Hereditary First-Order Logic and Extensional ESO

Santiago Guzmán-Pro
joint work with Manuel Bodirsky

Institute of Algebra
TU Dresden

Finite and Algorithmic Model Theory, Les Houches, 2025

European Research Council
Established by the European Commission

ERC Synergy Grant POCOCOP (GA 101071674)

Graph modification problems

Graph modification problems

Edge-modification

For a fixed graph class $\mathcal{C}$ and an input graph G determine the minimum number of edge modifications to G so that it belongs to $\mathcal{C}$.

- Edge-deletion problems (Yanakkakis 1981).
- Hardness of Edge-Modification problems (Alon, Stav 2009).
- Dichotomy Results on the Hardness of H-free Edge Modification Problems (Aravind, Sandeep, Sivadasan 2017).
- Hardness of approximation for H-free edge modification problems (Bliznets, Cygan, Komosa, Pilipczuk 2018)

Graph modification problems

Resilience problems

For a fixed query μ, determine the resilience of μ in an input database $\mathbb{D}$.

- The Complexity of Resilience and Responsibility for Self-Join-Free Conjunctive Queries (Freire, Gatterbauer, Immerman, Meliou 2015)
- New Results for the Complexity of Resilience for Binary with Self-Joins (Freire, Gatterbauer, Immerman, Meliou 2020).
- A Unified Approach for Resilience and Causal Responsibility with Integer Linear Programming (ILP) and LP Relaxations. (Makhija, Gatterbauer 2023).
- The Complexity of Resilience Problems via Valued Constraint Satisfaction (Bodirsky, Semanišinová, Lutz 2024).

Graph modification problems

Vertex-deletion

For a fixed graph class $\mathcal{C}$ and an input graph G determine the minimum k so that $G - U$ belongs to $\mathcal{C}$ for some $|U| \leq k$.

- Node-Deletion NP-Complete Problems (Krishnamoorthy, Deo 1979)
- The node-deletion problem for hereditary properties is NP-complete (Lewis, Yanakkakis 1980).
- Finding odd-cycle transversals (Reed, Smith, Vetta 2004)
- On the Descriptive Complexity of Vertex Deletion Problems (Bannach, Chudigiewitsch, Tantau 2024)

Graph modification problems

Modification to first-order logic

Edge-modification: Given a graph G and a positive integer k test whether it is possible to modify at most k edges so that it satisfies ϕ.

Edge-completion: Given a graph G and a positive integer k test whether it is possible to add at most k edges from G so that it satisfies ϕ.

Edge-deletion: Given a graph G and a positive integer k test whether it is possible to remove at most k edges from G so that it satisfies ϕ.

Vertex-deletion: Given a graph G and a positive integer k test whether it is possible to remove at most k vertices from G so that it satisfies ϕ.

On the parameterized complexity of graph modification to first-order logic properties (Fomin, Golovach, Thilikos 2020)

Graph modification problems

Modification to first-order logic (without parameter k)

Edge-modification: Given a graph G test whether we can modify the edges of G so that it satisfies ϕ.

Edge-completion: Given a graph G test whether we can add edges to G so that it satisfies ϕ.

Edge-deletion: Given a graph G test whether we can remove edges from G so that it satisfies ϕ.

Vertex-deletion: Given a graph G test whether we can remove vertices of G so that it satisfies ϕ.

Graph modification problems

Modification to first-order logic (without parameter k)

Edge-modification: Given a graph G test whether we can modify the edges of G so that it satisfies ϕ.

Edge-completion: Given a graph G test whether we can add edges to G so that it satisfies ϕ.

Edge-deletion: Given a graph G test whether we can remove edges from G so that it satisfies ϕ.

Vertex-deletion: Given a graph G test whether we can remove vertices of G so that it satisfies ϕ.

Graph modification problems

Modification to first-order logic (without parameter k)

Edge-modification: Given a graph G test whether we can modify the edges of G so that it satisfies ϕ.

Edge-completion: Given a graph G test whether we can add edges to G so that it satisfies ϕ.

Edge-deletion: Given a graph G test whether we can remove edges from G so that it satisfies ϕ.

Vertex-deletion: Given a graph G test whether we can remove vertices of G so that it satisfies ϕ.

Hereditary first-order logic

A structure $\mathbb{A}$ *hereditarily satisfies* ϕ if every (induced) substructure $\mathbb{A}'$ of $\mathbb{A}$ satisfies ϕ.

Hereditary first-order logic

A structure $\mathbb{A}$ *hereditarily satisfies* ϕ if every (induced) substructure $\mathbb{A}'$ of $\mathbb{A}$ satisfies ϕ.

Obs. $\mathbb{A}$ hereditarily satisfies ϕ iff $(\mathbb{A}, k)$ is a no-instance of vertex-deletion to $\neg\phi$ whenever $k \leq |A| - 1$.

Hereditary first-order logic

A structure $\mathbb{A}$ *hereditarily satisfies* ϕ if every (induced) substructure $\mathbb{A}'$ of $\mathbb{A}$ satisfies ϕ.

Obs. $\mathbb{A}$ hereditarily satisfies ϕ iff $(\mathbb{A}, k)$ is a no-instance of vertex-deletion to $\neg\phi$ whenever $k \leq |A| - 1$.

Ex. 1 Forests: $\phi :=$ exists a vertex of degree 1.

Ex. 2 Chordal graphs: $\phi :=$ exists a simplicial vertex (Rose, 1970).

Ex. 3 Acyclic digraphs: $\phi :=$ exists a source.

Hereditary first-order logic

A structure $\mathbb{A}$ *hereditarily satisfies* ϕ if every (induced) substructure $\mathbb{A}'$ of $\mathbb{A}$ satisfies ϕ.

Obs. $\mathbb{A}$ hereditarily satisfies ϕ iff $(\mathbb{A}, k)$ is a no-instance of vertex-deletion to $\neg\phi$ whenever $k \leq |A| - 1$.

Ex. 1 Forests: $\phi :=$ exists a vertex of degree 1.

Ex. 2 Chordal graphs: $\phi :=$ exists a simplicial vertex (Rose, 1970).

Ex. 3 Acyclic digraphs: $\phi :=$ exists a source.

Hereditary first-order logic

A structure $\mathbb{A}$ *hereditarily satisfies* ϕ if every (induced) substructure $\mathbb{A}'$ of $\mathbb{A}$ satisfies ϕ.

Obs. $\mathbb{A}$ hereditarily satisfies ϕ iff $(\mathbb{A}, k)$ is a no-instance of vertex-deletion to $\neg\phi$ whenever $k \leq |A| - 1$.

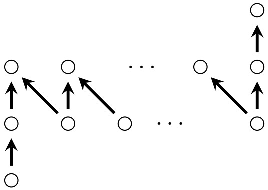
Ex. 1 Forests: $\phi :=$ exists a vertex of degree 1.

Ex. 2 Chordal graphs: $\phi :=$ exists a simplicial vertex (Rose, 1970).

Ex. 3 Acyclic digraphs: $\phi :=$ exists a source.

Hereditary first-order logic

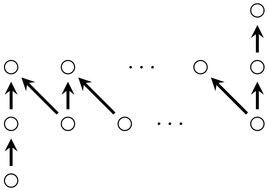
Ex. 4 CSP($\to\to$) is in HerFO.



Non-Ex. CSP(K_2) is not in HerFO.

Hereditary first-order logic

Ex. 4 CSP($\rightarrow\rightarrow$) is in HerFO.



Non-Ex. CSP(K_2) is not in HerFO.

Hereditary first-order logic

Obs. Every problem in HerFO is expressible in UMSO: $\mathbb{A}$ hereditarily satisfies ϕ if and only if

$$\mathbb{A} \models \forall S.\ S \neq A \implies \phi_{\overline{S}}$$

Hereditary first-order logic

Obs. Every problem in HerFO is expressible in UMSO: $\mathbb{A}$ hereditarily satisfies ϕ if and only if

$$\mathbb{A} \models \forall S.\ S \neq A \implies \phi_{\overline{S}}$$

Qst. Is every problem in HerFO solvable in polynomial-time?

Hereditary first-order logic

Obs. Every problem in HerFO is expressible in UMSO: $\mathbb{A}$ hereditarily satisfies ϕ if and only if

$$\mathbb{A} \models \forall S.\, S \neq A \implies \phi_{\overline{S}}$$

Qst. Is every problem in HerFO solvable in polynomial-time?

Lem. If the quantifier prefix of ϕ is of the form $\forall^*\exists\forall^*$ or $\forall^*\exists^*$, then $\mathrm{Her}(\phi)$ is in P and in SNP.

Hereditary first-order logic

Obs. Every problem in HerFO is expressible in UMSO: $\mathbb{A}$ hereditarily satisfies ϕ if and only if

$$\mathbb{A} \models \forall S.\, S \neq A \implies \phi_{\overline{S}}$$

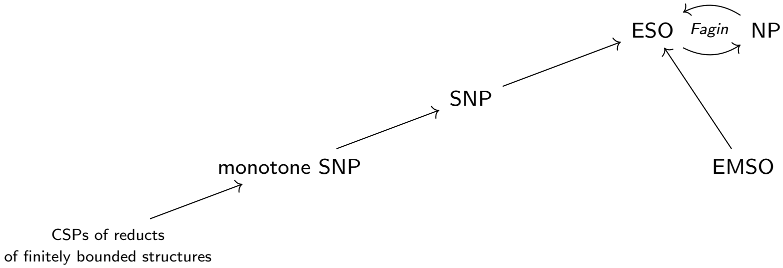
Qst. Is every problem in HerFO solvable in polynomial-time?

Thm. If the quantifier prefix of ϕ is of the form $\forall^*\exists\forall^*$ or $\forall^*\exists^*$, then Her(ϕ) is in P and in SNP. For every other quantifier prefix $\mathcal{Q}$, there is an FO formula ϕ with quantifier prefix $\mathcal{Q}$ such that Her(ϕ) is coNP-complete.

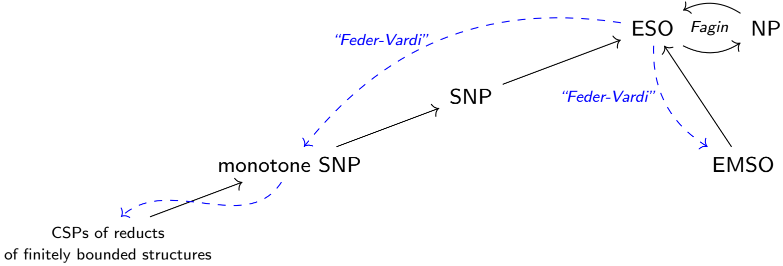
Hereditary first-order logic

“Qst.” What is the computational power of HerFO?

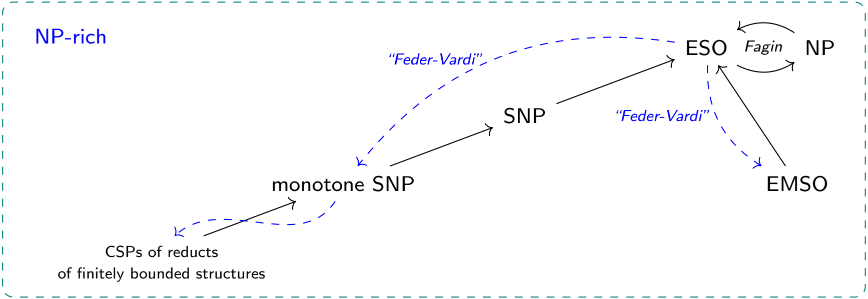
Existential Second-Order Logic



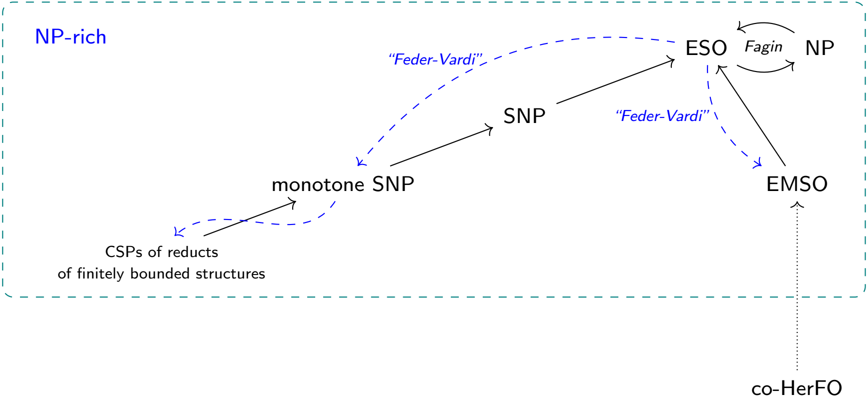
Existential Second-Order Logic



Existential Second-Order Logic



Existential Second-Order Logic

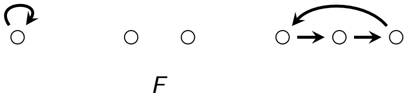


Extensional ESO

Extensional ESO

F-free edge-completion: Given a graph G test whether it is possible to add edges to G so that it becomes F-free.

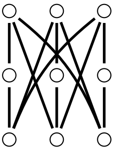
Ex 1. Acyclic digraphs: extend the edge relation to a (strict) linear order.



Extensional ESO

Edge-completion to ϕ: Given a graph G test whether it is possible to add edges so that it satisfies ϕ.

Ex 2. (Pach, 1971) A graph G has circular chromatic number < 3 iff G can be extended to a maximal triangle-free graphs that avoids:



Petersen minus vertex

Extensional ESO

Extensional ESO is the fragment of ESO

$$\exists R_1 \ldots, R_k. \left[\bigwedge_{i \in [n]} \forall \bar{x} R_i'(x) \implies R(\bar{x}) \right] \wedge \phi$$

Extensional ESO

Extensional ESO is the fragment of ESO

$$\exists R_1 \ldots, R_k. \left[\bigwedge_{i \in [n]} \forall \bar{x} R_i'(x) \implies R(\bar{x}) \right] \wedge \phi$$

Non-Ex $\exists W, B. \forall x \; (W(x) \Leftrightarrow \neg B(x)) \wedge \mathrm{Ind}(W) \wedge \mathrm{Ind}(B)$

Ex 3. $\exists W, B. \forall x. \; W'(x) \Rightarrow W(x) \;\wedge\; B'(x) \Rightarrow B(x) \;\wedge\; \phi.$

Extensional ESO

Extensional ESO is the fragment of ESO

$$\exists R_1 \dots, R_k.\ \left[\bigwedge_{i \in [n]} \forall \bar{x} R_i'(x) \implies R(\bar{x}) \right] \ \wedge \ \phi$$

Non-Ex $\exists W, B.\ \forall x\ (W(x) \Leftrightarrow \neg B(x)) \wedge \mathrm{Ind}(W) \wedge \mathrm{Ind}(B)$

Ex 3. Pre-coloured CSP($\mathbb{A}$) is in extensional ESO for every finite $\mathbb{A}$.

Extensional ESO

Extensional ESO is the fragment of ESO

$$\exists R_1 \ldots, R_k.\ \left[\bigwedge_{i \in [n]} \forall \bar{x} R'_i(x) \implies R(\bar{x}) \right] \ \wedge \ \phi$$

Non-Ex $\exists W, B.\ \forall x\ (W(x) \Leftrightarrow \neg B(x)) \wedge \mathrm{Ind}(W) \wedge \mathrm{Ind}(B)$.

Ex 3. Pre-coloured CSP($\mathbb{A}$) is in extensional ESO for every finite $\mathbb{A}$.

Ex 4. A graph G is k-colourable iff it has a $\{K_1 + K_2, K_{k+1}\}$-free extension.

Extensional ESO

Extensional ESO is the fragment of ESO

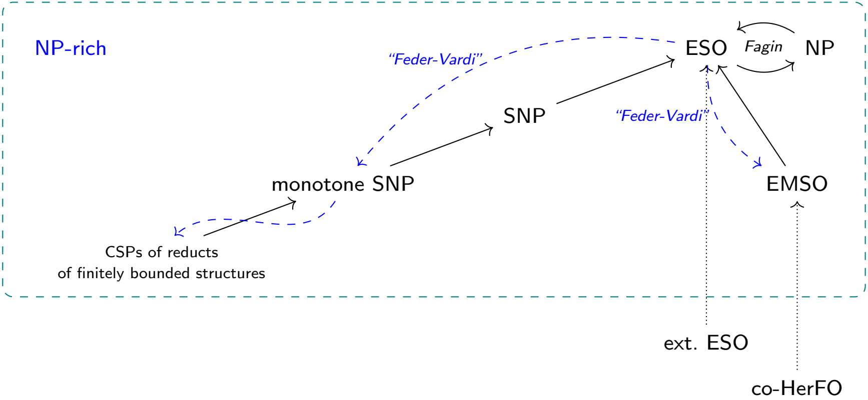
$$\exists R_1 \ldots, R_k. \left[\bigwedge_{i \in [n]} \forall \bar{x} R_i'(x) \implies R(\bar{x}) \right] \wedge \phi$$

Non-Ex $\exists W, B. \forall x \ (W(x) \Leftrightarrow \neg B(x)) \wedge \mathrm{Ind}(W) \wedge \mathrm{Ind}(B)$.

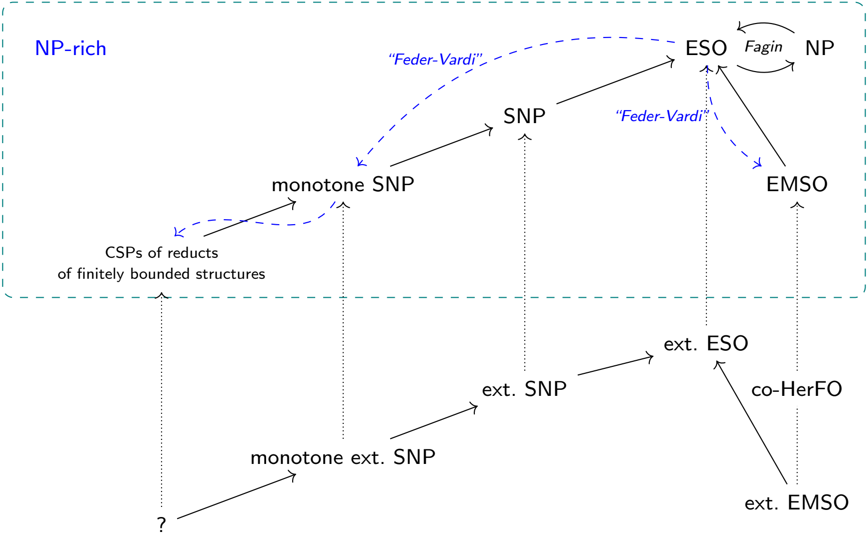
Ex 3. Pre-coloured CSP($\mathbb{A}$) is in extensional ESO for every finite $\mathbb{A}$.

Ex 4. CSP($\mathbb{A}$) and surj-CSP($\mathbb{A}$) are in extensional ESO for every finite $\mathbb{A}$.

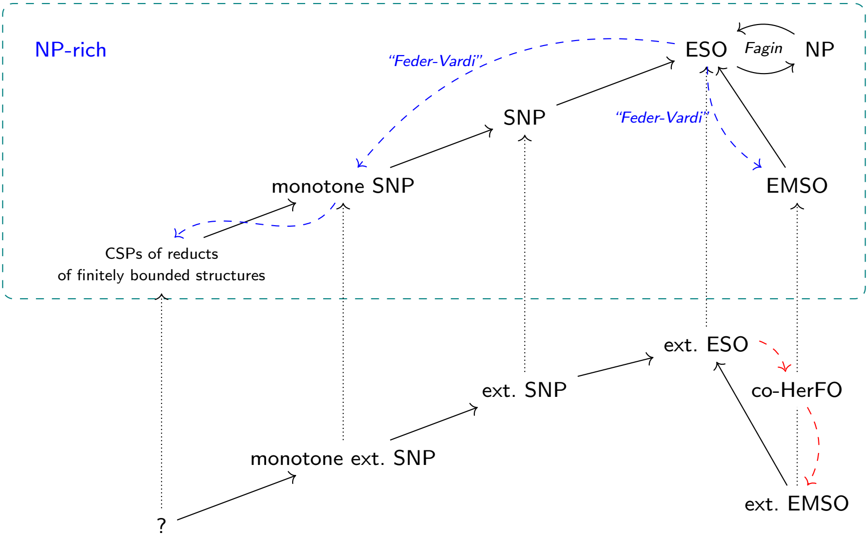
Extensional ESO



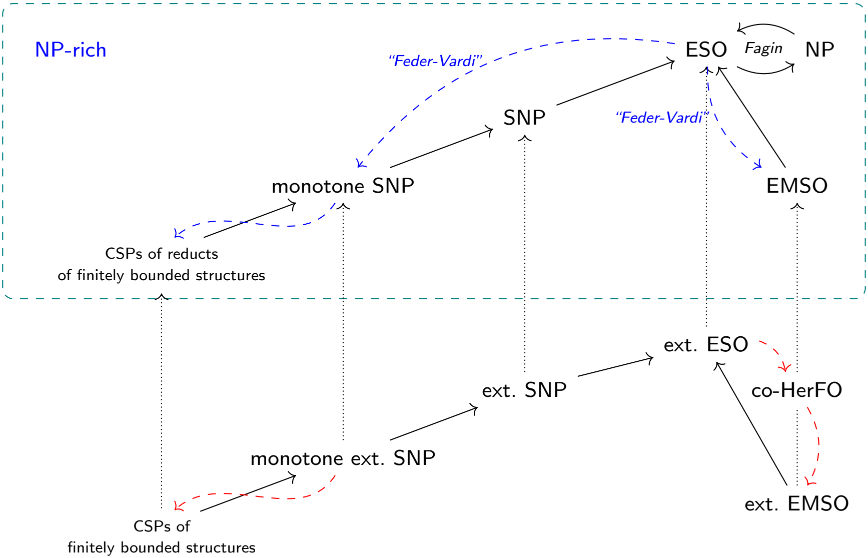
Extensional ESO



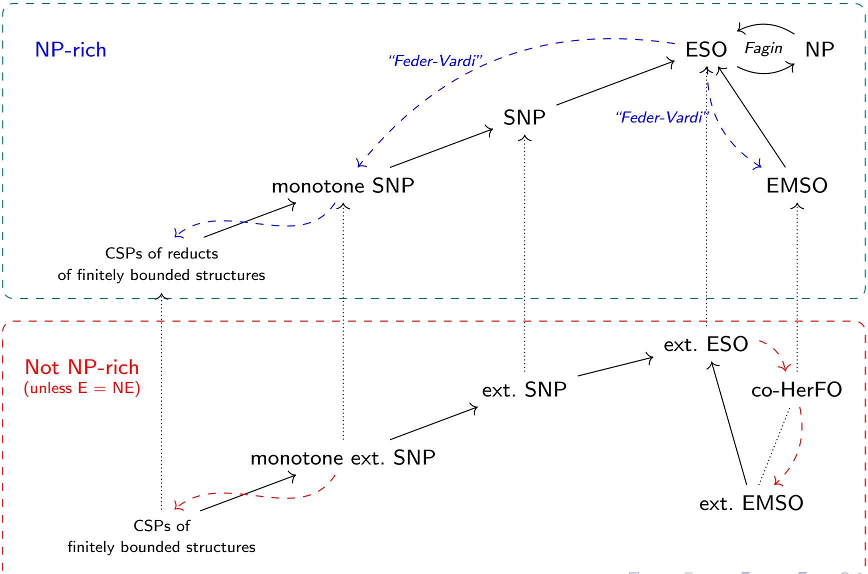
Extensional ESO



Extensional ESO



Extensional ESO



Thank you for your attention!