Clones on Finite Sets
Up To Minion Homomorphisms

Manuel Bodirsky, Institut für Algebra, TU Dresden

reporting on joint work with F. Starke, A. Vucaj, D. Zhuk,
and on work of F. Starke and S. Meyer

September 11, 2024

European Research Council
Established by the European Commission

ERC Synergy Grant POCOCOP (GA 101071674).

Outline

1. (Abstract and concrete) Clones and Minions
   - Clone lattices on finite domains
   - Clones on finite sets up to minion homomorphisms
   - Open questions
2. TCS Applications: (finite-domain) CSP, PCSP
   - Primitive positive constructions
   - Connection with minion homomorphisms
   - Open questions
3. Recent news: Minions and finite simple groups (Meyer+Starke'24)
4. Datalog fragments (B.+Starke'24)

Minions

Minors and (Concrete) Minions

A, B: sets. $f: A^k \to B$.

Minors and (Concrete) Minions

A, B: sets. $f\colon A^k \to B$.
$m \in \mathbb{N}$, $\alpha\colon \underbrace{[k]}_{:=\{1,\dots,k\}} \to [m]$.

Minors and (Concrete) Minions

A, B: sets. $f\colon A^k \to B$.

$m \in \mathbb{N}$, $\alpha\colon \underbrace{[k]}_{:=\{1,\dots,k\}} \to [m]$.

Def. f_α defined by $f_\alpha(x_1,\dots,x_m) := f(x_{\alpha(1)},\dots,x_{\alpha(k)})$. 'minor' of f.

Minors and (Concrete) Minions

A, B: sets. $f\colon A^k \to B$.
$m \in \mathbb{N}$, $\alpha\colon \underbrace{[k]}_{:=\{1,\dots,k\}} \to [m]$.

Def. f_α defined by $f_\alpha(x_1,\dots,x_m) := f(x_{\alpha(1)},\dots,x_{\alpha(k)})$. 'minor' of f.
Obs. if $\alpha\colon [k] \to [m]$, $\beta\colon [m] \to [n]$, then

$$(f_\alpha)_\beta = f_{\beta\circ\alpha}$$

Minors and (Concrete) Minions

A, B: sets. $f \colon A^k \to B$.
$m \in \mathbb{N}$, $\alpha \colon \underbrace{[k]}_{:=\{1,\dots,k\}} \to [m]$.

Def. f_α defined by $f_\alpha(x_1,\dots,x_m) := f(x_{\alpha(1)},\dots,x_{\alpha(k)})$. 'minor' of f.
Obs. if $\alpha \colon [k] \to [m]$, $\beta \colon [m] \to [n]$, then

$$(f_\alpha)_\beta = f_{\beta \circ \alpha}$$

Def. A function minion $\mathcal{M}$ is a subset of $\bigcup_{k \geq 1} B^{A^k}$
which is closed under taking minors. Aka concrete minion (on (A, B)).

Minors and (Concrete) Minions

A, B: sets. $f\colon A^k \to B$.
$m \in \mathbb{N}$, $\alpha\colon \underbrace{[k]}_{:=\{1,\dots,k\}} \to [m]$.

Def. f_α defined by $f_\alpha(x_1,\dots,x_m) := f(x_{\alpha(1)},\dots,x_{\alpha(k)})$. 'minor' of f.
Obs. if $\alpha\colon [k] \to [m]$, $\beta\colon [m] \to [n]$, then

$$(f_\alpha)_\beta = f_{\beta \circ \alpha}$$

Def. A function minion $\mathcal{M}$ is a subset of $\bigcup_{k \geq 1} B^{A^k}$
which is closed under taking minors. Aka concrete minion (on (A, B)).

Examples.

- $A = B = \{0,1\}$, $\mathcal{M} = \{\pi^k_i \mid i \leq k\} =: Proj$

Minors and (Concrete) Minions

A, B: sets. $f \colon A^k \to B$.
$m \in \mathbb{N}$, $\alpha \colon \underbrace{[k]}_{:=\{1,\dots,k\}} \to [m]$.

Def. f_α defined by $f_\alpha(x_1, \dots, x_m) := f(x_{\alpha(1)}, \dots, x_{\alpha(k)})$. 'minor' of f.
Obs. if $\alpha \colon [k] \to [m]$, $\beta \colon [m] \to [n]$, then

$$(f_\alpha)_\beta = f_{\beta \circ \alpha}$$

Def. A function minion $\mathcal{M}$ is a subset of $\bigcup_{k \geq 1} B^{A^k}$
which is closed under taking minors. Aka concrete minion (on (A, B)).

Examples.

- $A = B = \{0, 1\}$, $\mathcal{M} = \{\pi_i^k \mid i \leq k\} =: Proj$
- For groups (graphs, structures) G and H,
  the set $\mathrm{Pol}(G, H)$ of all homomorphisms from G^k to H, for all k.

Minors and (Concrete) Minions

A, B: sets. $f\colon A^k \to B$.
$m \in \mathbb{N}$, $\alpha\colon \underbrace{[k]}_{:=\{1,\dots,k\}} \to [m]$.

Def. f_α defined by $f_\alpha(x_1,\dots,x_m) := f(x_{\alpha(1)},\dots,x_{\alpha(k)})$. 'minor' of f.
Obs. if $\alpha\colon [k] \to [m]$, $\beta\colon [m] \to [n]$, then

$$(f_\alpha)_\beta = f_{\beta\circ\alpha}$$

Def. A function minion $\mathcal{M}$ is a subset of $\bigcup_{k\geq 1} B^{A^k}$
which is closed under taking minors. Aka concrete minion (on (A, B)).

Examples.

- $A = B = \{0,1\}$, $\mathcal{M} = \{\pi_i^k \mid i \leq k\} =: Proj$
- For groups (graphs, structures) G and H,
  the set $\mathrm{Pol}(G, H)$ of all homomorphisms from G^k to H, for all k.
- For topological spaces S and T,
  the set of all continuous maps from S^k to T.

Minion Homomorphisms, (Abstract) Minions

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}$, $\mathcal{N}$: function minions.

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}$, $\mathcal{N}$: function minions.

$\xi\colon \mathcal{M} \to \mathcal{N}$ is called minion homomorphism if

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}$, $\mathcal{N}$: function minions.

$\xi\colon \mathcal{M} \to \mathcal{N}$ is called minion homomorphism if

1. ξ preserves arities, and

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}$, $\mathcal{N}$: function minions.

$\xi \colon \mathcal{M} \to \mathcal{N}$ is called minion homomorphism if

1. ξ preserves arities, and
2. ξ preserves taking minors, i.e.,
   for every $n, m \in \mathbb{N}$, $\alpha \colon [n] \to [m]$, and $f \in \mathcal{M}$ of arity n:

$$\xi(f_\alpha) = \xi(f)_\alpha.$$

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}$, $\mathcal{N}$: function minions.

$\xi\colon \mathcal{M} \to \mathcal{N}$ is called minion homomorphism if

1. ξ preserves arities, and
2. ξ preserves taking minors, i.e.,
   for every $n, m \in \mathbb{N}$, $\alpha\colon [n] \to [m]$, and $f \in \mathcal{M}$ of arity n:

$$\xi(f_\alpha) = \xi(f)_\alpha.$$

Def. An (abstract) minion is a multi-sorted algebra $\underline{M}$ with sorts $M^{(1)}, M^{(2)}, \dots$

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}$, $\mathcal{N}$: function minions.
$\xi \colon \mathcal{M} \to \mathcal{N}$ is called minion homomorphism if

1. ξ preserves arities, and
2. ξ preserves taking minors, i.e.,
for every $n, m \in \mathbb{N}$, $\alpha \colon [n] \to [m]$, and $f \in \mathcal{M}$ of arity n:

$$\xi(f_\alpha) = \xi(f)_\alpha.$$

Def. An (abstract) minion is a multi-sorted algebra $\underline{M}$ with sorts $M^{(1)}, M^{(2)}, \dots$ and for each $\alpha \colon [n] \to [m]$ the operation $_\alpha \colon M^{(n)} \to M^{(m)}$

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}$, $\mathcal{N}$: function minions.
$\xi\colon \mathcal{M} \to \mathcal{N}$ is called minion homomorphism if

1. ξ preserves arities, and
2. ξ preserves taking minors, i.e.,
   for every $n, m \in \mathbb{N}$, $\alpha\colon [n] \to [m]$, and $f \in \mathcal{M}$ of arity n:

$$\xi(f_\alpha) = \xi(f)_\alpha.$$

Def. An (abstract) minion is a multi-sorted algebra $\underline{M}$ with sorts $M^{(1)}, M^{(2)}, \dots$ and for each $\alpha\colon [n] \to [m]$ the operation $_\alpha\colon M^{(n)} \to M^{(m)}$ such that for every $\alpha\colon [n] \to [m]$ and $\beta\colon [m] \to [k]$ and $f \in M^{(n)}$

$$(f_\alpha)_\beta = f_{\beta \circ \alpha}.$$

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}$, $\mathcal{N}$: function minions.
$\xi\colon \mathcal{M} \to \mathcal{N}$ is called minion homomorphism if

1. ξ preserves arities, and
2. ξ preserves taking minors, i.e.,
   for every $n, m \in \mathbb{N}$, $\alpha\colon [n] \to [m]$, and $f \in \mathcal{M}$ of arity n:

$$\xi(f_\alpha) = \xi(f)_\alpha.$$

Def. An (abstract) minion is a multi-sorted algebra $\underline{M}$ with sorts $M^{(1)}, M^{(2)}, \dots$ and for each $\alpha\colon [n] \to [m]$ the operation $_\alpha\colon M^{(n)} \to M^{(m)}$ such that for every $\alpha\colon [n] \to [m]$ and $\beta\colon [m] \to [k]$ and $f \in M^{(n)}$

$$(f_\alpha)_\beta = f_{\beta\circ\alpha}.$$

Expl. Every function minion is an abstract minion.

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}, \mathcal{N}$: function minions.
$\xi \colon \mathcal{M} \to \mathcal{N}$ is called minion homomorphism if

1. ξ preserves arities, and
2. ξ preserves taking minors, i.e.,
   for every $n, m \in \mathbb{N}$, $\alpha \colon [n] \to [m]$, and $f \in \mathcal{M}$ of arity n:

$$\xi(f_\alpha) = \xi(f)_\alpha.$$

Def. An (abstract) minion is a multi-sorted algebra $\underline{M}$ with sorts $M^{(1)}, M^{(2)}, \dots$ and for each $\alpha \colon [n] \to [m]$ the operation $_\alpha \colon M^{(n)} \to M^{(m)}$ such that for every $\alpha \colon [n] \to [m]$ and $\beta \colon [m] \to [k]$ and $f \in M^{(n)}$

$$(f_\alpha)_\beta = f_{\beta \circ \alpha}.$$

Expl. Every function minion is an abstract minion.

Rem. Every abstract minion arises as a function minion (requires proof).

Minion Homomorphisms, (Abstract) Minions

$\mathcal{M}$, $\mathcal{N}$: function minions.
$\xi \colon \mathcal{M} \to \mathcal{N}$ is called minion homomorphism if

1. ξ preserves arities, and
2. ξ preserves taking minors, i.e.,
   for every $n, m \in \mathbb{N}$, $\alpha \colon [n] \to [m]$, and $f \in \mathcal{M}$ of arity n:

$$\xi(f_\alpha) = \xi(f)_\alpha.$$

Def. An (abstract) minion is a multi-sorted algebra $\underline{M}$ with sorts $M^{(1)}, M^{(2)}, \dots$ and for each $\alpha \colon [n] \to [m]$ the operation $_\alpha \colon M^{(n)} \to M^{(m)}$ such that for every $\alpha \colon [n] \to [m]$ and $\beta \colon [m] \to [k]$ and $f \in M^{(n)}$

$$(f_\alpha)_\beta = f_{\beta \circ \alpha}.$$

Expl. Every function minion is an abstract minion.

Rem. Every abstract minion arises as a function minion (requires proof).

Rem. May be viewed as functors from **FinSet** to **Set**.

Special case: clones

Special case: clones

B: a set.
Def. An (operation) clone $\mathcal{C}$ is a subset of $\bigcup_{k \geq 1} B^{B^k}$
which contains the projections and is closed under composition.

Special case: clones

B: a set.
Def. An (operation) clone $\mathcal{C}$ is a subset of $\bigcup_{k \geq 1} B^{B^k}$
which contains the projections and is closed under composition.

Expl 1. $\text{Pol}(\underline{B}) := \text{Pol}(\underline{B}, \underline{B})$ for some structure $\underline{B}$.

Special case: clones

B: a set.
Def. An (operation) clone $\mathcal{C}$ is a subset of $\bigcup_{k \geq 1} B^{B^k}$
which contains the projections and is closed under composition.

Expl 1. $\text{Pol}(\underline{B}) := \text{Pol}(\underline{B}, \underline{B})$ for some structure $\underline{B}$.
Expl 2. For an algebra $\mathbb{B}$, the set of all term operations of $\mathbb{B}$.

Special case: clones

B: a set.
Def. An (operation) clone $\mathcal{C}$ is a subset of $\bigcup_{k \geq 1} B^{B^k}$
which contains the projections and is closed under composition.

Expl 1. $\mathrm{Pol}(\underline{B}) := \mathrm{Pol}(\underline{B}, \underline{B})$ for some structure $\underline{B}$.
Expl 2. For an algebra $\mathbb{B}$, the set of all term operations of $\mathbb{B}$.

Aka concrete clone (on B).
Analogously as in the case of minions:
clone homomorphisms, (abstract) clones.

B: a set.

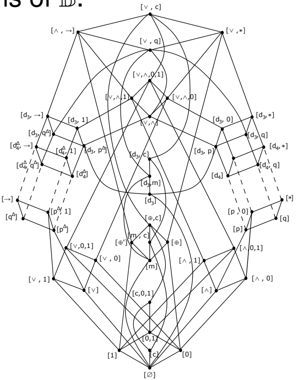
Def. An (operation) clone $\mathcal{C}$ is a subset of $\bigcup_{k \geq 1} B^{B^k}$
which contains the projections and is closed under composition.

Expl 1. $\text{Pol}(\underline{B}) := \text{Pol}(\underline{B}, \underline{B})$ for some structure $\underline{B}$.

Expl 2. For an algebra $\mathbb{B}$, the set of all term operations of $\mathbb{B}$.

Aka concrete clone (on B).
Analogously as in the case of minions:
clone homomorphisms, (abstract) clones.

Expl. Clones over $\{0, 1\}$
with respect to containment:
Post's lattice



Next steps?

- classify clones on larger domain sizes?
- classify function minions on $\{0, 1\}, \{0, 1\}$?

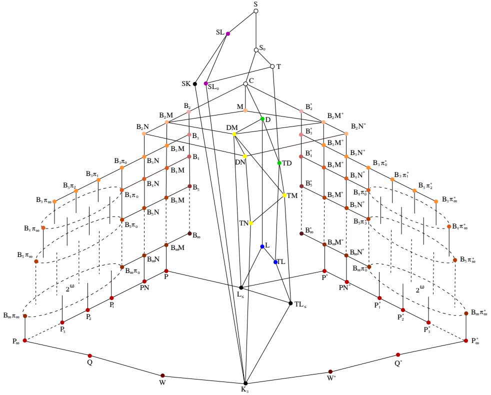
Clone Facts

Fact 1 (Yanov+Muchnik 1959) There are uncountably many operation clones on $\{0, 1, 2\}$.

Clone Facts

Fact 1 (Yanov+Muchnik 1959) There are uncountably many operation clones on $\{0, 1, 2\}$.

Fact 2 (B.+Vucaj+Zhuk 2023) There are uncountably many operation clones on $\{0, 1, 2\}$ up to clone homomorphism equivalence.



Minion Facts

Minion Facts

Fact 3 (Sparks 2019). There are uncountably many function minions contained in $\bigcup_k \{0,1\}^{\{0,1\}^k}$.

Minion Facts

Fact 3 (Sparks 2019). There are uncountably many function minions contained in $\bigcup_k \{0,1\}^{\{0,1\}^k}$.

Fact 4 (Kazda+Moore 2019). Uncountable many even after factoring by minion homomorphism equivalence.

Minion Facts

Fact 3 (Sparks 2019). There are uncountably many function minions contained in $\bigcup_k \{0,1\}^{\{0,1\}^k}$.

Fact 4 (Kazda+Moore 2019). Uncountable many even after factoring by minion homomorphism equivalence.

Proposal: study clones on finite sets
up to minion homomorphism equivalence.

Clones on Finite Sets Up To Minion Homomorphisms

Clones on Finite Sets Up To Minion Homomorphisms

$\mathcal{C}$, $\mathcal{D}$: clones on finite sets.

Clones on Finite Sets Up To Minion Homomorphisms

$\mathcal{C}$, $\mathcal{D}$: clones on finite sets.
Put $\mathcal{C} \leq \mathcal{D}$ if there exists a minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

Clones on Finite Sets Up To Minion Homomorphisms

$\mathcal{C}$, $\mathcal{D}$: clones on finite sets.
Put $\mathcal{C} \leq \mathcal{D}$ if there exists a minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

Resulting poset: P_{fin}.

Clones on Finite Sets Up To Minion Homomorphisms

$\mathcal{C}$, $\mathcal{D}$: clones on finite sets.
Put $\mathcal{C} \leq \mathcal{D}$ if there exists a minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

Resulting poset: P_{fin}.

Idea: If $\mathcal{C}$ contains a binary symmetric operation f such that

$$f(x, y) \approx f(y, x)$$

Clones on Finite Sets Up To Minion Homomorphisms

$\mathcal{C}$, $\mathcal{D}$: clones on finite sets.
Put $\mathcal{C} \leq \mathcal{D}$ if there exists a minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

Resulting poset: P_{fin}.

Idea: If $\mathcal{C}$ contains a binary symmetric operation f such that

$$f(x, y) \approx f(y, x)$$

and $\mathcal{D}$ does not, then there is no minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

Clones on Finite Sets Up To Minion Homomorphisms

$\mathcal{C}$, $\mathcal{D}$: clones on finite sets.
Put $\mathcal{C} \leq \mathcal{D}$ if there exists a minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

Resulting poset: P_{fin}.

Idea: If $\mathcal{C}$ contains a binary symmetric operation f such that

$$f(x,y) \approx f(y,x)$$

and $\mathcal{D}$ does not, then there is no minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

- Unique smallest element:
  equivalence class of Proj.

Clones on Finite Sets Up To Minion Homomorphisms

$\mathcal{C}$, $\mathcal{D}$: clones on finite sets.
Put $\mathcal{C} \leq \mathcal{D}$ if there exists a minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

Resulting poset: P_{fin}.

Idea: If $\mathcal{C}$ contains a binary symmetric operation f such that

$$f(x,y) \approx f(y,x)$$

and $\mathcal{D}$ does not, then there is no minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

- Unique smallest element:
  equivalence class of Proj.
- Unique largest element: Const,
  Equivalence class of all clones containing a constant operation.

Clones on Finite Sets Up To Minion Homomorphisms

$\mathcal{C}$, $\mathcal{D}$: clones on finite sets.
Put $\mathcal{C} \leq \mathcal{D}$ if there exists a minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

Resulting poset: P_{fin}.

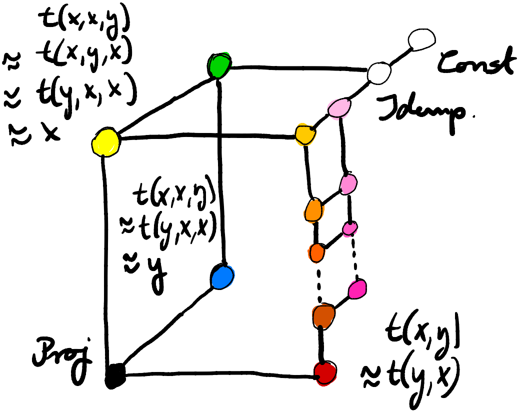
Idea: If $\mathcal{C}$ contains a binary symmetric operation f such that

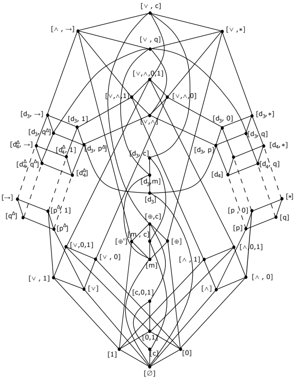
$$f(x, y) \approx f(y, x)$$

and $\mathcal{D}$ does not, then there is no minion homomorphism from $\mathcal{C}$ to $\mathcal{D}$.

- Unique smallest element:
  equivalence class of Proj.
- Unique largest element: Const,
  Equivalence class of all clones containing a constant operation.
- Unique lower cover of Const: Idem,
  Equivalence class of all idempotent clones with at least two elements.
  (f idempotent if $\hat{f}(x) := f(x, \ldots, x) \approx x$)

Clones on $\{0, 1\}$ Up To Minion Homomorphisms





B+Vucaj'2020

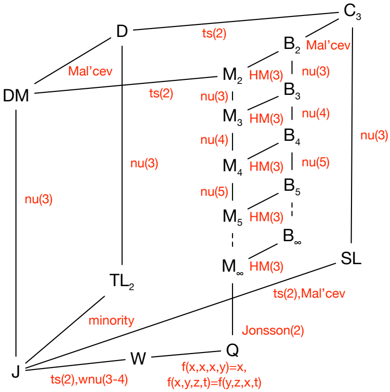
Subclones of Pol$(\{0,1,2\}; C_3)$ up to Minion Homomorphisms

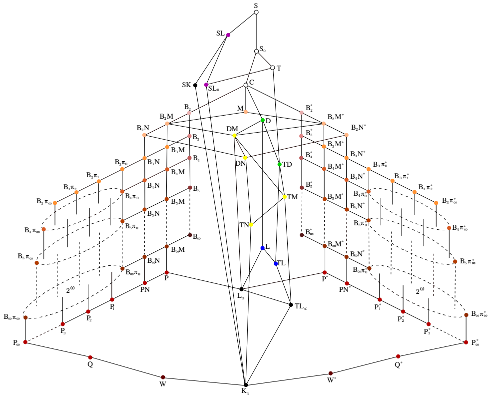
Subclones of $\mathrm{Pol}(\{0,1,2\}; C_3)$ up to Minion Homomorphisms

$C_3 := \{(0,1),(1,2),(2,0)\}$.

Subclones of Pol$(\{0,1,2\}; C_3)$ up to Minion Homomorphisms

$C_3 := \{(0,1),(1,2),(2,0)\}$.





B+Vucaj+Zhuk'2023

Open Problems Part 1

Open Problems Part 1

1. Is $\leq_{\text{con}}$ a lattice?

Open Problems Part 1

1. Is $\leq_{\text{con}}$ a lattice?
2. What is the cardinality of P_{fin}? $\omega \leq |P_{\text{fin}}| \leq 2^{\omega}$

Open Problems Part 1

1. Is $\leq_{\text{con}}$ a lattice?
2. What is the cardinality of P_{fin}? $\omega \leq |P_{\text{fin}}| \leq 2^{\omega}$
3. What is the cardinality of the restriction of P_{fin} to clones on 3 elements?

Open Problems Part 1

1. Is $\leq_{\text{con}}$ a lattice?
2. What is the cardinality of P_{fin}? $\omega \leq |P_{\text{fin}}| \leq 2^{\omega}$
3. What is the cardinality of the restriction of P_{fin} to clones on 3 elements?
4. Are there infinite ascending chains?

Open Problems Part 1

1. Is $\leq_{\text{con}}$ a lattice?
2. What is the cardinality of P_{fin}? $\omega \leq |P_{\text{fin}}| \leq 2^{\omega}$
3. What is the cardinality of the restriction of P_{fin} to clones on 3 elements?
4. Are there infinite ascending chains?
5. What are the maximal elements below Idem?

Open Problems Part 1

1. Is $\leq_{\text{con}}$ a lattice?
2. What is the cardinality of P_{fin}? $\omega \leq |P_{\text{fin}}| \leq 2^{\omega}$
3. What is the cardinality of the restriction of P_{fin} to clones on 3 elements?
4. Are there infinite ascending chains?
5. What are the maximal elements below Idem?
   "Rosenberg theorem for P_{fin}"

Open Problems Part 1

1. Is $\leq_{\text{con}}$ a lattice?
2. What is the cardinality of P_{fin}? $\omega \leq |P_{\text{fin}}| \leq 2^{\omega}$
3. What is the cardinality of the restriction of P_{fin} to clones on 3 elements?
4. Are there infinite ascending chains?
5. What are the maximal elements below Idem?
   "Rosenberg theorem for P_{fin}"

One of them has been solved recently...

Theoretical Computer Science Applications

Homomorphisms

Homomorphisms

$G = (V(G), E(G))$, $H = (V(H), E(H))$.

Homomorphisms

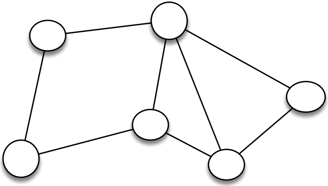
$G = (V(G), E(G))$, $H = (V(H), E(H))$.
$f \colon V(G) \to V(H)$ is called a homomorphism

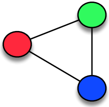
Homomorphisms

$G = (V(G), E(G))$, $H = (V(H), E(H))$.
$f \colon V(G) \to V(H)$ is called a homomorphism
if for every $(u, v) \in E(G)$ have $(f(u), f(v)) \in E(H)$.

Homomorphisms

$G = (V(G), E(G))$, $H = (V(H), E(H))$.
$f\colon V(G) \to V(H)$ is called a homomorphism
if for every $(u, v) \in E(G)$ have $(f(u), f(v)) \in E(H)$.
Write: $G \to H$ if there exists a homomorphism from G to H.

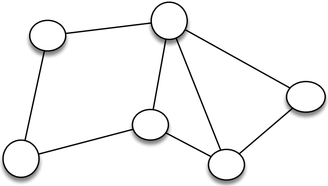


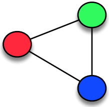


K_3

Homomorphisms

$G = (V(G), E(G))$, $H = (V(H), E(H))$.
$f\colon V(G) \to V(H)$ is called a homomorphism
if for every $(u, v) \in E(G)$ have $(f(u), f(v)) \in E(H)$.
Write: $G \to H$ if there exists a homomorphism from G to H.

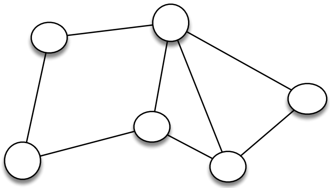


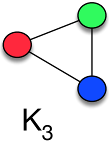


Obs/Def. A graph G is n-colourable
if and only if it has a homomorphism to K_n. (NP-hard for $n \geq 3$).

Homomorphisms

$G = (V(G), E(G))$, $H = (V(H), E(H))$.
$f \colon V(G) \to V(H)$ is called a homomorphism
if for every $(u, v) \in E(G)$ have $(f(u), f(v)) \in E(H)$.
Write: $G \to H$ if there exists a homomorphism from G to H.



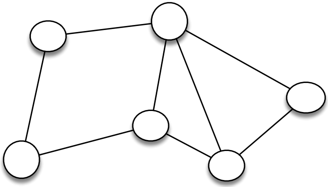


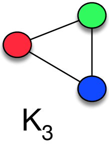
Obs/Def. A graph G is n-colourable
if and only if it has a homomorphism to K_n. (NP-hard for $n \geq 3$).

CSP(H): class of all finite graphs $G \to H$.

Homomorphisms

$G = (V(G), E(G))$, $H = (V(H), E(H))$.
$f\colon V(G) \to V(H)$ is called a homomorphism
if for every $(u, v) \in E(G)$ have $(f(u), f(v)) \in E(H)$.
Write: $G \to H$ if there exists a homomorphism from G to H.





Obs/Def. A graph G is n-colourable
if and only if it has a homomorphism to K_n. (NP-hard for $n \geq 3$).

CSP(H): class of all finite graphs $G \to H$.

Fact: If there is an efficient algorithm to decide CSP(H) (decision!),
there is also an efficient algorithm to compute $f\colon G \to H$ for given G (search!)

Promise CSPs

Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.

Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs

Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Definition PCSP$(\underline{B}, \underline{C})$.

Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Definition PCSP$(\underline{B}, \underline{C})$.
Input: A finite τ-structure $\underline{A}$.

Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Definition PCSP($\underline{B}, \underline{C}$).
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \to \underline{B}$.

Promise CSPs

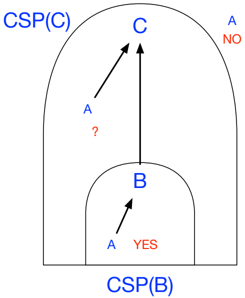
τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Definition PCSP($\underline{B}$, $\underline{C}$).
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \to \underline{B}$.
Answer No if $\underline{A} \not\to \underline{C}$.

Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \rightarrow \underline{C}$.

Definition PCSP$(\underline{B}, \underline{C})$.
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \rightarrow \underline{B}$.
Answer No if $\underline{A} \not\rightarrow \underline{C}$.

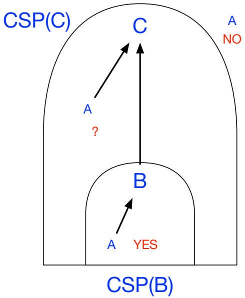


Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Definition PCSP$(\underline{B}, \underline{C})$.
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \to \underline{B}$.
Answer No if $\underline{A} \not\to \underline{C}$.

Obs. PCSP$(\underline{B}, \underline{B})$ = CSP$(\underline{B})$.



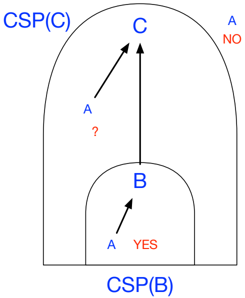
Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Definition PCSP$(\underline{B}, \underline{C})$.
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \to \underline{B}$.
Answer No if $\underline{A} \not\to \underline{C}$.

Obs. PCSP$(\underline{B}, \underline{B})$ = CSP$(\underline{B})$.

Variant: Find a homomorphism to $\underline{C}$



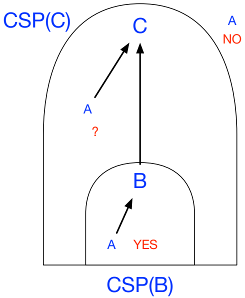
Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Definition PCSP($\underline{B}, \underline{C}$).
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \to \underline{B}$.
Answer No if $\underline{A} \not\to \underline{C}$.

Obs. PCSP($\underline{B}, \underline{B}$) = CSP($\underline{B}$).

Variant: Find a homomorphism to $\underline{C}$
under the promise that there is a homomorphism to $\underline{B}$.



Promise CSPs

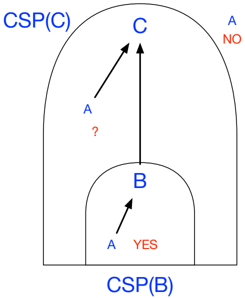
τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Definition PCSP$(\underline{B}, \underline{C})$.
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \to \underline{B}$.
Answer No if $\underline{A} \not\to \underline{C}$.

Obs. PCSP$(\underline{B}, \underline{B})$ = CSP$(\underline{B})$.

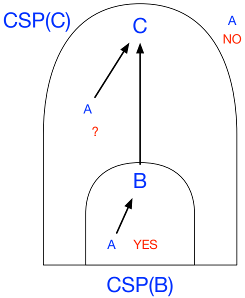
Variant: Find a homomorphism to $\underline{C}$
under the promise that there is a homomorphism to $\underline{B}$.

Might be harder than PCSP$(\underline{B}, \underline{C})$.



Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.



Definition PCSP$(\underline{B}, \underline{C})$.
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \to \underline{B}$.
Answer No if $\underline{A} \not\to \underline{C}$.

Obs. PCSP$(\underline{B}, \underline{B})$ = CSP$(\underline{B})$.

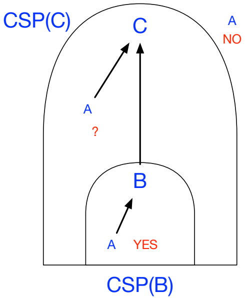
Variant: Find a homomorphism to $\underline{C}$
under the promise that there is a homomorphism to $\underline{B}$.

Might be harder than PCSP$(\underline{B}, \underline{C})$.

Example 1. $\underline{B} = K_3$, $\underline{C} = K_4$. NP-hard (Brakensiek, Guruswami 2016).

Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.



Definition PCSP$(\underline{B}, \underline{C})$.
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \to \underline{B}$.
Answer No if $\underline{A} \not\to \underline{C}$.

Obs. PCSP$(\underline{B}, \underline{B})$ = CSP$(\underline{B})$.

Variant: Find a homomorphism to $\underline{C}$
under the promise that there is a homomorphism to $\underline{B}$.

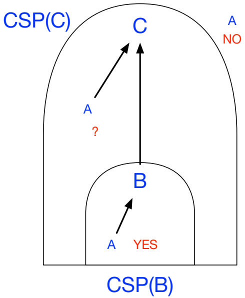
Might be harder than PCSP$(\underline{B}, \underline{C})$.

Example 1. $\underline{B} = K_3$, $\underline{C} = K_4$. NP-hard (Brakensiek, Guruswami 2016).
Example 2. $\underline{B} = K_3$, $\underline{C} = K_5$. NP-hard (Bulin, Krokhin, Opršal 2019).

Promise CSPs

τ: finite set of relation symbols. e.g., $\{E\}$.
$\underline{B}$, $\underline{C}$: τ-structures e.g.: two graphs
Assumption: $\underline{B} \to \underline{C}$.

Definition PCSP$(\underline{B}, \underline{C})$.
Input: A finite τ-structure $\underline{A}$.
Answer Yes if $\underline{A} \to \underline{B}$.
Answer No if $\underline{A} \not\to \underline{C}$.



Obs. PCSP$(\underline{B}, \underline{B})$ = CSP$(\underline{B})$.

Variant: Find a homomorphism to $\underline{C}$
under the promise that there is a homomorphism to $\underline{B}$.

Might be harder than PCSP$(\underline{B}, \underline{C})$.

Example 1. $\underline{B} = K_3$, $\underline{C} = K_4$. NP-hard (Brakensiek, Guruswami 2016).
Example 2. $\underline{B} = K_3$, $\underline{C} = K_5$. NP-hard (Bulin, Krokhin, Opršal 2019).
Example 3. $\underline{B} = K_3$, $\underline{C} = K_6$. Complexity open.

Reductions

Reductions

- If $\underline{B}' \to \underline{B}$ and $\underline{C} \to \underline{C}'$, then $(\underline{B}', \underline{C}')$ is called relaxation of $(\underline{B}, \underline{C})$.

Reductions

- If $\underline{B}' \to \underline{B}$ and $\underline{C} \to \underline{C}'$, then $(\underline{B}', \underline{C}')$ is called relaxation of $(\underline{B}, \underline{C})$.
- $(\underline{B}', \underline{C}')$ is called (n-th) pp-power of $(\underline{B}, \underline{C})$ if
  $B' = B^n$, $C' = C^n$,

Reductions

- If $\underline{B}' \to \underline{B}$ and $\underline{C} \to \underline{C}'$, then $(\underline{B}', \underline{C}')$ is called relaxation of $(\underline{B}, \underline{C})$.
- $(\underline{B}', \underline{C}')$ is called (n-th) pp-power of $(\underline{B}, \underline{C})$ if
  $B' = B^n$, $C' = C^n$, and the relations of $\underline{B}'$ and $\underline{C}'$ are
  $\underbrace{\text{primitively positively definable}}_{\exists, \wedge, =}$ in $\underline{B}$ and $\underline{C}$, resp.

Reductions

- If $\underline{B}' \to \underline{B}$ and $\underline{C} \to \underline{C}'$, then $(\underline{B}', \underline{C}')$ is called relaxation of $(\underline{B}, \underline{C})$.
- $(\underline{B}', \underline{C}')$ is called (n-th) pp-power of $(\underline{B}, \underline{C})$ if $B' = B^n$, $C' = C^n$, and the relations of $\underline{B}'$ and $\underline{C}'$ are $\underbrace{\text{primitively positively definable}}_{\exists, \wedge, =}$ in $\underline{B}$ and $\underline{C}$, resp.
- $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$ if $(\underline{B}', \underline{C}')$ is a relaxation of a pp-power of $(\underline{B}, \underline{C})$.

Reductions

- If $\underline{B}' \to \underline{B}$ and $\underline{C} \to \underline{C}'$, then $(\underline{B}', \underline{C}')$ is called relaxation of $(\underline{B}, \underline{C})$.
- $(\underline{B}', \underline{C}')$ is called (n-th) pp-power of $(\underline{B}, \underline{C})$ if $B' = B^n$, $C' = C^n$, and the relations of $\underline{B}'$ and $\underline{C}'$ are $\underbrace{\text{primitively positively definable}}_{\exists, \wedge, =}$ in $\underline{B}$ and $\underline{C}$, resp.
- $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$ if $(\underline{B}', \underline{C}')$ is a relaxation of a pp-power of $(\underline{B}, \underline{C})$.
- $\underline{B}$ pp-constructs $\underline{B}'$ if $(\underline{B}, \underline{B})$ pp-constructs $(\underline{B}', \underline{B}')$.

Reductions

- If $\underline{B}' \to \underline{B}$ and $\underline{C} \to \underline{C}'$, then $(\underline{B}', \underline{C}')$ is called relaxation of $(\underline{B}, \underline{C})$.
- $(\underline{B}', \underline{C}')$ is called (n-th) pp-power of $(\underline{B}, \underline{C})$ if
  $B' = B^n$, $C' = C^n$, and the relations of $\underline{B}'$ and $\underline{C}'$ are $\underbrace{\text{primitively positively definable}}_{\exists, \wedge, =}$ in $\underline{B}$ and $\underline{C}$, resp.
- $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$
  if $(\underline{B}', \underline{C}')$ is a relaxation of a pp-power of $(\underline{B}, \underline{C})$.
- $\underline{B}$ pp-constructs $\underline{B}'$ if $(\underline{B}, \underline{B})$ pp-constructs $(\underline{B}', \underline{B}')$.

Fact.
If $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$ then PCSP$(\underline{B}', \underline{C}')$ reduces to PCSP$(\underline{B}, \underline{C})$.

Reductions

- If $\underline{B}' \to \underline{B}$ and $\underline{C} \to \underline{C}'$, then $(\underline{B}', \underline{C}')$ is called relaxation of $(\underline{B}, \underline{C})$.
- $(\underline{B}', \underline{C}')$ is called (n-th) pp-power of $(\underline{B}, \underline{C})$ if $B' = B^n$, $C' = C^n$, and the relations of $\underline{B}'$ and $\underline{C}'$ are $\underbrace{\text{primitively positively definable}}_{\exists, \wedge, =}$ in $\underline{B}$ and $\underline{C}$, resp.
- $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$ if $(\underline{B}', \underline{C}')$ is a relaxation of a pp-power of $(\underline{B}, \underline{C})$.
- $\underline{B}$ pp-constructs $\underline{B}'$ if $(\underline{B}, \underline{B})$ pp-constructs $(\underline{B}', \underline{B}')$.

Fact.
If $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$ then PCSP$(\underline{B}', \underline{C}')$ reduces to PCSP$(\underline{B}, \underline{C})$.

Consequence.
If $\underline{B}$ pp-constructs $\underline{B}'$ then CSP$(\underline{B}')$ reduces to CSP$(\underline{B})$.

Primitive Positive Constructions: Example 1

Primitive Positive Constructions: Example 1

$$\vec{C}_3 := \big(\{0,1,2\} \mid \{(0,1),(1,2),(2,0)\}\big)$$

Primitive Positive Constructions: Example 1

$$\vec{C}_3 := \big(\{0,1,2\} \mid \{(0,1),(1,2),(2,0)\}\big)$$

has a pp construction in

$$\vec{C}_6 := \big(\{0,1,2,\dots,5\} \mid \{(x,y) \mid y = x+1 \bmod 6\}\big) :$$

Primitive Positive Constructions: Example 1

$$\vec{C}_3 := \big(\{0,1,2\} \mid \{(0,1),(1,2),(2,0)\}\big)$$

has a pp construction in

$$\vec{C}_6 := \big(\{0,1,2,\dots,5\} \mid \{(x,y) \mid y = x+1 \bmod 6\}\big) :$$

$$\exists u \big(E(x,u) \wedge E(u,y)\big)$$

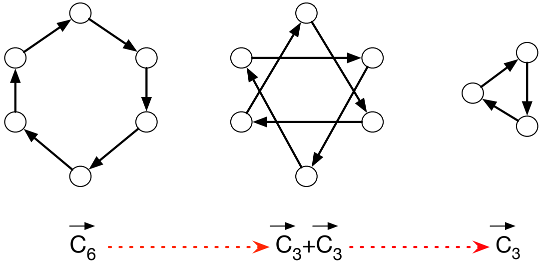
Primitive Positive Constructions: Example 1

$$\vec{C}_3 := \big(\{0,1,2\} \mid \{(0,1),(1,2),(2,0)\}\big)$$

has a pp construction in

$$\vec{C}_6 := \big(\{0,1,2,\dots,5\} \mid \{(x,y) \mid y = x+1 \bmod 6\}\big) :$$

$$\exists u \big(E(x,u) \wedge E(u,y)\big)$$

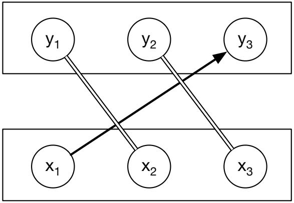


Primitive Positive Constructions: Example 2

$\vec{C}_9$ has a pp construction in $\vec{C}_3$:

Primitive Positive Constructions: Example 2

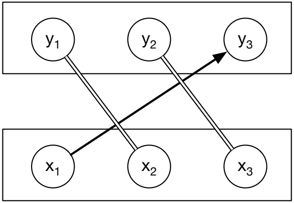
$\vec{C}_9$ has a pp construction in $\vec{C}_3$:



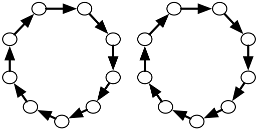
$E(x_1, y_3) \wedge y_1 = x_2 \wedge y_2 = x_3$

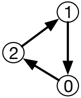
Primitive Positive Constructions: Example 2

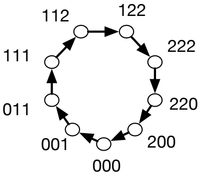
$\vec{C}_9$ has a pp construction in $\vec{C}_3$:

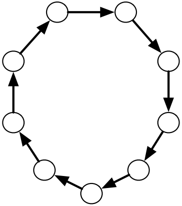


$E(x_1, y_3) \wedge y_1 = x_2 \wedge y_2 = x_3$









$\vec{C}_3 \dashrightarrow \vec{C}_9 + \vec{C}_9 + \vec{C}_9 \dashrightarrow \vec{C}_9$

PCSPs and Minions

PCSPs and Minions

$\mathrm{Pol}(\underline{B}, \underline{C}) := \bigcup_k \underline{B}^k \to \underline{C}$. Function Minion

PCSPs and Minions

$\text{Pol}(\underline{B}, \underline{C}) := \bigcup_k \underline{B}^k \to \underline{C}$. Function Minion
If B, C are finite: every function minion on (B, C) is of this form.

PCSPs and Minions

$\text{Pol}(\underline{B}, \underline{C}) := \bigcup_k \underline{B}^k \to \underline{C}$. Function Minion
If B, C are finite: every function minion on (B, C) is of this form.

Example:

- $\text{Pol}(K_3, K_3)$, $\text{Pol}(K_6, K_6)$: 'small'. (essentially only automorphisms)

PCSPs and Minions

$\text{Pol}(\underline{B}, \underline{C}) := \bigcup_k \underline{B}^k \to \underline{C}$. Function Minion
If B, C are finite: every function minion on (B, C) is of this form.

Example:

- $\text{Pol}(K_3, K_3)$, $\text{Pol}(K_6, K_6)$: 'small'. (essentially only automorphisms)
- $\text{Pol}(K_3, K_6)$: 'large'. (6-colourings of K_3^n, for some n)

PCSPs and Minions

$\text{Pol}(\underline{B}, \underline{C}) := \bigcup_k \underline{B}^k \to \underline{C}$. Function Minion
If B, C are finite: every function minion on (B, C) is of this form.

Example:

- $\text{Pol}(K_3, K_3)$, $\text{Pol}(K_6, K_6)$: 'small'. (essentially only automorphisms)
- $\text{Pol}(K_3, K_6)$: 'large'. (6-colourings of K_3^n, for some n)

Theorem (Bulín, Barto, Krokhin, Opršal 2021).

For finite structures $\underline{B}, \underline{B}', \underline{C}, \underline{C}'$ TFAE:

PCSPs and Minions

$\text{Pol}(\underline{B}, \underline{C}) := \bigcup_k \underline{B}^k \to \underline{C}$. Function Minion
If B, C are finite: every function minion on (B, C) is of this form.

Example:

- $\text{Pol}(K_3, K_3)$, $\text{Pol}(K_6, K_6)$: 'small'. (essentially only automorphisms)
- $\text{Pol}(K_3, K_6)$: 'large'. (6-colourings of K_3^n, for some n)

Theorem (Bulín, Barto, Krokhin, Opršal 2021).

For finite structures $\underline{B}, \underline{B}', \underline{C}, \underline{C}'$ TFAE:

1. $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$

PCSPs and Minions

$\text{Pol}(\underline{B}, \underline{C}) := \bigcup_k \underline{B}^k \to \underline{C}$. Function Minion
If B, C are finite: every function minion on (B, C) is of this form.

Example:

- $\text{Pol}(K_3, K_3)$, $\text{Pol}(K_6, K_6)$: 'small'. (essentially only automorphisms)
- $\text{Pol}(K_3, K_6)$: 'large'. (6-colourings of K_3^n, for some n)

Theorem (Bulín, Barto, Krokhin, Opršal 2021).

For finite structures $\underline{B}, \underline{B}', \underline{C}, \underline{C}'$ TFAE:

1. $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$
2. there is a minion homomorphism from $\text{Pol}(\underline{B}, \underline{C})$ to $\text{Pol}(\underline{B}', \underline{C}')$.

PCSPs and Minions

$\text{Pol}(\underline{B}, \underline{C}) := \bigcup_k \underline{B}^k \to \underline{C}$. Function Minion
If B, C are finite: every function minion on (B, C) is of this form.

Example:

- $\text{Pol}(K_3, K_3)$, $\text{Pol}(K_6, K_6)$: 'small'. (essentially only automorphisms)
- $\text{Pol}(K_3, K_6)$: 'large'. (6-colourings of K_3^n, for some n)

Theorem (Bulín, Barto, Krokhin, Opršal 2021).

For finite structures $\underline{B}, \underline{B}', \underline{C}, \underline{C}'$ TFAE:

1. $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$
2. there is a minion homomorphism from $\text{Pol}(\underline{B}, \underline{C})$ to $\text{Pol}(\underline{B}', \underline{C}')$.

Consequence. For finite $\underline{B}, \underline{C}$, the computational complexity of $\text{PCSP}(\underline{B}, \underline{C})$ only depends on $\text{Pol}(\underline{B}, \underline{C})$, viewed as an abstract minion!

PCSPs and Minions

$\mathrm{Pol}(\underline{B}, \underline{C}) := \bigcup_k \underline{B}^k \to \underline{C}$. Function Minion

If B, C are finite: every function minion on (B, C) is of this form.

Example:

- $\mathrm{Pol}(K_3, K_3)$, $\mathrm{Pol}(K_6, K_6)$: 'small'. (essentially only automorphisms)
- $\mathrm{Pol}(K_3, K_6)$: 'large'. (6-colourings of K_3^n, for some n)

Theorem (Bulín, Barto, Krokhin, Opršal 2021).

For finite structures $\underline{B}, \underline{B}', \underline{C}, \underline{C}'$ TFAE:

1. $(\underline{B}, \underline{C})$ pp-constructs $(\underline{B}', \underline{C}')$
2. there is a minion homomorphism from $\mathrm{Pol}(\underline{B}, \underline{C})$ to $\mathrm{Pol}(\underline{B}', \underline{C}')$.

Consequence. For finite $\underline{B}, \underline{C}$, the computational complexity of $\mathrm{PCSP}(\underline{B}, \underline{C})$ only depends on $\mathrm{Pol}(\underline{B}, \underline{C})$, viewed as an abstract minion!

In fact, even homomorphic equivalence preserves the complexity.

PP-Definitions and Subalgebras

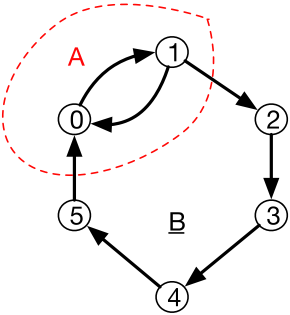
PP-Definitions and Subalgebras

$\underline{B}$: relational structure.
$\text{Pol}(\underline{B})$: polymorphism clone of $\underline{B}$.

PP-Definitions and Subalgebras

$\underline{B}$: relational structure.

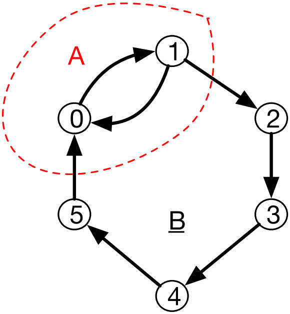
$\text{Pol}(\underline{B})$: polymorphism clone of $\underline{B}$.



PP-Definitions and Subalgebras

$\underline{B}$: relational structure.
$\mathrm{Pol}(\underline{B})$: polymorphism clone of $\underline{B}$.

$A \subseteq B$ is subalgebra of $\mathrm{Pol}(\underline{B})$:iff
$f(a_1, \dots, a_n) \in A$ for all $f \in \mathrm{Pol}(\underline{B})$
of arity n and $a_1, \dots, a_n \in A$.

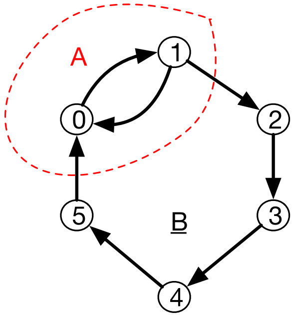


PP-Definitions and Subalgebras

$\underline{B}$: relational structure.
$\mathrm{Pol}(\underline{B})$: polymorphism clone of $\underline{B}$.

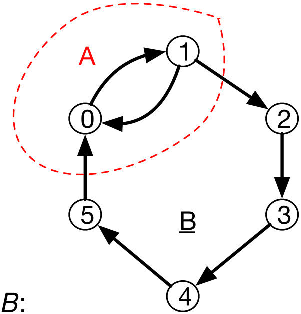
$A \subseteq B$ is subalgebra of $\mathrm{Pol}(\underline{B})$:iff
$f(a_1, \ldots, a_n) \in A$ for all $f \in \mathrm{Pol}(\underline{B})$
of arity n and $a_1, \ldots, a_n \in A$.

Example: $\{0, 1\}$ is a subalgebra of $\underline{B}$.



PP-Definitions and Subalgebras

$\underline{B}$: relational structure.
$\mathrm{Pol}(\underline{B})$: polymorphism clone of $\underline{B}$.



$A \subseteq B$ is subalgebra of $\mathrm{Pol}(\underline{B})$:iff $f(a_1, \ldots, a_n) \in A$ for all $f \in \mathrm{Pol}(\underline{B})$ of arity n and $a_1, \ldots, a_n \in A$.

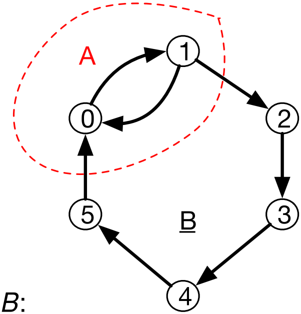
Example: $\{0, 1\}$ is a subalgebra of $\underline{B}$.

$\{0, 1\}$ has the following primitive positive definition in $\underline{B}$:

$$\phi(x) := \exists u \big(E(x, u) \wedge E(u, x) \big)$$

PP-Definitions and Subalgebras

$\underline{B}$: relational structure.
$\mathrm{Pol}(\underline{B})$: polymorphism clone of $\underline{B}$.



$A \subseteq B$ is subalgebra of $\mathrm{Pol}(\underline{B})$:iff
$f(a_1, \dots, a_n) \in A$ for all $f \in \mathrm{Pol}(\underline{B})$
of arity n and $a_1, \dots, a_n \in A$.

Example: $\{0, 1\}$ is a subalgebra of $\underline{B}$.

$\{0, 1\}$ has the following primitive positive definition in $\underline{B}$:

$$\phi(x) := \exists u \big(E(x, u) \wedge E(u, x) \big)$$

Fact for finite B:
$A \subseteq B$ is subalgebra of $\mathrm{Pol}(\underline{B})$ if and only if
A is primitively positively definable in $\underline{B}$.

PP-Definitions and Powers

$\underline{A}$: relational structure.
$\text{Pol}(\underline{A})^d$: clone with domain A^d and the operation

$$\big((a_1^1,\dots,a_1^d),\dots,(a_k^1,\dots,a_k^d)\big) \mapsto \big(f(a_1^1,\dots,a_k^1),\dots,f(a_1^d,\dots,a_k^d)\big)$$

for each $f \in \text{Pol}(\underline{A})$.

PP-Definitions and Powers

$\underline{A}$: relational structure.
$\text{Pol}(\underline{A})^d$: clone with domain A^d and the operation

$$\big((a_1^1,\dots,a_1^d),\dots,(a_k^1,\dots,a_k^d)\big) \mapsto \big(f(a_1^1,\dots,a_k^1),\dots,f(a_1^d,\dots,a_k^d)\big)$$

for each $f \in \text{Pol}(\underline{A})$.

Fact.

$$\text{Pol}(\underline{A})^d = \text{Pol}(\underline{A}^d; E_{1,2},\dots,E_{d-1,d})$$

where $E_{i,j} := \{(s,t) \in A^d \mid s_i = t_j\}$.

Reflections

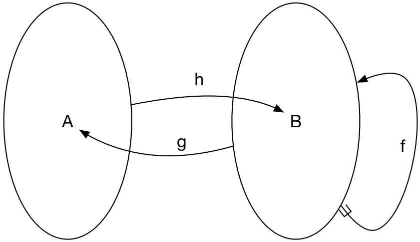
$\underline{A}, \underline{B}$: τ-structures
with homomorphisms
$h\colon \underline{A} \to \underline{B}$ and $g\colon \underline{B} \to \underline{A}$.

Reflections

$\underline{A}, \underline{B}$: τ-structures
with homomorphisms
$h\colon \underline{A} \to \underline{B}$ and $g\colon \underline{B} \to \underline{A}$.
$f\colon B^n \to B$.

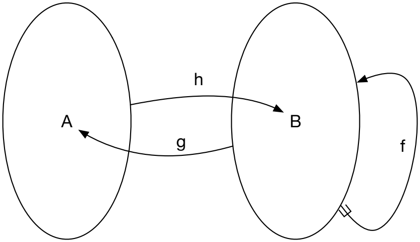
Reflections

$\underline{A}, \underline{B}$: τ-structures
with homomorphisms
$h\colon \underline{A} \to \underline{B}$ and $g\colon \underline{B} \to \underline{A}$.
$f\colon B^n \to B$.



Reflections

$\underline{A}, \underline{B}$: τ-structures
with homomorphisms
$h \colon \underline{A} \to \underline{B}$ and $g \colon \underline{B} \to \underline{A}$.
$f \colon B^n \to B$.

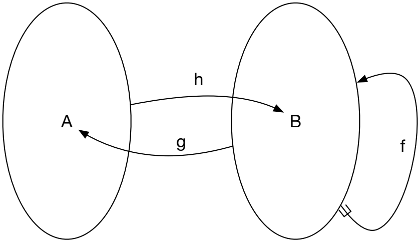


$f^* \colon A^n \to A$ defined by

$$f^*(x_1, \dots, x_n) := g(f(h(x_1), \dots, h(x_n)))$$

Reflections

$\underline{A}, \underline{B}$: τ-structures
with homomorphisms
$h \colon \underline{A} \to \underline{B}$ and $g \colon \underline{B} \to \underline{A}$.
$f \colon B^n \to B$.



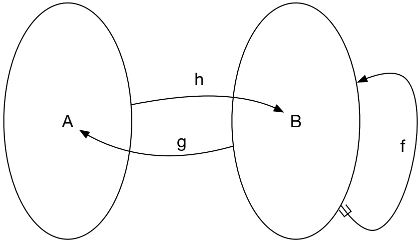
$f^* \colon A^n \to A$ defined by

$$f^*(x_1, \dots, x_n) := g(f(h(x_1), \dots, h(x_n)))$$

Pol$(\underline{A})$ contains

Reflections

$\underline{A}, \underline{B}$: τ-structures
with homomorphisms
$h \colon \underline{A} \to \underline{B}$ and $g \colon \underline{B} \to \underline{A}$.
$f \colon B^n \to B$.



$f^* \colon A^n \to A$ defined by

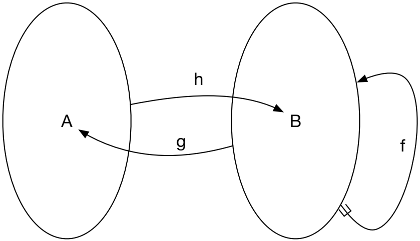
$$f^*(x_1, \ldots, x_n) := g(f(h(x_1), \ldots, h(x_n)))$$

$\text{Pol}(\underline{A})$ contains

$$\text{Refl}(\text{Pol}(\underline{B})) := \{f^* \mid f \in \text{Pol}(\underline{B})\}$$

Reflections

$\underline{A}, \underline{B}$: τ-structures
with homomorphisms
$h \colon \underline{A} \to \underline{B}$ and $g \colon \underline{B} \to \underline{A}$.
$f \colon B^n \to B$.



$f^* \colon A^n \to A$ defined by

$$f^*(x_1, \ldots, x_n) := g(f(h(x_1), \ldots, h(x_n)))$$

$\mathrm{Pol}(\underline{A})$ contains

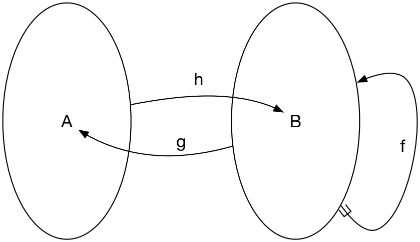
$$\mathrm{Refl}(\mathrm{Pol}(\underline{B})) := \{f^* \mid f \in \mathrm{Pol}(\underline{B})\}$$

Observations.

- $f \mapsto f^*$ is a minor-preserving map from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Reflections

$\underline{A}, \underline{B}$: τ-structures
with homomorphisms
$h \colon \underline{A} \to \underline{B}$ and $g \colon \underline{B} \to \underline{A}$.
$f \colon B^n \to B$.



$f^* \colon A^n \to A$ defined by

$$f^*(x_1, \dots, x_n) := g(f(h(x_1), \dots, h(x_n)))$$

$\mathrm{Pol}(\underline{A})$ contains

$$\mathrm{Refl}(\mathrm{Pol}(\underline{B})) := \{f^* \mid f \in \mathrm{Pol}(\underline{B})\}$$

Observations.

- $f \mapsto f^*$ is a minor-preserving map from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.
- $\mathrm{Refl}(\mathrm{Pol}(\underline{B}))$ is in general not a clone, but still a minion.

Wonderland

Wonderland

Theorem ('Wonderland of reflections',
Barto+Opršal+Pinsker 2015).

Theorem ('Wonderland of reflections', Barto+Opršal+Pinsker 2015).
$\underline{A}$, $\underline{B}$: finite structures. Then:

$\underline{B}$ pp-constructs $\underline{A}$

$\Leftrightarrow$

Wonderland

Theorem ('Wonderland of reflections', Barto+Opršal+Pinsker 2015).
$\underline{A}$, $\underline{B}$: finite structures. Then:

$$\underline{B} \text{ pp-constructs } \underline{A}$$
$$\Leftrightarrow$$
$$\mathrm{Pol}(\underline{A}) \in \mathrm{Exp}\big(\mathrm{Refl}(\mathrm{P}^{\mathrm{fin}}(\mathrm{Pol}(\underline{B})))\big).$$

Wonderland

Theorem ('Wonderland of reflections', Barto+Opršal+Pinsker 2015).
$\underline{A}$, $\underline{B}$: finite structures. Then:

$$\underline{B} \text{ pp-constructs } \underline{A}$$
$$\Leftrightarrow$$
$$\mathrm{Pol}(\underline{A}) \in \mathrm{Exp}\big(\mathrm{Refl}(\mathrm{P}^{\mathrm{fin}}(\mathrm{Pol}(\underline{B})))\big).$$

Note. $\mathrm{Refl}\,\big(\mathrm{P}^{\mathrm{fin}}(\mathrm{Pol}(\underline{B}))\big)$ contains $\mathrm{HSP}^{\mathrm{fin}}(\mathrm{Pol}(\underline{B}))$.

Hight-One Birkhoff

Hight-One Birkhoff

A height-one identity is an identity of the form $s \approx t$

Hight-One Birkhoff

A height-one identity is an identity of the form $s \approx t$ where s and t involve exactly one function symbol.

Hight-One Birkhoff

A height-one identity is an identity of the form $s \approx t$
where s and t involve exactly one function symbol.

Example: $f(x,y) \approx f(y,x)$ (f is symmetric)
Example: $m(y,y,x) \approx m(x,y,y) \approx m(x,x,x)$ (m is quasi-Maltsev)

Hight-One Birkhoff

A height-one identity is an identity of the form $s \approx t$
where s and t involve exactly one function symbol.

Example: $f(x, y) \approx f(y, x)$ (f is symmetric)
Example: $m(y, y, x) \approx m(x, y, y) \approx m(x, x, x)$ (m is quasi-Maltsev)
Non-examples: $f(x, x, x) \approx x$, $f(f(x, y), z) \approx f(x, f(y, z))$.

Hight-One Birkhoff

A height-one identity is an identity of the form $s \approx t$
where s and t involve exactly one function symbol.

Example: $f(x, y) \approx f(y, x)$ (f is symmetric)
Example: $m(y, y, x) \approx m(x, y, y) \approx m(x, x, x)$ (m is quasi-Maltsev)
Non-examples: $f(x, x, x) \approx x$, $f(f(x, y), z) \approx f(x, f(y, z))$.

Theorem (Barto, Opršal, Pinsker 2015). $\underline{A}, \underline{B}$: finite structures. TFAE:

Hight-One Birkhoff

A height-one identity is an identity of the form $s \approx t$
where s and t involve exactly one function symbol.

Example: $f(x, y) \approx f(y, x)$ (f is symmetric)
Example: $m(y, y, x) \approx m(x, y, y) \approx m(x, x, x)$ (m is quasi-Maltsev)
Non-examples: $f(x, x, x) \approx x$, $f(f(x, y), z) \approx f(x, f(y, z))$.

Theorem (Barto, Opršal, Pinsker 2015). $\underline{A}, \underline{B}$: finite structures. TFAE:

1. $\text{Pol}(\underline{A}) \in \text{Exp}(\text{Refl P}(\text{Pol}(\underline{B})))$.

Hight-One Birkhoff

A height-one identity is an identity of the form $s \approx t$
where s and t involve exactly one function symbol.

Example: $f(x, y) \approx f(y, x)$ (f is symmetric)
Example: $m(y, y, x) \approx m(x, y, y) \approx m(x, x, x)$ (m is quasi-Maltsev)
Non-examples: $f(x, x, x) \approx x$, $f(f(x, y), z) \approx f(x, f(y, z))$.

Theorem (Barto, Opršal, Pinsker 2015). $\underline{A}, \underline{B}$: finite structures. TFAE:

1. $\mathrm{Pol}(\underline{A}) \in \mathrm{Exp}(\mathrm{Refl}\ \mathrm{P}(\mathrm{Pol}(\underline{B})))$.
2. There exists a minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Hight-One Birkhoff

A height-one identity is an identity of the form $s \approx t$ where s and t involve exactly one function symbol.

Example: $f(x,y) \approx f(y,x)$ (f is symmetric)
Example: $m(y,y,x) \approx m(x,y,y) \approx m(x,x,x)$ (m is quasi-Maltsev)
Non-examples: $f(x,x,x) \approx x$, $f(f(x,y),z) \approx f(x,f(y,z))$.

Theorem (Barto, Opršal, Pinsker 2015). $\underline{A}, \underline{B}$: finite structures. TFAE:

1. $\text{Pol}(\underline{A}) \in \text{Exp}(\text{Refl P}(\text{Pol}(\underline{B})))$.
2. There exists a minion homomorphism from $\text{Pol}(\underline{B})$ to $\text{Pol}(\underline{A})$.
3. Every set of height-one identities that is satisfied in $\text{Pol}(\underline{B})$ is also satisfied in $\text{Pol}(\underline{A})$.

A height-one identity is an identity of the form $s \approx t$
where s and t involve exactly one function symbol.

Example: $f(x,y) \approx f(y,x)$ (f is symmetric)
Example: $m(y,y,x) \approx m(x,y,y) \approx m(x,x,x)$ (m is quasi-Maltsev)
Non-examples: $f(x,x,x) \approx x$, $f(f(x,y),z) \approx f(x,f(y,z))$.

Theorem (Barto, Opršal, Pinsker 2015). $\underline{A}, \underline{B}$: finite structures. TFAE:

1. $\text{Pol}(\underline{A}) \in \text{Exp}(\text{Refl}\ \text{P}(\text{Pol}(\underline{B})))$.
2. There exists a minion homomorphism from $\text{Pol}(\underline{B})$ to $\text{Pol}(\underline{A})$.
3. Every set of height-one identities that is satisfied in $\text{Pol}(\underline{B})$ is also satisfied in $\text{Pol}(\underline{A})$.

Can be adapted to $\text{Pol}(\underline{B}, \underline{C})$ (Bulín, Barto, Krokhin, Opršal 2021).

Connection with pp Constructions

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Proof.

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\text{Pol}(\underline{B})$ to $\text{Pol}(\underline{A})$.

Proof. $\underline{A}$ has pp construction in $\underline{B}$ iff

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Proof. $\underline{A}$ has pp construction in $\underline{B}$ iff
there exists $M \in \mathrm{Refl}\ \mathrm{P_{fin}}(\mathrm{Pol}(\underline{B}))$ such that $M \subseteq \mathrm{Pol}(\underline{A})$ iff

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Proof. $\underline{A}$ has pp construction in $\underline{B}$ iff
there exists $M \in \mathrm{Refl}\ \mathrm{P}_{\mathrm{fin}}(\mathrm{Pol}(\underline{B}))$ such that $M \subseteq \mathrm{Pol}(\underline{A})$ iff
$\mathrm{Pol}(\underline{B})$ has minion homomorphism to $\mathrm{Pol}(\underline{A})$. $\square$

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Proof. $\underline{A}$ has pp construction in $\underline{B}$ iff
there exists $M \in \mathrm{Refl}\ \mathrm{P_{fin}}(\mathrm{Pol}(\underline{B}))$ such that $M \subseteq \mathrm{Pol}(\underline{A})$ iff
$\mathrm{Pol}(\underline{B})$ has minion homomorphism to $\mathrm{Pol}(\underline{A})$. □

Examples.

- All polymorphisms of K_3 are projections composed with S_3.

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Proof. $\underline{A}$ has pp construction in $\underline{B}$ iff
there exists $M \in \mathrm{Refl}\ \mathrm{P}_{\mathrm{fin}}(\mathrm{Pol}(\underline{B}))$ such that $M \subseteq \mathrm{Pol}(\underline{A})$ iff
$\mathrm{Pol}(\underline{B})$ has minion homomorphism to $\mathrm{Pol}(\underline{A})$. $\square$

Examples.

- All polymorphisms of K_3 are projections composed with S_3.
  In particular: $\hat{f}$ has an inverse for every $f \in \mathrm{Pol}(K_3)$.

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Proof. $\underline{A}$ has pp construction in $\underline{B}$ iff
there exists $M \in \mathrm{Refl}\ \mathrm{P}_{\mathrm{fin}}(\mathrm{Pol}(\underline{B}))$ such that $M \subseteq \mathrm{Pol}(\underline{A})$ iff
$\mathrm{Pol}(\underline{B})$ has minion homomorphism to $\mathrm{Pol}(\underline{A})$. □

Examples.

- All polymorphisms of K_3 are projections composed with S_3.
  In particular: $\hat{f}$ has an inverse for every $f \in \mathrm{Pol}(K_3)$.

$$\xi \colon \mathrm{Pol}(K_3) \to \mathrm{Proj} : \quad f \mapsto (\hat{f})^{-1} \circ f$$

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Proof. $\underline{A}$ has pp construction in $\underline{B}$ iff
there exists $M \in \mathrm{Refl}\ \mathrm{P}_{\mathrm{fin}}(\mathrm{Pol}(\underline{B}))$ such that $M \subseteq \mathrm{Pol}(\underline{A})$ iff
$\mathrm{Pol}(\underline{B})$ has minion homomorphism to $\mathrm{Pol}(\underline{A})$. $\square$

Examples.

- All polymorphisms of K_3 are projections composed with S_3.
  In particular: $\hat{f}$ has an inverse for every $f \in \mathrm{Pol}(K_3)$.

$$\xi \colon \mathrm{Pol}(K_3) \to \mathrm{Proj} : \quad f \mapsto (\hat{f})^{-1} \circ f$$

  Corollary: K_3 pp-constructs all finite structures.

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\text{Pol}(\underline{B})$ to $\text{Pol}(\underline{A})$.

Proof. $\underline{A}$ has pp construction in $\underline{B}$ iff
there exists $M \in \text{Refl P}_{\text{fin}}(\text{Pol}(\underline{B}))$ such that $M \subseteq \text{Pol}(\underline{A})$ iff
$\text{Pol}(\underline{B})$ has minion homomorphism to $\text{Pol}(\underline{A})$. □

Examples.

- All polymorphisms of K_3 are projections composed with S_3.
  In particular: $\hat{f}$ has an inverse for every $f \in \text{Pol}(K_3)$.

  $$\xi\colon \text{Pol}(K_3) \to \text{Proj} : \quad f \mapsto (\hat{f})^{-1} \circ f$$

  Corollary: K_3 pp-constructs all finite structures.
- $\underline{B}$: a finite structure.
  There exists a minion homomorphism from $\text{Pol}(\underline{B})$ to Proj

Connection with pp Constructions

$\underline{A}$ has pp construction in $\underline{B}$ if and only if
there exists minion homomorphism from $\mathrm{Pol}(\underline{B})$ to $\mathrm{Pol}(\underline{A})$.

Proof. $\underline{A}$ has pp construction in $\underline{B}$ iff
there exists $M \in \mathrm{Refl}\ \mathrm{P_{fin}}(\mathrm{Pol}(\underline{B}))$ such that $M \subseteq \mathrm{Pol}(\underline{A})$ iff
$\mathrm{Pol}(\underline{B})$ has minion homomorphism to $\mathrm{Pol}(\underline{A})$. □

Examples.

- All polymorphisms of K_3 are projections composed with S_3.
  In particular: $\hat{f}$ has an inverse for every $f \in \mathrm{Pol}(K_3)$.

  $$\xi\colon \mathrm{Pol}(K_3) \to \mathrm{Proj} : \quad f \mapsto (\hat{f})^{-1} \circ f$$

  Corollary: K_3 pp-constructs all finite structures.
- $\underline{B}$: a finite structure.
  There exists a minion homomorphism from $\mathrm{Pol}(\underline{B})$ to Proj
  if and only if $\underline{B}$ pp-constructs K_3.

Open Problems

Open Problems

Fact 1. Suppose that $\text{Pol}(\underline{B}, \underline{C})$ contains for every $n \in \mathbb{N}$ a function f such that $f = f_\sigma$ for every $\sigma \in S_n$. Then $\text{Pol}(\underline{B}, \underline{C})$ can be solved in polynomial time.

Open Problems

Fact 1. Suppose that Pol($\underline{B}, \underline{C}$) contains for every $n \in \mathbb{N}$ a function f such that $f = f_\sigma$ for every $\sigma \in S_n$. Then Pol($\underline{B}, \underline{C}$) can be solved in polynomial time.

Open Problem 1. For which minions M does $M \to$ Pol($\underline{B}, \underline{C}$) imply polynomial-time tractability for PCSP($\underline{B}, \underline{C}$)?

Open Problems

Fact 1. Suppose that $\text{Pol}(\underline{B}, \underline{C})$ contains for every $n \in \mathbb{N}$ a function f such that $f = f_\sigma$ for every $\sigma \in S_n$. Then $\text{Pol}(\underline{B}, \underline{C})$ can be solved in polynomial time.

Open Problem 1. For which minions M does $M \to \text{Pol}(\underline{B}, \underline{C})$ imply polynomial-time tractability for $\text{PCSP}(\underline{B}, \underline{C})$?

Fact 2. Suppose that $\text{Pol}(\underline{B}, \underline{C}) \to \mathcal{P}_2$. Then $\text{PCSP}(\underline{B}, \underline{C})$ is NP-hard.

Open Problems

Fact 1. Suppose that $\text{Pol}(\underline{B}, \underline{C})$ contains for every $n \in \mathbb{N}$ a function f such that $f = f_\sigma$ for every $\sigma \in S_n$. Then $\text{Pol}(\underline{B}, \underline{C})$ can be solved in polynomial time.

Open Problem 1. For which minions M does $M \to \text{Pol}(\underline{B}, \underline{C})$ imply polynomial-time tractability for $\text{PCSP}(\underline{B}, \underline{C})$?

Fact 2. Suppose that $\text{Pol}(\underline{B}, \underline{C}) \to \mathcal{P}_2$. Then $\text{PCSP}(\underline{B}, \underline{C})$ is NP-hard.

Open Problem 2. For which minions M does $\text{Pol}(\underline{B}, \underline{C}) \to M$ imply NP-hardness for $\text{PCSP}(\underline{B}, \underline{C})$?

Open Problems

Fact 1. Suppose that $\mathrm{Pol}(\underline{B}, \underline{C})$ contains for every $n \in \mathbb{N}$ a function f such that $f = f_\sigma$ for every $\sigma \in S_n$. Then $\mathrm{Pol}(\underline{B}, \underline{C})$ can be solved in polynomial time.

Open Problem 1. For which minions M does $M \to \mathrm{Pol}(\underline{B}, \underline{C})$ imply polynomial-time tractability for $\mathrm{PCSP}(\underline{B}, \underline{C})$?

Fact 2. Suppose that $\mathrm{Pol}(\underline{B}, \underline{C}) \to \mathcal{P}_2$. Then $\mathrm{PCSP}(\underline{B}, \underline{C})$ is NP-hard.

Open Problem 2. For which minions M does $\mathrm{Pol}(\underline{B}, \underline{C}) \to M$ imply NP-hardness for $\mathrm{PCSP}(\underline{B}, \underline{C})$?

Fact 3 (Barto+Kozik 2012). Suppose that $\mathrm{Pol}(\underline{B}) \not\to \mathcal{P}_2$. Then there is a cyclic $f \in \mathrm{Pol}(\underline{B})$ of some arity $n \geq 2$, i.e., f satisfies $f(x_1, \dots, x_n) \approx f(x_2, \dots, x_n, x_1)$.

Open Problems

Fact 1. Suppose that $\mathrm{Pol}(\underline{B}, \underline{C})$ contains for every $n \in \mathbb{N}$ a function f such that $f = f_\sigma$ for every $\sigma \in S_n$. Then $\mathrm{Pol}(\underline{B}, \underline{C})$ can be solved in polynomial time.

Open Problem 1. For which minions M does $M \to \mathrm{Pol}(\underline{B}, \underline{C})$ imply polynomial-time tractability for $\mathrm{PCSP}(\underline{B}, \underline{C})$?

Fact 2. Suppose that $\mathrm{Pol}(\underline{B}, \underline{C}) \to \mathcal{P}_2$. Then $\mathrm{PCSP}(\underline{B}, \underline{C})$ is NP-hard.

Open Problem 2. For which minions M does $\mathrm{Pol}(\underline{B}, \underline{C}) \to M$ imply NP-hardness for $\mathrm{PCSP}(\underline{B}, \underline{C})$?

Fact 3 (Barto+Kozik 2012). Suppose that $\mathrm{Pol}(\underline{B}) \not\to \mathcal{P}_2$. Then there is a cyclic $f \in \mathrm{Pol}(\underline{B})$ of some arity $n \geq 2$, i.e., f satisfies $f(x_1, \ldots, x_n) \approx f(x_2, \ldots, x_n, x_1)$.

Open Problem 3. Do we have similar statements for $\mathrm{Pol}(\underline{B}, \underline{C})$?

Open Problems

Fact 1. Suppose that $\mathrm{Pol}(\underline{B}, \underline{C})$ contains for every $n \in \mathbb{N}$ a function f such that $f = f_\sigma$ for every $\sigma \in S_n$. Then $\mathrm{Pol}(\underline{B}, \underline{C})$ can be solved in polynomial time.

Open Problem 1. For which minions M does $M \to \mathrm{Pol}(\underline{B}, \underline{C})$ imply polynomial-time tractability for $\mathrm{PCSP}(\underline{B}, \underline{C})$?

Fact 2. Suppose that $\mathrm{Pol}(\underline{B}, \underline{C}) \to \mathcal{P}_2$. Then $\mathrm{PCSP}(\underline{B}, \underline{C})$ is NP-hard.

Open Problem 2. For which minions M does $\mathrm{Pol}(\underline{B}, \underline{C}) \to M$ imply NP-hardness for $\mathrm{PCSP}(\underline{B}, \underline{C})$?

Fact 3 (Barto+Kozik 2012). Suppose that $\mathrm{Pol}(\underline{B}) \not\to \mathcal{P}_2$. Then there is a cyclic $f \in \mathrm{Pol}(\underline{B})$ of some arity $n \geq 2$, i.e., f satisfies $f(x_1, \dots, x_n) \approx f(x_2, \dots, x_n, x_1)$.

Open Problem 3. Do we have similar statements for $\mathrm{Pol}(\underline{B}, \underline{C})$?

Fact 4 (Bulatov 2017, Zhuk'2017).
$\mathrm{CSP}(\underline{B})$ is in P or NP-hard for every finite structure $\underline{B}$.

Open Problems

Fact 1. Suppose that $\text{Pol}(\underline{B}, \underline{C})$ contains for every $n \in \mathbb{N}$ a function f such that $f = f_\sigma$ for every $\sigma \in S_n$. Then $\text{Pol}(\underline{B}, \underline{C})$ can be solved in polynomial time.

Open Problem 1. For which minions M does $M \to \text{Pol}(\underline{B}, \underline{C})$ imply polynomial-time tractability for $\text{PCSP}(\underline{B}, \underline{C})$?

Fact 2. Suppose that $\text{Pol}(\underline{B}, \underline{C}) \to \mathcal{P}_2$. Then $\text{PCSP}(\underline{B}, \underline{C})$ is NP-hard.

Open Problem 2. For which minions M does $\text{Pol}(\underline{B}, \underline{C}) \to M$ imply NP-hardness for $\text{PCSP}(\underline{B}, \underline{C})$?

Fact 3 (Barto+Kozik 2012). Suppose that $\text{Pol}(\underline{B}) \not\to \mathcal{P}_2$. Then there is a cyclic $f \in \text{Pol}(\underline{B})$ of some arity $n \geq 2$, i.e., f satisfies $f(x_1, \dots, x_n) \approx f(x_2, \dots, x_n, x_1)$.

Open Problem 3. Do we have similar statements for $\text{Pol}(\underline{B}, \underline{C})$?

Fact 4 (Bulatov 2017, Zhuk'2017).
$\text{CSP}(\underline{B})$ is in P or NP-hard for every finite structure $\underline{B}$.

Open Problem 4. Classify the complexity of $\text{CSP}(\underline{B})$ within P.

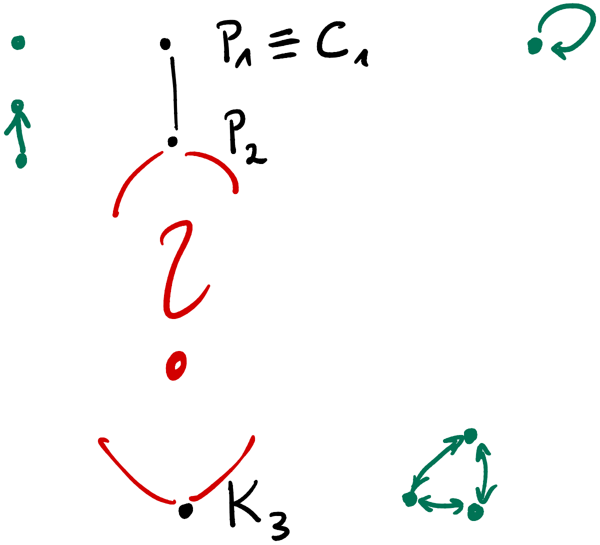
Minions and
Finite Simple Groups

Digraphs

B+Starke'22

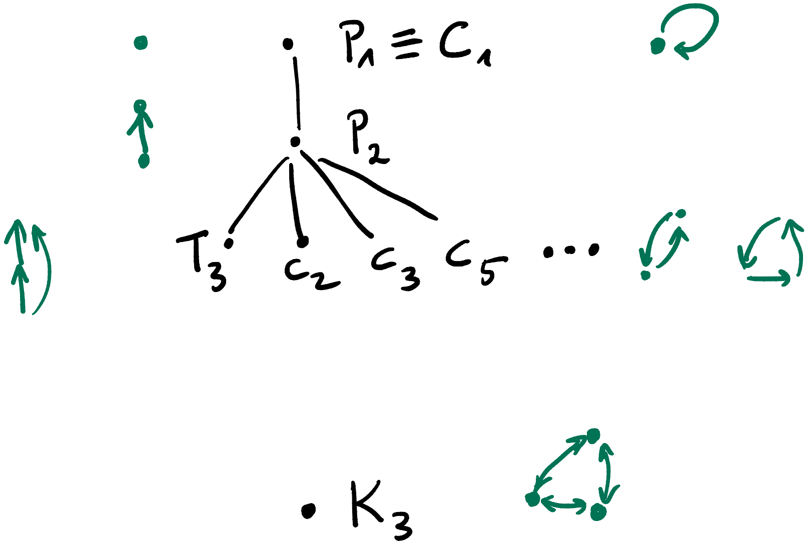
Digraphs

B+Starke'22



Digraphs

B+Starke'22



From Group Actions to Structures

From Group Actions to Structures

G: permutation group on X.

From Group Actions to Structures

G: permutation group on X.

S_G: structure with domain X and for every $g \in G$ the binary relation

$$\{(x, gx) \mid x \in X\}.$$

From Group Actions to Structures

G: permutation group on X.
S_G: structure with domain X and for every $g \in G$ the binary relation

$$\{(x, gx) \mid x \in X\}.$$

$P(G)$: disjoint union of all primitive actions of G (up to isomorphism).

From Group Actions to Structures

G: permutation group on X.

S_G: structure with domain X and for every $g \in G$ the binary relation

$$\{(x, gx) \mid x \in X\}.$$

$P(G)$: disjoint union of all primitive actions of G (up to isomorphism).

Obs.

- If G is finite, then $P(G)$ is finite.

From Group Actions to Structures

G: permutation group on X.
S_G: structure with domain X and for every $g \in G$ the binary relation

$$\{(x, gx) \mid x \in X\}.$$

$P(G)$: disjoint union of all primitive actions of G (up to isomorphism).

Obs.

- If G is finite, then $P(G)$ is finite.
- $S_{P(\mathbb{Z}/p\mathbb{Z})}$ pp-interconstructible with C_p.

From Group Actions to Structures

G: permutation group on X.
S_G: structure with domain X and for every $g \in G$ the binary relation

$$\{(x, gx) \mid x \in X\}.$$

$P(G)$: disjoint union of all primitive actions of G (up to isomorphism).

Obs.

- If G is finite, then $P(G)$ is finite.
- $S_{P(\mathbb{Z}/p\mathbb{Z})}$ pp-interconstructible with C_p.
- S_G has a Maltsev polymorphism:

From Group Actions to Structures

G: permutation group on X.
S_G: structure with domain X and for every $g \in G$ the binary relation

$$\{(x, gx) \mid x \in X\}.$$

$P(G)$: disjoint union of all primitive actions of G (up to isomorphism).

Obs.

- If G is finite, then $P(G)$ is finite.
- $S_{P(\mathbb{Z}/p\mathbb{Z})}$ pp-interconstructible with C_p.
- S_G has a Maltsev polymorphism: the minority which projects which equals the first projection if the three arguments are distinct.

From Group Actions to Structures

G: permutation group on X.
S_G: structure with domain X and for every $g \in G$ the binary relation

$$\{(x, gx) \mid x \in X\}.$$

$P(G)$: disjoint union of all primitive actions of G (up to isomorphism).

Obs.

- If G is finite, then $P(G)$ is finite.
- $S_{P(\mathbb{Z}/p\mathbb{Z})}$ pp-interconstructible with C_p.
- S_G has a Maltsev polymorphism: the minority which projects which equals the first projection if the three arguments are distinct.

Theorem (Meyer+Starke'2024).

Let $\underline{B}$ be a finite structure. Then either

- P_2 pp-constructs $\underline{B}$, or

From Group Actions to Structures

G: permutation group on X.
S_G: structure with domain X and for every $g \in G$ the binary relation

$$\{(x, gx) \mid x \in X\}.$$

$P(G)$: disjoint union of all primitive actions of G (up to isomorphism).

Obs.

- If G is finite, then $P(G)$ is finite.
- $S_{P(\mathbb{Z}/p\mathbb{Z})}$ pp-interconstructible with C_p.
- S_G has a Maltsev polymorphism: the minority which projects which equals the first projection if the three arguments are distinct.

Theorem (Meyer+Starke'2024).

Let $\underline{B}$ be a finite structure. Then either

- P_2 pp-constructs $\underline{B}$, or
- $\underline{B}$ pp-constructs T_3 or $S_{P(G)}$ for some finite simple group G.

From Group Actions to Linear Maltsev Conditions

From Group Actions to Linear Maltsev Conditions

G: permutation group on $[k]$.

From Group Actions to Linear Maltsev Conditions

G: permutation group on $[k]$.
Σ_G: set of all identities of the form

$$f(x_1, \dots, x_k) \approx f(x_{g(1)}, \dots, x_{g(x_k)}), \quad \text{for } g \in G$$

From Group Actions to Linear Maltsev Conditions

G: permutation group on $[k]$.
Σ_G: set of all identities of the form

$$f(x_1, \dots, x_k) \approx f(x_{g(1)}, \dots, x_{g(x_k)}), \quad \text{for } g \in G$$

Examples:

- $\Sigma_{\mathbb{Z}/p\mathbb{Z}}$ expresses the existence of p-cyclic operation.

From Group Actions to Linear Maltsev Conditions

G: permutation group on $[k]$.
Σ_G: set of all identities of the form

$$f(x_1, \ldots, x_k) \approx f(x_{g(1)}, \ldots, x_{g(x_k)}), \quad \text{for } g \in G$$

Examples:

- $\Sigma_{\mathbb{Z}/p\mathbb{Z}}$ expresses the existence of p-cyclic operation.
- Σ_{S_n} expresses the existence of (fully) symmetric operation of arity n.

From Group Actions to Linear Maltsev Conditions

G: permutation group on $[k]$.
Σ_G: set of all identities of the form

$$f(x_1,\dots,x_k) \approx f(x_{g(1)},\dots,x_{g(x_k)}), \quad \text{for } g \in G$$

Examples:

- $\Sigma_{\mathbb{Z}/p\mathbb{Z}}$ expresses the existence of p-cyclic operation.
- Σ_{S_n} expresses the existence of (fully) symmetric operation of arity n.
- $\mathrm{Pol}(C_p) \not\models \Sigma_{\mathbb{Z}/p\mathbb{Z}}$.

From Group Actions to Linear Maltsev Conditions

G: permutation group on $[k]$.
Σ_G: set of all identities of the form

$$f(x_1, \dots, x_k) \approx f(x_{g(1)}, \dots, x_{g(x_k)}), \quad \text{for } g \in G$$

Examples:

- $\Sigma_{\mathbb{Z}/p\mathbb{Z}}$ expresses the existence of p-cyclic operation.
- Σ_{S_n} expresses the existence of (fully) symmetric operation of arity n.
- $\mathrm{Pol}(C_p) \not\models \Sigma_{\mathbb{Z}/p\mathbb{Z}}$.

Lemma: G, H: finite simple groups.

From Group Actions to Linear Maltsev Conditions

G: permutation group on $[k]$.
Σ_G: set of all identities of the form

$$f(x_1,\dots,x_k) \approx f(x_{g(1)},\dots,x_{g(x_k)}), \quad \text{for } g \in G$$

Examples:

- $\Sigma_{\mathbb{Z}/p\mathbb{Z}}$ expresses the existence of p-cyclic operation.
- Σ_{S_n} expresses the existence of (fully) symmetric operation of arity n.
- $\mathrm{Pol}(C_p) \not\models \Sigma_{\mathbb{Z}/p\mathbb{Z}}$.

Lemma: G, H: finite simple groups.
Then $\mathrm{Pol}(S_H) \not\models \Sigma_{P(G)}$ if and only if $G \simeq H$.

From Group Actions to Linear Maltsev Conditions

G: permutation group on $[k]$.
Σ_G: set of all identities of the form

$$f(x_1, \ldots, x_k) \approx f(x_{g(1)}, \ldots, x_{g(x_k)}), \quad \text{for } g \in G$$

Examples:

- $\Sigma_{\mathbb{Z}/p\mathbb{Z}}$ expresses the existence of p-cyclic operation.
- Σ_{S_n} expresses the existence of (fully) symmetric operation of arity n.
- $\mathrm{Pol}(C_p) \not\models \Sigma_{\mathbb{Z}/p\mathbb{Z}}$.

Lemma: G, H: finite simple groups.
Then $\mathrm{Pol}(S_H) \not\models \Sigma_{P(G)}$ if and only if $G \simeq H$.

Theorem (Meyer+Starke'2024).

In P_{fin}, the lower covers of Idem are precisely

$$\{\mathrm{Pol}(T_3)\} \cup \{\mathrm{Pol}(S_{P(G)}) \mid G \text{ finite simple group}\}$$

Linear Maltsev Conditions

Linear Maltsev Conditions

Confirming 2019 speculations
in the communication room of the Institut of Algebra:

Linear Maltsev Conditions

Confirming 2019 speculations
in the communication room of the Institut of Algebra:

Corollary. Let $\mathcal{C}$ be a clone on a finite set with

- a quasi-Maltsev operation, and

Linear Maltsev Conditions

Confirming 2019 speculations
in the communication room of the Institut of Algebra:

Corollary. Let $\mathcal{C}$ be a clone on a finite set with

- a quasi-Maltsev operation, and
- fully symmetric operations of all arities.

Linear Maltsev Conditions

Confirming 2019 speculations
in the communication room of the Institut of Algebra:

Corollary. Let $\mathcal{C}$ be a clone on a finite set with

- a quasi-Maltsev operation, and
- fully symmetric operations of all arities.

Then $\mathcal{C}$ satisfies ‘all linear Maltsev conditions’
(equivalent: all sets of height-one identities that do not imply $f(x) \approx f(y)$)

Linear Maltsev Conditions

Confirming 2019 speculations
in the communication room of the Institut of Algebra:

Corollary. Let $\mathcal{C}$ be a clone on a finite set with

- a quasi-Maltsev operation, and
- fully symmetric operations of all arities.

Then $\mathcal{C}$ satisfies 'all linear Maltsev conditions'
(equivalent: all sets of height-one identities that do not imply $f(x) \approx f(y)$)

Answers question of Vucaj and Zhuk in a strong way
(they asked it for totally symmetric operations of all arities '$\text{ts}(n)\ \forall n$')

Open Problems Revisited

1. Is $\leq_{\text{con}}$ a lattice?
2. What is the cardinality of P_{fin}? $\omega \leq |P_{\text{fin}}| \leq 2^{\omega}$
3. What is the cardinality of the restriction of P_{fin} to clones on 3 elements?
4. Are there infinite ascending chains?
5. What are the maximal elements below Idem? **Solved!**

Datalog Fragments

Datalog

Datalog

$P_3 = (\{0, 1, 2\}; \{(0, 1), (1, 2)\})$.

Datalog

$P_3 = (\{0,1,2\}; \{(0,1),(1,2)\})$.

Example. Datalog program Π for CSP(P_3).

Datalog

$P_3 = (\{0,1,2\};\{(0,1),(1,2)\})$.

Example. Datalog program Π for $\mathrm{CSP}(P_3)$.

$$A(x) \coloneq E(x,y) \qquad\qquad B(x) \coloneq A(y), E(x,y)$$
$$A(y) \coloneq B(x), E(x,y) \qquad\qquad \mathrm{Goal} \coloneq B(y), E(x,y)$$

Datalog

$P_3 = (\{0,1,2\};\{(0,1),(1,2)\})$.

Example. Datalog program Π for CSP(P_3).

$$A(x) :- E(x,y) \qquad B(x) :- A(y), E(x,y)$$
$$A(y) :- B(x), E(x,y) \qquad \text{Goal} :- B(y), E(x,y)$$

E: EDB (extensional database predicate, ‘input’)

Datalog

$P_3 = (\{0,1,2\};\{(0,1),(1,2)\})$.

Example. Datalog program Π for $\mathrm{CSP}(P_3)$.

$$A(x) :- E(x,y) \qquad B(x) :- A(y), E(x,y)$$
$$A(y) :- B(x), E(x,y) \qquad \text{Goal} :- B(y), E(x,y)$$

E: EDB (extensional database predicate, ‘input’)
$A, B,$ Goal: IDBs (intensional database predicates, ‘auxiliary’)

Datalog

$P_3 = (\{0,1,2\};\{(0,1),(1,2)\})$.

Example. Datalog program Π for CSP(P_3).

$$A(x) :- E(x,y) \qquad B(x) :- A(y), E(x,y)$$
$$A(y) :- B(x), E(x,y) \qquad \text{Goal} :- B(y), E(x,y)$$

E: EDB (extensional database predicate, 'input')
$A, B,$ Goal: IDBs (intensional database predicates, 'auxiliary')

This program is:

Datalog

$P_3 = (\{0, 1, 2\}; \{(0, 1), (1, 2)\})$.

Example. Datalog program Π for CSP(P_3).

$$A(x) :- E(x, y) \qquad B(x) :- A(y), E(x, y)$$
$$A(y) :- B(x), E(x, y) \qquad \text{Goal} :- B(y), E(x, y)$$

E: EDB (extensional database predicate, ‘input’)
$A, B,$ Goal: IDBs (intensional database predicates, ‘auxiliary’)

This program is:

- monadic: all IDBs of arity 1

Datalog

$P_3 = (\{0,1,2\};\{(0,1),(1,2)\})$.

Example. Datalog program Π for CSP(P_3).

$$A(x) :- E(x,y) \qquad B(x) :- A(y), E(x,y)$$
$$A(y) :- B(x), E(x,y) \qquad \text{Goal} :- B(y), E(x,y)$$

E: EDB (extensional database predicate, 'input')
$A, B,$ Goal: IDBs (intensional database predicates, 'auxiliary')

This program is:

- monadic: all IDBs of arity 1
- linear: ≤ 1 IDB per rule

Datalog

$P_3 = (\{0,1,2\};\{(0,1),(1,2)\})$.

Example. Datalog program Π for CSP(P_3).

$$A(x) :- E(x,y) \qquad B(x) :- A(y), E(x,y)$$
$$A(y) :- B(x), E(x,y) \qquad \text{Goal} :- B(y), E(x,y)$$

E: EDB (extensional database predicate, 'input')
$A, B,$ Goal: IDBs (intensional database predicates, 'auxiliary')

This program is:

- monadic: all IDBs of arity 1
- linear: ≤ 1 IDB per rule
- arc: ≤ 1 EDB per rule

Datalog

$P_3 = (\{0,1,2\};\{(0,1),(1,2)\})$.

Example. Datalog program Π for CSP(P_3).

$$A(x) :- E(x,y) \qquad B(x) :- A(y), E(x,y)$$
$$A(y) :- B(x), E(x,y) \qquad \text{Goal} :- B(y), E(x,y)$$

E: EDB (extensional database predicate, 'input')
$A, B,$ Goal: IDBs (intensional database predicates, 'auxiliary')

This program is:

- monadic: all IDBs of arity 1
- linear: ≤ 1 IDB per rule
- arc: ≤ 1 EDB per rule
- symmetric: if $(\psi_1 :- \phi, \psi_2) \in \Pi$ for IDBs ψ_1, ψ_2 then $(\psi_2 :- \phi, \psi_1) \in \Pi$.

Classic Result and Conjecture

Theorem (Feder+Vardi 1998). Equivalent:

1. CSP($\underline{B}$) can be solved by monadic arc Datalog

Classic Result and Conjecture

Theorem (Feder+Vardi 1998). Equivalent:

1. CSP($\underline{B}$) can be solved by monadic arc Datalog
2. Pol($\underline{B}$) $\models$ ts(n) $\forall n$

Classic Result and Conjecture

Theorem (Feder+Vardi 1998). Equivalent:

1. CSP($\underline{B}$) can be solved by monadic arc Datalog
2. Pol($\underline{B}$) $\models$ ts(n) $\forall n$
3. $\underline{B}$ has tree duality: $\underline{A} \to \underline{B}$ if all trees that map to $\underline{A}$ also map to $\underline{B}$.

Classic Result and Conjecture

Theorem (Feder+Vardi 1998). Equivalent:

1. CSP($\underline{B}$) can be solved by monadic arc Datalog
2. Pol($\underline{B}$) $\models$ ts(n) $\forall n$
3. $\underline{B}$ has tree duality: $\underline{A} \to \underline{B}$ if all trees that map to $\underline{A}$ also map to $\underline{B}$.

Conjecture (Zadori,Tesson,Dalmau). Equivalent:

1. CSP($\underline{B}$) can be solved by linear Datalog.

Classic Result and Conjecture

Theorem (Feder+Vardi 1998). Equivalent:

1. CSP($\underline{B}$) can be solved by monadic arc Datalog
2. $\mathrm{Pol}(\underline{B}) \models \mathrm{ts}(n)\ \forall n$
3. $\underline{B}$ has tree duality: $\underline{A} \to \underline{B}$ if all trees that map to $\underline{A}$ also map to $\underline{B}$.

Conjecture (Zadori,Tesson,Dalmau). Equivalent:

1. CSP($\underline{B}$) can be solved by linear Datalog.
2. $\mathrm{Pol}(\underline{B}) \models$ Kearnes-Kiss (a specific set of height-one identities)

Classic Result and Conjecture

Theorem (Feder+Vardi 1998). Equivalent:

1. CSP($\underline{B}$) can be solved by monadic arc Datalog
2. Pol($\underline{B}$) $\models$ ts(n) $\forall n$
3. $\underline{B}$ has tree duality: $\underline{A} \to \underline{B}$ if all trees that map to $\underline{A}$ also map to $\underline{B}$.

Conjecture (Zadori,Tesson,Dalmau). Equivalent:

1. CSP($\underline{B}$) can be solved by linear Datalog.
2. Pol($\underline{B}$) $\models$ Kearnes-Kiss (a specific set of height-one identities)
3. $\underline{B}$ does not pp-construct Horn-SAT or $(\mathbb{Z}_p; +, 1)$ for all primes p

Classic Result and Conjecture

Theorem (Feder+Vardi 1998). Equivalent:

1. CSP($\underline{B}$) can be solved by monadic arc Datalog
2. Pol($\underline{B}$) $\models$ ts(n) $\forall n$
3. $\underline{B}$ has tree duality: $\underline{A} \to \underline{B}$ if all trees that map to $\underline{A}$ also map to $\underline{B}$.

Conjecture (Zadori,Tesson,Dalmau). Equivalent:

1. CSP($\underline{B}$) can be solved by linear Datalog.
2. Pol($\underline{B}$) $\models$ Kearnes-Kiss (a specific set of height-one identities)
3. $\underline{B}$ does not pp-construct Horn-SAT or $(\mathbb{Z}_p; +, 1)$ for all primes p
4. $\underline{B}$ has bounded pathwidth duality

Classic Result and Conjecture

Theorem (Feder+Vardi 1998). Equivalent:

1. CSP($\underline{B}$) can be solved by monadic arc Datalog
2. Pol($\underline{B}$) $\models$ ts(n) $\forall n$
3. $\underline{B}$ has tree duality: $\underline{A} \to \underline{B}$ if all trees that map to $\underline{A}$ also map to $\underline{B}$.

Conjecture (Zadori,Tesson,Dalmau). Equivalent:

1. CSP($\underline{B}$) can be solved by linear Datalog.
2. Pol($\underline{B}$) $\models$ Kearnes-Kiss (a specific set of height-one identities)
3. $\underline{B}$ does not pp-construct Horn-SAT or $(\mathbb{Z}_p; +, 1)$ for all primes p
4. $\underline{B}$ has bounded pathwidth duality
5. CSP($\underline{B}$) in NL (non-deterministic logspace)

Another Conjecture, Another Result

Another Conjecture, Another Result

Conjecture (Egri,Dalmau,Larose,Tesson,Zadori). Equivalent:

1. CSP($\underline{B}$) can be solved by symmetric linear Datalog.

Another Conjecture, Another Result

Conjecture (Egri,Dalmau,Larose,Tesson,Zadori). Equivalent:

1. CSP($\underline{B}$) can be solved by symmetric linear Datalog.
2. Pol($\underline{B}$) $\models$ noname (a specific set of height-one identities)

Another Conjecture, Another Result

Conjecture (Egri,Dalmau,Larose,Tesson,Zadori). Equivalent:

1. CSP($\underline{B}$) can be solved by symmetric linear Datalog.
2. Pol($\underline{B}$) $\models$ noname (a specific set of height-one identities)
3. $\underline{B}$ does not pp-construct st-Conn or $(\mathbb{Z}_p; +, 1)$ for all primes p

Another Conjecture, Another Result

Conjecture (Egri,Dalmau,Larose,Tesson,Zadori). Equivalent:

1. CSP($\underline{B}$) can be solved by symmetric linear Datalog.
2. Pol($\underline{B}$) $\models$ noname (a specific set of height-one identities)
3. $\underline{B}$ does not pp-construct st-Conn or $(\mathbb{Z}_p; +, 1)$ for all primes p
4. CSP($\underline{B}$) in L (deterministic logspace)

Another Conjecture, Another Result

Conjecture (Egri,Dalmau,Larose,Tesson,Zadori). Equivalent:

1. CSP($\underline{B}$) can be solved by symmetric linear Datalog.
2. Pol($\underline{B}$) $\models$ noname (a specific set of height-one identities)
3. $\underline{B}$ does not pp-construct st-Conn or $(\mathbb{Z}_p; +, 1)$ for all primes p
4. CSP($\underline{B}$) in L (deterministic logspace)

Theorem (Carvalho+Dalmau+Krokhin 2010). Equivalent:

1. CSP($\underline{B}$) can be solved by linear monadic arc Datalog

Another Conjecture, Another Result

Conjecture (Egri,Dalmau,Larose,Tesson,Zadori). Equivalent:

1. CSP($\underline{B}$) can be solved by symmetric linear Datalog.
2. Pol($\underline{B}$) $\models$ noname (a specific set of height-one identities)
3. $\underline{B}$ does not pp-construct st-Conn or $(\mathbb{Z}_p; +, 1)$ for all primes p
4. CSP($\underline{B}$) in L (deterministic logspace)

Theorem (Carvalho+Dalmau+Krokhin 2010). Equivalent:

1. CSP($\underline{B}$) can be solved by linear monadic arc Datalog
2. Pol($\underline{B}$) $\models$ n-k-abs(n, k) $\forall n$

$$f(x_{11}, \dots, x_{nk}) \approx f(y_{11}, \dots, y_{nk}) \text{ if } \{S_1, \dots, S_n\} = \{T_1, \dots, T_n\},$$
$$S_i = \{x_{i1}, \dots, x_{ik}\}, T_i = \{y_{11}, \dots, y_{ik}\}$$

$$f(S_1, S_2, \dots, S_n) \approx f(S_2, S_2, \dots, S_n) \text{ if } S_2 \subseteq S_1$$

Another Conjecture, Another Result

Conjecture (Egri,Dalmau,Larose,Tesson,Zadori). Equivalent:

1. CSP($\underline{B}$) can be solved by symmetric linear Datalog.
2. Pol($\underline{B}$) $\models$ noname (a specific set of height-one identities)
3. $\underline{B}$ does not pp-construct st-Conn or $(\mathbb{Z}_p; +, 1)$ for all primes p
4. CSP($\underline{B}$) in L (deterministic logspace)

Theorem (Carvalho+Dalmau+Krokhin 2010). Equivalent:

1. CSP($\underline{B}$) can be solved by linear monadic arc Datalog
2. Pol($\underline{B}$) $\models$ n-k-abs(n, k) $\forall n$

$$f(x_{11}, \dots, x_{nk}) \approx f(y_{11}, \dots, y_{nk}) \text{ if } \{S_1, \dots, S_n\} = \{T_1, \dots, T_n\},$$
$$S_i = \{x_{i1}, \dots, x_{ik}\}, T_i = \{y_{11}, \dots, y_{ik}\}$$
$$f(S_1, S_2, \dots, S_n) \approx f(S_2, S_2, \dots, S_n) \text{ if } S_2 \subseteq S_1$$

3. $\underline{B}$ has caterpillar duality.

Another Conjecture, Another Result

Conjecture (Egri,Dalmau,Larose,Tesson,Zadori). Equivalent:

1. CSP($\underline{B}$) can be solved by symmetric linear Datalog.
2. Pol($\underline{B}$) $\models$ noname (a specific set of height-one identities)
3. $\underline{B}$ does not pp-construct st-Conn or $(\mathbb{Z}_p; +, 1)$ for all primes p
4. CSP($\underline{B}$) in L (deterministic logspace)

Theorem (Carvalho+Dalmau+Krokhin 2010). Equivalent:

1. CSP($\underline{B}$) can be solved by linear monadic arc Datalog
2. Pol($\underline{B}$) $\models$ n-k-abs(n, k) $\forall n$

$$f(x_{11}, \ldots, x_{nk}) \approx f(y_{11}, \ldots, y_{nk}) \text{ if } \{S_1, \ldots, S_n\} = \{T_1, \ldots, T_n\},$$
$$S_i = \{x_{i1}, \ldots, x_{ik}\}, T_i = \{y_{11}, \ldots, y_{ik}\}$$
$$f(S_1, S_2, \ldots, S_n) \approx f(S_2, S_2, \ldots, S_n) \text{ if } S_2 \subseteq S_1$$

3. $\underline{B}$ has caterpillar duality.

The ‘smallest natural Datalog fragment’?

The Smallest Natural Datalog Fragment

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

Theorem (B+Starke'24). TFAE:

1. CSP($\underline{B}$) solved by SLAM Datalog

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

Theorem (B+Starke'24). TFAE:

1. CSP($\underline{B}$) solved by SLAM Datalog
2. $\underline{B}$ has unfolded caterpillar duality

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

Theorem (B+Starke'24). TFAE:

1. CSP($\underline{B}$) solved by SLAM Datalog
2. $\underline{B}$ has unfolded caterpillar duality
3. If Pol($\underline{B}$) $\not\models \Sigma$, then $\Sigma \models f(x) \approx f(y)$

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

Theorem (B+Starke'24). TFAE:

1. CSP($\underline{B}$) solved by SLAM Datalog
2. $\underline{B}$ has unfolded caterpillar duality
3. If Pol($\underline{B}$) $\not\models \Sigma$, then $\Sigma \models f(x) \approx f(y)$
4. Every set of height one identities that is satisfies in Pol(P_2) is also satisfied in Pol($\underline{B}$)

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

Theorem (B+Starke'24). TFAE:

1. CSP($\underline{B}$) solved by SLAM Datalog
2. $\underline{B}$ has unfolded caterpillar duality
3. If Pol($\underline{B}$) $\not\models \Sigma$, then $\Sigma \models f(x) \approx f(y)$
4. Every set of height one identities that is satisfies in Pol(P_2) is also satisfied in Pol($\underline{B}$)
5. Pol(P_2) $\rightarrow$ Pol($\underline{B}$)

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

Theorem (B+Starke'24). TFAE:

1. CSP($\underline{B}$) solved by SLAM Datalog
2. $\underline{B}$ has unfolded caterpillar duality
3. If Pol($\underline{B}$) $\not\models \Sigma$, then $\Sigma \models f(x) \approx f(y)$
4. Every set of height one identities that is satisfies in Pol(P_2) is also satisfied in Pol($\underline{B}$)
5. Pol$(P_2) \to$ Pol($\underline{B}$)
6. P_2 pp-constructs $\underline{B}$

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

Theorem (B+Starke'24). TFAE:

1. CSP($\underline{B}$) solved by SLAM Datalog
2. $\underline{B}$ has unfolded caterpillar duality
3. If Pol($\underline{B}$) $\not\models \Sigma$, then $\Sigma \models f(x) \approx f(y)$
4. Every set of height one identities that is satisfies in Pol(P_2) is also satisfied in Pol$(\underline{B})$
5. Pol$(P_2) \to$ Pol$(\underline{B})$
6. P_2 pp-constructs $\underline{B}$
7. $\underline{B}$ is homomorphically equivalent to a structure with Maltsev and lattice polymorphisms

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

Theorem (B+Starke'24). TFAE:

1. CSP($\underline{B}$) solved by SLAM Datalog
2. $\underline{B}$ has unfolded caterpillar duality
3. If $\text{Pol}(\underline{B}) \not\models \Sigma$, then $\Sigma \models f(x) \approx f(y)$
4. Every set of height one identities that is satisfies in $\text{Pol}(P_2)$ is also satisfied in $\text{Pol}(\underline{B})$
5. $\text{Pol}(P_2) \to \text{Pol}(\underline{B})$
6. P_2 pp-constructs $\underline{B}$
7. $\underline{B}$ is homomorphically equivalent to a structure with Maltsev and lattice polymorphisms
8. $\text{Pol}(\underline{B}) \models$ quasi-Maltsev, n-k-abs $\forall n, k$

The Smallest Natural Datalog Fragment

Symmetric linear arc monadic Datalog:

SLAM Datalog

Theorem (B+Starke'24). TFAE:

1. CSP($\underline{B}$) solved by SLAM Datalog
2. $\underline{B}$ has unfolded caterpillar duality
3. If Pol($\underline{B}$) $\not\models \Sigma$, then $\Sigma \models f(x) \approx f(y)$
4. Every set of height one identities that is satisfies in Pol(P_2) is also satisfied in Pol($\underline{B}$)
5. Pol$(P_2) \to$ Pol($\underline{B}$)
6. P_2 pp-constructs $\underline{B}$
7. $\underline{B}$ is homomorphically equivalent to a structure with Maltsev and lattice polymorphisms
8. Pol($\underline{B}$) $\models$ quasi-Maltsev, n-k-abs $\forall n, k$
9. $\underline{B}$ does not pp-construct T_3 or a $S_{P(G)}$ for some finite simple group G.

Open Problems

Open Problems

- Prove that every CSP that is not P-hard is in NC.

Open Problems

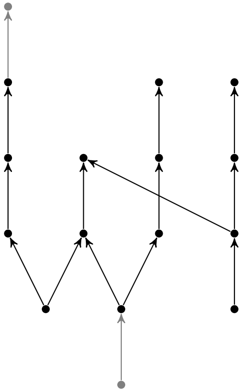
- Prove that every CSP that is not P-hard is in NC.
- Resolve the NL conjecture!

Open Problems

- Prove that every CSP that is not P-hard is in NC.
- Resolve the NL conjecture!
- Is even open when restricted to orientations of trees (B.+Bulin+Starke+Wernthaler 2023)

Open Problems

- Prove that every CSP that is not P-hard is in NC.
- Resolve the NL conjecture!
- Is even open when restricted to orientations of trees (B.+Bulin+Starke+Wernthaler 2023)
- There is a concrete tree with 16 vertices, whose CSP should be in NL, but we cannot prove it (B.+Bulin+Starke+Wernthaler 2023)



References

- Sebastian Meyer and Florian Starke: Finite Simple Groups in the Primitive Positive Constructability Poset, arXiv:2409.06487.
- Manuel Bodirsky and Florian Starke: Symmetric Linear Arc Monadic Datalog and Gadget Reductions, arXiv:2407.04924.
- Manuel Bodirsky, Jakub Bulín, Florian Starke, Michael Wernthaler: The smallest hard trees. Constraints, 28(2): 105-137 (2023).
- Manuel Bodirsky, Albert Vucaj, Dmitriy Zhuk: The lattice of clones of self-dual operations collapsed. Int. J. Algebra Comput., 2023.
- Manuel Bodirsky, Florian Starke: Maximal Digraphs with Respect to Primitive Positive Constructability. Combinatorica, 2022.
- Manuel Bodirsky, Florian Starke, Albert Vucaj: Smooth digraphs modulo primitive positive constructability and cyclic loop conditions. Int. J. Algebra Comput., 2021.
- Manuel Bodirsky and Albert Vucaj: Two-element structures modulo primitive positive constructability, Algebra Universalis, 2020.