φ

		Parameterized Complexity of Vertex Removal towards Φ parameterized by deletion budget k			
		Polynomial-time algorithm	Polynomial Kernel no $2^{o(n+m)}$ algorithm, unless ETH fails	FPT no $2^{o(n+m)}$ algorithm, unless ETH fails	W[2]-hard no $n^{o(k)}\lvert G\rvert^{\mathcal{O}(1)}$ algorithm, unless W[1] = FPT
#quantifier blocks in Φ	1	Σ_1, $\Pi_1^{(1)}$	$\Pi_1^{(\geq 2)}$		
	2	$\Sigma_2^{(\geq 1,1)}$, $\Pi_2^{(1,\geq 1)}$		$\Sigma_2^{(\geq 1,\geq 2)}$, $\Pi_2^{(\geq 2,\geq 1)}$	
	3	$\Sigma_3^{(\geq 1,1,\geq 1)}$		$\Sigma_3^{(\geq 1,\geq 2,\geq 1)}$	Π_3
	≥ 4				all

Results — Edge Removal/Editing towards Φ

	Parameterized Complexity of Edge Removal towards Φ parameterized by deletion budget k			
	Polynomial-time algorithm	Polynomial Kernel	FPT	W[2]-hard
		no $2^{o(n+m)}$ algorithm, unless ETH fails	no $2^{o(n+m)}$ algorithm, unless ETH fails	no $n^{o(k)}\lvert G\rvert^{\mathcal{O}(1)}$ algorithm, unless W[1] = FPT
#quantifier blocks in Φ: 1	Σ_1, $\Pi_1^{(\leq 2)}$	$\Pi_1^{(3)}$	$\Pi_1^{(\geq 4)}$	
2	$\Sigma_2^{(\geq 1,1)}$, $\Pi_2^{(1,1)}$		$\Sigma_2^{(\geq 1,\geq 2)}$	$\Pi_2^{(\geq 1,\geq 2)}$, $\Pi_2^{(\geq 2,\geq 1)}$
3				Σ_3, Π_3
≥ 4				all

Conclusion

- studied interplay between descriptive and computational complexity regarding graph modification problems
- identified features beyond traditional prefix classes that indicate computational complexity like the number of variables per quantifier block

Future Work

- find other structural features of FOL-formulas and study how they influence the complexity of graph modification
- analyse the complexity of graph modification problems when the desired graph property is expressed in higher-order logic (e.g. Monadic Second-Order Logic)