Representations of permutation groups in Fixed-point logic

Anatole Dahan

University of Cambridge

June 30, 2025

Plan

1 Motivation

2 First approach: definable generating sets

3 Morphism-definability

4 Ordered sets of permutations

Quest of a logic for P, short recap

- Find an enumeration of the P queries (more or less: [Gur88])
- Immerman-Vardi theorem: FP over *ordered structures*
- Over arbitrary structures:
  - FP does not express Parity
  - FP + C $\neq$ P [CFI92]
  - FP + rk $\neq$ P [Lic23]

Quest of a logic for P, short recap

- Find an enumeration of the P queries (more or less: [Gur88])
- Immerman-Vardi theorem: FP over *ordered structures*
- Over arbitrary structures:
  - FP does not express Parity
  - FP + C $\neq$ P [CFI92]
  - FP + rk $\neq$ P [Lic23]
  - CPT ?

Quest of a logic for P, short recap

- Find an enumeration of the P queries (more or less: [Gur88])
- Immerman-Vardi theorem: FP over *ordered structures*
- Over arbitrary structures:
  - FP does not express Parity
  - FP + C $\neq$ P [CFI92]
  - FP + rk $\neq$ P [Lic23]
  - CPT ?
- Over restricted classes of unordered structures:
  - FP + C captures P on any class of structures with bounded tree-width [GM99]
  - More generally, FP + C captures P on any class of structures which excludes a minor [Gro17].

Quest of a logic for P, short recap

- Find an enumeration of the P queries (more or less: [Gur88])
- Immerman-Vardi theorem: FP over *ordered structures*
- Over arbitrary structures:
  - FP does not express Parity
  - FP + C $\neq$ P [CFI92]
  - FP + rk $\neq$ P [Lic23]
  - CPT ?
- Over restricted classes of unordered structures:
  - FP + C captures P on any class of structures with bounded tree-width [GM99]
  - More generally, FP + C captures P on any class of structures which excludes a minor [Gro17].
  - Relies on canonisation

The role of permutation groups in Graph Isomorphism

- Graph Automorphism problem ($\mathsf{GA}_{\mathcal{C}}$): given $\mathcal{G} \in \mathcal{C}$, output a generating set for $\mathrm{Aut}(\mathcal{G}) \leq \mathrm{Sym}(V_{\mathcal{G}})$.
- Note: a polynomial-size such generating set always exists.
- $\mathsf{GI}_{\mathcal{C}} \leq_{\mathsf{P}} \mathsf{GA}_{\mathcal{C}}$ for any union-closed class of (connected) graphs $\mathcal{C}$.
- Based on this insight, many upper bounds have been found for Graph Iso and/or Canonisation for restricted (or not) classes of graphs [Bab79, Luk82, BL83, Bab16]. Most use the CFSG. One exception: Bounded Colour-class Graph Isomorphism.

Primitive operations on permutation groups

All those results rely on some fundamental operations on permutation groups which can be carried out in polynomial time, thanks to the Schreier-Sims algorithm:

- Given $S \subseteq \mathrm{Sym}(D)$, output $|\langle S \rangle|$.
- Given $S \subseteq \mathrm{Sym}(D)$ and $\sigma \in \mathrm{Sym}(D)$, does $\sigma \in \langle S \rangle$?
- Given $S \subseteq \mathrm{Sym}(D)$ and a black-box membership test for $H \leq \langle S \rangle$ such that $|\langle S \rangle|/|H| < |D|^k$ (for some fixed k), output $T \subseteq \mathrm{Sym}(D)$ s.t. $\langle T \rangle = H$.

This last operation gives rise to the definition of *k*-accessible subgroups of $\langle S \rangle$.

How to represent permutation groups in relational structures ?

- Can we bring those methods in reach of isomorphism-invariant formalisms for polynomial-time computation ?

How to represent permutation groups in relational structures ?

- Can we bring those methods in reach of isomorphism-invariant formalisms for polynomial-time computation ?
- One main issue: it is not clear how to represent permutation groups in relational structures.

Plan

1 Motivation

2 First approach: definable generating sets

3 Morphism-definability

4 Ordered sets of permutations

Direct translation of the algorithmic framework

Definition (definable permutation)

A permutation $\sigma \in \mathrm{Sym}(A^k)$ is definable in $\mathfrak{A}$ if there is a formula $\varphi(\vec{s}, \vec{t})$ with $|\vec{s}| = |\vec{t}| = k$ such that

$$\forall \vec{b}, \vec{c} \in A^k, \sigma(\vec{b}) = \vec{c} \iff (\mathfrak{A}, \vec{b}, \vec{c}) \models \varphi$$

Definition (definable permutation group)

A group $G \leq \mathrm{Sym}(A^k)$ is definable in $\mathfrak{A}$ if there is a formula $\varphi(\vec{p}, \vec{s}, \vec{t})$ such that

$$\langle \{ \mathrm{perm}(\varphi(\mathfrak{A}, \vec{a})) \mid \vec{a} \in A^{\vec{p}} \} \rangle = G$$

Low-hanging fruits

- FO + tc can define the orbits of any definable group.
- (Schreier-Sims) Given any structure $\mathfrak{A}$ and any formula φ (in a poly-time model checking logic), $|\langle\varphi\rangle|$ is computable in polynomial time.
- If G and H are $\mathcal{L}$-definable, with $\mathcal{L} \geq \mathrm{FO}$, $\langle G \cup H\rangle$ is $\mathcal{L}$-definable.

Low-hanging fruits

- FO + tc can define the orbits of any definable group.
- (Schreier-Sims) Given any structure $\mathfrak{A}$ and any formula φ (in a poly-time model checking logic), $|\langle\varphi\rangle|$ is computable in polynomial time.
- If G and H are $\mathcal{L}$-definable, with $\mathcal{L} \geq$ FO, $\langle G \cup H\rangle$ is $\mathcal{L}$-definable.
- Corollary: the membership problem reduces (via "Turing FO reduction") to group-order computation.

Low-hanging fruits

- FO + tc can define the orbits of any definable group.
- (Schreier-Sims) Given any structure $\mathfrak{A}$ and any formula φ (in a poly-time model checking logic), $|\langle\varphi\rangle|$ is computable in polynomial time.
- If G and H are $\mathcal{L}$-definable, with $\mathcal{L} \geq$ FO, $\langle G \cup H\rangle$ is $\mathcal{L}$-definable.
- Corollary: the membership problem reduces (via "Turing FO reduction") to group-order computation.
- Turing FO reduction: What if we add an operator to FP that computes the order of a group ? ord

Spoiler

Theorem (D.25)

$\mathrm{FP} + \mathrm{rk} < \mathrm{FP} + \mathrm{ord}$.

Limits

In an isomorphism-invariant context, this representation is not *complete*, i.e. there are groups which admit no small, isomorphism-invariant, generating set:

$$\mathfrak{A}_n := K_{n,n} \quad \text{and} \quad G_n := \mathrm{Aut}(\mathfrak{A}_n)$$

Limits

In an isomorphism-invariant context, this representation is not *complete*, i.e. there are groups which admit no small, isomorphism-invariant, generating set:

$$\mathfrak{A}_n := K_{n,n} \quad \text{and} \quad G_n := \mathrm{Aut}(\mathfrak{A}_n)$$

Worse, we can find a (sequence of) structure $\mathfrak{A}$ s.t.:

- A group $\mathbf{G}(\mathfrak{A})$ is (FP-)definable from $\mathfrak{A}$
- A subgroup $\mathbf{H}(\mathfrak{A})$ is accessible from $\mathbf{G}(\mathfrak{A})$
- FP defines a witness of the accessibility of $\mathbf{H}(\mathfrak{A})$ in $\mathbf{G}(\mathfrak{A})$.
- Any symmetric generating set of $\mathbf{H}(\mathfrak{A})$ has exponential size.

Idea: find restricted cases where we can leverage structural properties of the groups at hand to represent them differently.

Plan

1 Motivation

2 First approach: definable generating sets

3 Morphism-definability

4 Ordered sets of permutations

Morphism-definability

Definition

Let R be a relation symbol of arity $2k$.
$\varphi_m(R, \vec{x}, \vec{y})$ *defines a morphism* $m : G \to \mathrm{Sym}(A^{|\vec{x}|})$ on $\mathfrak{A}$, where $G \leq \mathrm{Sym}(A^k)$ if, for all $\sigma \in G$,

$$\varphi_m(\mathfrak{A}, R \leftarrow \mathrm{graph}(\sigma)) = \mathrm{graph}(m(\sigma))$$

$H \trianglelefteq G \leq \mathrm{Sym}(A^T)$ is morphism-definable from G in $\mathfrak{A}$ if:

- There is a definable generating set for G in $\mathfrak{A}$
- there is a $\mathcal{L}$-formula φ_m which defines a morphism $m : G \to \mathrm{Sym}(A^{T'})$ on $\mathfrak{A}$ such that $\ker(m) = H$.

Operations on morphism-definable subgroups

- If H is morphism-definable from G in $\mathfrak{A}$, then $\mathrm{FP} + \mathrm{ord}$ defines $|H|$, and defines membership to H.
- If H_1 and H_2 are morphism-definable from G in $\mathfrak{A}$, then so is $H_1 \cap H_2$.

Implications

- The separation of $\mathrm{FP} + \mathrm{rk}$ from P relies on structures with abelian colours. In this case, all the relevant groups are morphism-definable.
- Yields a definable canonisation of those structures (following the algorithm from [BL83])
- Theorem: $\mathrm{FP} + \mathrm{rk} < \mathrm{FP} + \mathrm{ord}$ [D.25].

Plan

1 Motivation

2 First approach: definable generating sets

3 Morphism-definability

4 Ordered sets of permutations

- If $\mathrm{FP} + \mathrm{ord}$ defines an *ordered* generating set of permutations for G on $\mathfrak{A}$, and $H \leq G$ is $\mathrm{FP} + \mathrm{ord}$-definably accessible in G, then $\mathrm{FP} + \mathrm{ord}$ defines a generating set for H.
- if G is abelian, FPC suffices.
- structures with abelian colours are equipped with ordered, abelian groups.
- FPC defines the automorphism groups of the structures with abelian colours (which separate $\mathrm{FP} + \mathrm{rk}$ from P).

Conclusion

Thank you !

References I

- László Babai, *Monte-Carlo algorithms in graph isomorphism testing*, Université de Montréal Technical Report, DMS (1979), no. 79-10.
- ______, *Graph Isomorphism in Quasipolynomial Time*, January 2016.
- László Babai and Eugene M. Luks, *Canonical labeling of graphs*, Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing - STOC '83, ACM Press, 1983, pp. 171–183.

References II

- Jin-Yi Cai, Martin Fürer, and Neil Immerman, *An optimal lower bound on the number of variables for graph identification*, Combinatorica **12** (1992), no. 4, 389–410.
- Martin Grohe and Julian Mariño, *Definability and Descriptive Complexity on Databases of Bounded Tree-Width*, Database Theory — ICDT'99 (Gerhard Goos, Juris Hartmanis, Jan Van Leeuwen, Catriel Beeri, and Peter Buneman, eds.), vol. 1540, Springer Berlin Heidelberg, Berlin, Heidelberg, 1999, pp. 70–82.
- Martin Grohe, *Descriptive Complexity, Canonisation, and Definable Graph Structure Theory*, 1 ed., Cambridge University Press, August 2017.

References III

- Yuri Gurevich, *Logic and the Challenge of Computer Science*, Current Trends in Theoretical Computer Science ed. Egon Boerger (1988).
- Moritz Lichter, *Separating Rank Logic from Polynomial Time*, J. ACM **70** (2023), no. 2, 14:1–14:53.
- Eugene M. Luks, *Isomorphism of graphs of bounded valence can be tested in polynomial time*, Journal of Computer and System Sciences **25** (1982), no. 1, 42–65.