Making the Infinite Finite:
Polymorphisms on Ramsey structures

Michael Pinsker

Université Denis Diderot - Paris 7  (60%)
Technische Universität Wien  (30%)
Hebrew University of Jerusalem  (10%)

Workshop on Algebra and CSPs
Fields Institute, Toronto, 2011
Part I
The global picture

Part II
Infinite template CSPs are natural
Homogeneous structures

Part III
Infinite polymorphisms → finite polymorphisms
Ramsey theory

Part IV
The past and the future
I liked the doors... I do not know what they mean, and they confused me, but they look nice.
“I liked the doors ... I do not know what they mean, and they confused me, but they look nice.”
Making the infinite finite

Michael Pinsker (Paris 7)
Welcome to the insane world of MP's talks

Madhouse of infinity

Making the infinite finite

Michael Pinsker (Paris 7)
Welcome to the insane world of MP’s talks
Welcome to the insane world of MP's talks
madhouse of infinity
Part I

Cloning is fun
The organizers of the workshop
Because most participants are [...]
The organizers of the workshop

Because most participants are [...] you can assume basic knowledge of algebra and CSP over a finite set, namely...
The organizers of the workshop

Because most participants are [...] you can assume basic knowledge of algebra and CSP over a finite set, namely

- pp-definitions, polymorphisms, the Galois correspondence
- the complexity of the CSP depends only on the variety generated by the polymorphism algebra, wlog idempotent
- the dichotomy conjecture
Cloning finite sheep

Let $\Gamma$ be a finite structure. Let $\text{Pol}(\Gamma)$ be its polymorphism clone. Let $A(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$. Equations $\rightarrow$ in $\text{P}$ No equations $\rightarrow$ $\text{NP}$-complete

Making the infinite finite

Michael Pinsker (Paris 7)
Cloning finite sheep

Let $\Gamma$ be a finite structure.
Cloning finite sheep

Let $\Gamma$ be a finite structure.
Let \( \Gamma \) be a finite structure. Let \( \text{Pol}(\Gamma) \) be its polymorphism clone.
Cloning finite sheep

Let $\Gamma$ be a finite structure.  Let $\text{Pol}(\Gamma)$ be its polymorphism clone.
Cloning finite sheep

Let $\Gamma$ be a finite structure. Let $\text{Pol}(\Gamma)$ be its polymorphism clone.

Let $\mathcal{A}(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$. 
Cloning finite sheep

Let $\Gamma$ be a finite structure. Let $\text{Pol}(\Gamma)$ be its polymorphism clone.

Let $\mathcal{A}(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$.
Let $\Gamma$ be a finite structure. Let $\text{Pol}(\Gamma)$ be its polymorphism clone.

Let $\mathcal{A}(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$.

Equations $\rightarrow$ in $P$
No equations $\rightarrow$ NP-complete

Michael Pinsker (Paris 7)
Let $\Gamma$ be an infinite structure.

For nice $\Gamma$: $\Gamma\text{Pol}(\Gamma)$

Let $A(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$.

Abstractions seem possible.

Reduction to the finite?

Making the infinite finite

Michael Pinsker (Paris 7)
Let $\Gamma$ be an infinite structure.
Let $\Gamma$ be an infinite structure.
Cloning infinite sheep

Let $\Gamma$ be an infinite structure.

For nice $\Gamma$:

$\Gamma$
Cloning infinite sheep

Let $\Gamma$ be an infinite structure.

For nice $\Gamma$:

$\Gamma$  

$\text{Pol}(\Gamma)$
Let $\Gamma$ be an infinite structure.

For nice $\Gamma$:

Let $\mathcal{A}(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$.
Cloning infinite sheep

Let $\Gamma$ be an infinite structure.

For nice $\Gamma$:

Let $\mathfrak{A}(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$.

$\mathfrak{A}(\text{Pol}(\Gamma))$
Cloning infinite sheep

Let $\Gamma$ be an infinite structure.

Let $\Gamma$ be an infinite structure.

For nice $\Gamma$:

Let $\mathcal{A}(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$.

Let $\mathcal{A}(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$.

Abstractions seem possible.

Reduction to the finite?

Making the infinite finite

Michael Pinsker (Paris 7)
Cloning infinite sheep

Let $\Gamma$ be an infinite structure.

For nice $\Gamma$:

Let $\mathcal{A}(\text{Pol}(\Gamma))$ be the abstraction of $\text{Pol}(\Gamma)$.

Abstractions seem possible. Reduction to the finite?
Wanted: Reduction of a certain class of infinite CSPs to finite CSPs. This involves:

- Model theory (pp-definability, homogeneous templates $\Gamma$)
- Ramsey theory (analyzing polymorphisms, make them finite for algorithms)
- Topological dynamics (topological automorphism groups and clones)
- Set theory (automatic continuity: topological clones vs. abstract clones)
- Universal algebra (equations)
- Complexity theory (algorithms)

It might never work out. But imagine it does...

Making the infinite finite

Michael Pinsker (Paris 7)
Science fiction

Wanted: Reduction of a certain class of infinite CSPs to finite CSPs. This involves:

- Model theory  
  (pp-definability, homogeneous templates \( \Gamma \))
- Ramsey theory  
  (analyzing polymorphisms, make them finite for algorithms)
- Topological dynamics  
  (topological automorphism groups and clones)
- Set theory  
  (automatic continuity: topological clones vs. abstract clones)
- Universal algebra  
  (equations)
- Complexity theory  
  (algorithms)
Science fiction

Wanted: Reduction of a certain class of infinite CSPs to finite CSPs. This involves:

- Model theory
  (pp-definability, homogeneous templates $\Gamma$)
- Ramsey theory
  (analyzing polymorphisms, make them finite for algorithms)
- Topological dynamics
  (topological automorphism groups and clones)
- Set theory
  (automatic continuity: topological clones vs. abstract clones)
- Universal algebra
  (equations)
- Complexity theory
  (algorithms)

It might never work out. But imagine it does...
We pass on to the next part.

Making the infinite finite

Michael Pinsker (Paris 7)
(We pass on to the next part.)
Part II

Do infinite sheep exist?
Infinite sheep in nature

Digraph acyclicity

Input: A finite directed graph \((V; E)\)

Question: Is \((V; E)\) acyclic?
Digraph acyclicity

Input: A finite directed graph \((V; E)\)

Question: Is \((V; E)\) acyclic?

Is CSP: template is \((\mathbb{Q}; <)\)
Infinite sheep in nature

Digraph acyclicity

Input: A finite directed graph $(V; E)$

Question: Is $(V; E)$ acyclic?

Is CSP: template is $(\mathbb{Q}; <)$

Betweenness

Input: A finite set of triples of variables $(x, y, z)$

Question: Is there a weak linear order on the variables such that for each triple either $x < y < z$ or $z < y < x$?
Infinite sheep in nature

Digraph acyclicity

Input: A finite directed graph \((V; E)\)

Question: Is \((V; E)\) acyclic?

Is CSP: template is \((\mathbb{Q}; <)\)

Betweenness

Input: A finite set of triples of variables \((x, y, z)\)

Question: Is there a weak linear order on the variables such that for each triple either \(x < y < z\) or \(z < y < x\)?

Is a CSP: template is \((\mathbb{Q}; \{(x, y, z) \mid (x < y < z) \lor (z < y < x)\})\)
More infinite sheep in nature

**Diophantine**

Input: A *finite* system of equations using $=, +, \cdot, 1$

Question: Is there a solution in $\mathbb{Z}$?
Diophantine

Input: A finite system of equations using $=, +, \cdot, 1$

Question: Is there a solution in $\mathbb{Z}$?

Is a CSP: template is $\Gamma := (\mathbb{Z}; 1, +, \cdot, =)$
More infinite sheep in nature

**Diophantine**

Input: A *finite* system of equations using $=, +, \cdot, 1$

Question: Is there a solution in $\mathbb{Z}$?

Is a CSP: template is $\Gamma := (\mathbb{Z}; 1, +, \cdot, =)$

**$K_n$-freeness**

Input: A *finite* undirected graph

Question: Is the graph $K_n$-free?
More infinite sheep in nature

**Diophantine**

Input: A finite system of equations using $=, +, \cdot, 1$

Question: Is there a solution in $\mathbb{Z}$?

Is a CSP: template is $\Gamma := (\mathbb{Z}; 1, +, \cdot, =)$

**$K_n$-freeness**

Input: A finite undirected graph

Question: Is the graph $K_n$-free?

Is a CSP: template is the homogeneous universal $K_n$-free graph
Even more infinite sheep in nature

Making the infinite finite  Michael Pinsker  (Paris 7)
Even more infinite sheep in nature

Klagenfurt sheep
The Graph Satisfiability Problem

Let $E$ be a binary relation symbol. (Imagine: edge relation of an undirected graph.) Let $\Psi$ be a finite set of quantifier-free $\{E\}$-formulas. Computational problem: $\text{Graph-SAT}(\Psi)$

**INPUT:** A finite set $W$ of variables (vertices), and statements $\phi_1,...,\phi_n$ about the elements of $W$, where each $\phi_i$ is taken from $\Psi$.

**QUESTION:** Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable in a graph?

Computational complexity depends on $\Psi$. Always in NP.

**Question** For which $\Psi$ is $\text{Graph-SAT}(\Psi)$ tractable?

Making the infinite finite

Michael Pinsker (Paris 7)
The Graph Satisfiability Problem

Let $E$ be a binary relation symbol.
(Imagine: edge relation of an undirected graph.)
Let $\psi$ be a finite set of quantifier-free $\{E\}$-formulas.
The Graph Satisfiability Problem

Let $E$ be a binary relation symbol. (Imagine: edge relation of an undirected graph.) Let $\Psi$ be a finite set of quantifier-free $\{E\}$-formulas.

**Computational problem: Graph-SAT($\Psi$)**

**INPUT:**
- A finite set $W$ of variables (vertices), and
- statements $\phi_1, \ldots, \phi_n$ about the elements of $W$, where each $\phi_i$ is taken from $\Psi$.

**QUESTION:** Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable in a graph?
The Graph Satisfiability Problem

Let $E$ be a binary relation symbol.
(Imagine: edge relation of an undirected graph.)
Let $\psi$ be a finite set of quantifier-free $\{E\}$-formulas.

**Computational problem: Graph-SAT($\psi$)**

**INPUT:**
- A finite set $W$ of variables (vertices), and
- statements $\phi_1, \ldots, \phi_n$ about the elements of $W$, where each $\phi_i$ is taken from $\psi$.

**QUESTION:** Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable in a graph?

Computational complexity depends on $\psi$. Always in NP.
The Graph Satisfiability Problem

Let $E$ be a binary relation symbol.
(Imagine: edge relation of an undirected graph.)
Let $\Psi$ be a finite set of quantifier-free $\{E\}$-formulas.

Computational problem: Graph-SAT($\Psi$)

**INPUT:**
- A finite set $W$ of variables (vertices), and
- statements $\phi_1, \ldots, \phi_n$ about the elements of $W$, where each $\phi_i$ is taken from $\Psi$.

**QUESTION:** Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable in a graph?

Computational complexity depends on $\Psi$. Always in NP.

**Question**

For which $\Psi$ is Graph-SAT($\Psi$) tractable?
Graph-SAT: Examples

Example 1
Let $\Psi_1$ only contain
$\psi_1(x, y, z) := (E(x, y) \land \neg E(y, z) \land \neg E(x, z)) \lor (\neg E(x, y) \land E(y, z) \land \neg E(x, z)) \lor (\neg E(x, y) \land \neg E(y, z) \land E(x, z)) \lor (E(x, y) \land E(y, z) \land E(x, z))$.

Graph-SAT($\Psi_1$) is NP-complete.

Example 2
Let $\Psi_2$ only contain
$\psi_2(x, y, z) := (E(x, y) \land \neg E(y, z) \land \neg E(x, z)) \lor (\neg E(x, y) \land E(y, z) \land \neg E(x, z)) \lor (\neg E(x, y) \land \neg E(y, z) \land E(x, z)) \lor (E(x, y) \land E(y, z) \land E(x, z))$.

Graph-SAT($\Psi_2$) is in P.
Graph-SAT: Examples

Example 1  Let $\Psi_1$ only contain

\[
\psi_1(x, y, z) := (E(x, y) \land \neg E(y, z) \land \neg E(x, z)) \\
\lor (\neg E(x, y) \land E(y, z) \land \neg E(x, z)) \\
\lor (\neg E(x, y) \land \neg E(y, z) \land E(x, z))
\]

$\text{Graph-SAT}(\Psi_1)$ is NP-complete.

Example 2  Let $\Psi_2$ only contain

\[
\psi_2(x, y, z) := (E(x, y) \land \neg E(y, z) \land \neg E(x, z)) \\
\lor (\neg E(x, y) \land E(y, z) \land \neg E(x, z)) \\
\lor (\neg E(x, y) \land \neg E(y, z) \land E(x, z)) \\
\lor (E(x, y) \land E(y, z) \land E(x, z))
\]

$\text{Graph-SAT}(\Psi_2)$ is in P.
Example 1 Let $\Psi_1$ only contain

$$\psi_1(x, y, z) := (E(x, y) \land \neg E(y, z) \land \neg E(x, z))$$
$$\lor (\neg E(x, y) \land E(y, z) \land \neg E(x, z))$$
$$\lor (\neg E(x, y) \land \neg E(y, z) \land E(x, z)) .$$

Graph-SAT($\Psi_1$) is NP-complete.
Graph-SAT: Examples

Example 1  Let $\Psi_1$ only contain

$$
\psi_1(x, y, z) := (E(x, y) \land \neg E(y, z) \land \neg E(x, z)) \\
\lor (\neg E(x, y) \land E(y, z) \land \neg E(x, z)) \\
\lor (\neg E(x, y) \land \neg E(y, z) \land E(x, z)) \\
\lor (E(x, y) \land E(y, z) \land E(x, z)) .
$$

Graph-SAT($\Psi_1$) is NP-complete.

Example 2  Let $\Psi_2$ only contain

$$
\psi_2(x, y, z) := (E(x, y) \land \neg E(y, z) \land \neg E(x, z)) \\
\lor (\neg E(x, y) \land E(y, z) \land \neg E(x, z)) \\
\lor (\neg E(x, y) \land \neg E(y, z) \land E(x, z)) \\
\lor (E(x, y) \land E(y, z) \land E(x, z)) .
$$
Graph-SAT: Examples

**Example 1**  Let $\Psi_1$ only contain

$$\psi_1(x, y, z) := (E(x, y) \land \neg E(y, z) \land \neg E(x, z))$$
$$\lor (\neg E(x, y) \land E(y, z) \land \neg E(x, z))$$
$$\lor (\neg E(x, y) \land \neg E(y, z) \land E(x, z))$$
$$\lor (E(x, y) \land E(y, z) \land E(x, z)) .$$

Graph-SAT($\Psi_1$) is NP-complete.

**Example 2**  Let $\Psi_2$ only contain

$$\psi_2(x, y, z) := (E(x, y) \land \neg E(y, z) \land \neg E(x, z))$$
$$\lor (\neg E(x, y) \land E(y, z) \land \neg E(x, z))$$
$$\lor (\neg E(x, y) \land \neg E(y, z) \land E(x, z))$$
$$\lor (E(x, y) \land E(y, z) \land E(x, z)) .$$

Graph-SAT($\Psi_2$) is in P.
Graph formulas and reducts of the random graph

Let $G = (V; E)$ denote the random graph, i.e., the unique countably infinite graph which is homogeneous, i.e., for all finite $A, B \subseteq G$, for all isomorphisms $i: A \rightarrow B$ there exists $\alpha \in Aut(G)$ extending $i$. Universal, i.e., contains all finite (even countable) graphs.

For a graph formula $\psi(x_1, \ldots, x_n)$, define a relation $R_\psi := \{(a_1, \ldots, a_n) \in V^n: \psi(a_1, \ldots, a_n)\}$.

For a set $\Psi$ of graph formulas, define a structure $\Gamma_\Psi := (V; (R_\psi: \psi \in \Psi))$.

$\Gamma_\Psi$ is a reduct of the random graph, i.e., a structure with a first-order definition in $G$.
Graph formulas and reducts of the random graph

Let $G = (V; E)$ denote the \textit{random graph}, i.e., the unique countably infinite graph which is universal, i.e., contains all finite (even countable) graphs.

For a graph formula $\psi(x_1, \ldots, x_n)$, define a relation $R_\psi := \{(a_1, \ldots, a_n) \in V^n : \psi(a_1, \ldots, a_n)\}$.

For a set $\Psi$ of graph formulas, define a structure $\Gamma_\Psi := (V; (R_\psi : \psi \in \Psi))$. $\Gamma_\Psi$ is a reduct of the random graph, i.e., a structure with a first-order definition in $G$. 

Michael Pinsker (Paris 7)
Graph formulas and reducts of the random graph

Let $G = (V; E)$ denote the random graph, i.e., the unique countably infinite graph which is

- homogeneous, i.e.,

  For all finite $A, B \subseteq G$, for all isomorphisms $i : A \to B$ there exists $\alpha \in \text{Aut}(G)$ extending $i$. 
Graph formulas and reducts of the random graph

Let $G = (V; E)$ denote the *random graph*, i.e., the unique countably infinite graph which is

- *homogeneous*, i.e.,
  
  For all finite $A, B \subseteq G$, for all isomorphisms $i : A \to B$
  there exists $\alpha \in \text{Aut}(G)$ extending $i$.

- *universal*, i.e., contains all finite (even countable) graphs.
Graph formulas and reducts of the random graph

Let $G = (V; E)$ denote the random graph, i.e., the unique countably infinite graph which is

- **homogeneous**, i.e.,
  
  For all finite $A, B \subseteq G$, for all isomorphisms $i : A \rightarrow B$
  there exists $\alpha \in \text{Aut}(G)$ extending $i$.

- **universal**, i.e., contains all finite (even countable) graphs.

For a graph formula $\psi(x_1, \ldots , x_n)$, define a relation

$$R_\psi := \{(a_1, \ldots , a_n) \in V^n : \psi(a_1, \ldots , a_n)\}.$$
Let $G = (V; E)$ denote the random graph, i.e., the unique countably infinite graph which is

- **homogeneous**, i.e.,
  
  For all finite $A, B \subseteq G$, for all isomorphisms $i : A \to B$ there exists $\alpha \in \text{Aut}(G)$ extending $i$.

- **universal**, i.e., contains all finite (even countable) graphs.

For a graph formula $\psi(x_1, \ldots, x_n)$, define a relation

$$R_\psi := \{(a_1, \ldots, a_n) \in V^n : \psi(a_1, \ldots, a_n)\}.$$

For a set $\Psi$ of graph formulas, define a structure

$$\Gamma_\psi := (V; (R_\psi : \psi \in \Psi)).$$
Graph formulas and reducts of the random graph

Let $G = (V; E)$ denote the random graph, i.e., the unique countably infinite graph which is

- **homogeneous**, i.e.,
  
  For all finite $A, B \subseteq G$, for all isomorphisms $i : A \to B$ there exists $\alpha \in \text{Aut}(G)$ extending $i$.

- **universal**, i.e., contains all finite (even countable) graphs.

For a graph formula $\psi(x_1, \ldots, x_n)$, define a relation

$$R_\psi := \{(a_1, \ldots, a_n) \in V^n : \psi(a_1, \ldots, a_n)\}.$$

For a set $\Psi$ of graph formulas, define a structure

$$\Gamma_\Psi := (V; (R_\psi : \psi \in \Psi)).$$

$\Gamma_\Psi$ is a **reduct** of the random graph, i.e.,

a structure with a first-order definition in $G$. 
An instance $W = \{w_1, \ldots, w_m\}$ of Graph-SAT($\Psi$) has a positive solution $\leftrightarrow$ the sentence $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$ holds in $\Gamma_{\Psi}$. So Graph-SAT($\Psi$) and CSP($\Gamma_{\Psi}$) are one and the same problem.

Could have used any universal graph. Classifying the complexity of all Graph-SAT problems is the same as classifying the complexity of CSPs of all reducts of the random graph.

Making the infinite finite

Michael Pinsker (Paris 7)
Graph-SAT as CSP

An instance

- \( \mathcal{W} = \{ w_1, \ldots, w_m \} \)
- \( \phi_1, \ldots, \phi_n \)

of Graph-SAT(\( \psi \)) has a positive solution \( \Leftrightarrow \)
the sentence \( \exists w_1, \ldots, w_m. \bigwedge_i \phi_i \) holds in \( \Gamma_\psi \).
Graph-SAT as CSP

An instance

- $\mathcal{W} = \{w_1, \ldots, w_m\}$
- $\phi_1, \ldots, \phi_n$

of Graph-SAT($\Psi$) has a positive solution $\iff$
the sentence $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$ holds in $\Gamma_\Psi$.

So Graph-SAT($\Psi$) and CSP($\Gamma_\Psi$) are one and the same problem.
Graph-SAT as CSP

An instance

- $W = \{w_1, \ldots, w_m\}$
- $\phi_1, \ldots, \phi_n$

of Graph-SAT($\Psi$) has a positive solution $\iff$ the sentence $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$ holds in $\Gamma_\Psi$.

So Graph-SAT($\Psi$) and CSP($\Gamma_\Psi$) are one and the same problem.

Could have used any universal graph.
Graph-SAT as CSP

An instance

- $W = \{w_1, \ldots, w_m\}$
- $\phi_1, \ldots, \phi_n$

of Graph-SAT($\Psi$) has a positive solution $\iff$ the sentence $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$ holds in $\Gamma_\Psi$.

So Graph-SAT($\Psi$) and CSP($\Gamma_\Psi$) are one and the same problem.

Could have used any universal graph.

Classifying the complexity of all Graph-SAT problems is the same as classifying the complexity of CSPs of all reducts of the random graph.
The Boolean satisfiability problem

Let $\Psi$ be a finite set of propositional formulas. Computational problem: Boolean-SAT($\Psi$)

**INPUT:** A finite set $W$ of propositional variables, and statements $\phi_1, \ldots, \phi_n$ about the variables in $W$, where each $\phi_i$ is taken from $\Psi$.

**QUESTION:** Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable?

Computational complexity depends on $\Psi$. Always in NP.

**Question** For which $\Psi$ is Boolean-SAT($\Psi$) tractable?

Making the infinite finite

Michael Pinsker (Paris 7)
The Boolean satisfiability problem

Let $\psi$ be a finite set of propositional formulas.
The Boolean satisfiability problem

Let $\psi$ be a finite set of propositional formulas.

**Computational problem: Boolean-SAT($\psi$)**

**INPUT:**
- A finite set $W$ of propositional variables, and
- statements $\phi_1, \ldots, \phi_n$ about the variables in $W$, where each $\phi_i$ is taken from $\psi$.

**QUESTION:** Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable?
The Boolean satisfiability problem

Let $\Psi$ be a finite set of propositional formulas.

Computational problem: Boolean-SAT($\Psi$)

**INPUT:**
- A finite set $W$ of propositional variables, and
- statements $\phi_1, \ldots, \phi_n$ about the variables in $W$, where each $\phi_i$ is taken from $\Psi$.

**QUESTION:** Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable?

Computational complexity depends on $\Psi$. Always in NP.

Making the infinite finite

Michael Pinsker (Paris 7)
The Boolean satisfiability problem

Let $\Psi$ be a finite set of propositional formulas.

Computational problem: Boolean-SAT($\Psi$)

INPUT:
- A finite set $W$ of propositional variables, and
- statements $\phi_1, \ldots, \phi_n$ about the variables in $W$, where each $\phi_i$ is taken from $\Psi$.

QUESTION: Is $\wedge_{1 \leq i \leq n} \phi_i$ satisfiable?

Computational complexity depends on $\Psi$. Always in NP.

Question

For which $\Psi$ is Boolean-SAT($\Psi$) tractable?
Boolean-SAT as CSP

For a Boolean formula $\psi(x_1, \ldots, x_n)$, define a relation $R_\psi := \{(a_1, \ldots, a_n) \in \{0, 1\}^n : \psi(a_1, \ldots, a_n)\}$.

For a set $\Psi$ of Boolean formulas, define a structure $\Gamma_\Psi := (\{0, 1\}; (R_\psi : \psi \in \Psi))$.

An instance $W = \{w_1, \ldots, w_m\}$ of Boolean-SAT$(\Psi)$ has a positive solution $\iff$ the sentence $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$ holds in $\Gamma_\Psi$.

So Boolean-SAT$(\Psi)$ and CSP$(\Gamma_\Psi)$ are one and the same problem.
Boolean-SAT as CSP

For a Boolean formula $\psi(x_1, \ldots, x_n)$, define a relation

$$R_\psi := \{(a_1, \ldots, a_n) \in \{0, 1\}^n : \psi(a_1, \ldots, a_n)\}.$$
Boolean-SAT as CSP

For a Boolean formula $\psi(x_1, \ldots, x_n)$, define a relation

$$R_\psi := \{(a_1, \ldots, a_n) \in \{0, 1\}^n : \psi(a_1, \ldots, a_n)\}.$$ 

For a set $\Psi$ of Boolean formulas, define a structure

$$\Gamma_\Psi := (\{0, 1\}; (R_\psi : \psi \in \Psi)).$$
Boolean-SAT as CSP

For a Boolean formula $\psi(x_1, \ldots, x_n)$, define a relation

$$R_\psi := \{(a_1, \ldots, a_n) \in \{0, 1\}^n : \psi(a_1, \ldots, a_n)\}.$$

For a set $\Psi$ of Boolean formulas, define a structure

$$\Gamma_\Psi := (\{0, 1\}; (R_\psi : \psi \in \Psi)).$$

An instance

- $W = \{w_1, \ldots, w_m\}$
- $\phi_1, \ldots, \phi_n$

of Boolean-SAT($\Psi$) has a positive solution $\iff$ the sentence $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$ holds in $\Gamma_\Psi$. 
Boolean-SAT as CSP

For a Boolean formula $\psi(x_1, \ldots, x_n)$, define a relation

$$R_\psi := \{(a_1, \ldots, a_n) \in \{0, 1\}^n : \psi(a_1, \ldots, a_n)\}.$$

For a set $\Psi$ of Boolean formulas, define a structure

$$\Gamma_\psi := (\{0, 1\}; (R_\psi : \psi \in \Psi)).$$

An instance

- $W = \{w_1, \ldots, w_m\}

- \phi_1, \ldots, \phi_n$

of Boolean-SAT($\Psi$) has a positive solution $\iff$ the sentence $\exists w_1, \ldots, w_m. \land_i \phi_i$ holds in $\Gamma_\psi$.

So Boolean-SAT($\Psi$) and CSP($\Gamma_\psi$) are one and the same problem.
Temporal constraints

Let $<\mathbin{\vphantom{<}}$ be a binary relation symbol. (Imagine: linear order relation.) Let $\Psi$ be a finite set of quantifier-free \{$<\mathbin{\vphantom{<}}$\}-formulas. Computational problem: Temp-SAT($\Psi$)

**INPUT:** A finite set $W$ of variables (vertices), and statements $\phi_1, \ldots, \phi_n$ about the elements of $W$, where each $\phi_i$ is taken from $\Psi$.

**QUESTION:** Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable in a linear order?

Computational complexity depends on $\Psi$. Always in NP.

Question: For which $\Psi$ is Temp-SAT($\Psi$) tractable?
Temporal constraints

Let \(<\) be a binary relation symbol.
(Imagine: linear order relation.)
Let \(\Psi\) be a finite set of quantifier-free \(\{<\}\)-formulas.

Computational problem: Temp-SAT(\(\Psi\))

**INPUT:** A finite set \(W\) of variables (vertices), and statements \(\phi_1, \ldots, \phi_n\) about the elements of \(W\), where each \(\phi_i\) is taken from \(\Psi\).

**QUESTION:** Is \(\bigwedge_{1 \leq i \leq n} \phi_i\) satisfiable in a linear order?

Computational complexity depends on \(\Psi\). Always in \(\mathsf{NP}\).

Question: For which \(\Psi\) is Temp-SAT(\(\Psi\)) tractable?
Temporal constraints

Let $<$ be a binary relation symbol.
(Imagine: linear order relation.)
Let $\Psi$ be a finite set of quantifier-free \{\textit{<}\}\-formulas.

Computational problem: Temp-SAT($\Psi$)

INPUT:
- A finite set $W$ of variables (vertices), and
- statements $\phi_1, \ldots, \phi_n$ about the elements of $W$, where each $\phi_i$ is taken from $\Psi$.

QUESTION: Is $\wedge_{1 \leq i \leq n} \phi_i$ satisfiable in a linear order?
Temporal constraints

Let $<$ be a binary relation symbol.
(Imagine: linear order relation.)
Let $\Psi$ be a finite set of quantifier-free \{<\}-formulas.

Computational problem: Temp-SAT($\Psi$)

INPUT:
- A finite set $W$ of variables (vertices), and
- statements $\phi_1, \ldots, \phi_n$ about the elements of $W$, where each $\phi_i$ is taken from $\Psi$.

QUESTION: Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable in a linear order?

Computational complexity depends on $\Psi$. Always in NP.
Temporal constraints

Let $<$ be a binary relation symbol. (Imagine: linear order relation.)
Let $\Psi$ be a finite set of quantifier-free $\{<\}$-formulas.

**Computational problem: Temp-SAT($\Psi$)**

**INPUT:**
- A finite set $W$ of variables (vertices), and
- statements $\phi_1, \ldots, \phi_n$ about the elements of $W$, where each $\phi_i$ is taken from $\Psi$.

**QUESTION:** Is $\bigwedge_{1 \leq i \leq n} \phi_i$ satisfiable in a linear order?

Computational complexity depends on $\Psi$. Always in NP.

**Question**

For which $\Psi$ is Temp-SAT($\Psi$) tractable?
Temporal formulas and reducts of \((\mathbb{Q}; <)\)
Temporal formulas and reducts of \((\mathbb{Q}; <)\)

Let \((\mathbb{Q}; <)\) denote the order of the rationals.
Temporal formulas and reducts of \((\mathbb{Q}; <)\)

Let \((\mathbb{Q}; <)\) denote the order of the rationals.

For a \(\{<\}\)-formula \(\psi(x_1, \ldots, x_n)\), define a relation

\[
R_\psi := \{(a_1, \ldots, a_n) \in \mathbb{Q}^n : \psi(a_1, \ldots, a_n)\}.
\]
Temporal formulas and reducts of \((\mathbb{Q}; <)\)

Let \((\mathbb{Q}; <)\) denote the order of the rationals.

For a \(\{<\}\)-formula \(\psi(x_1, \ldots, x_n)\), define a relation

\[ R_\psi := \{(a_1, \ldots, a_n) \in \mathbb{Q}^n : \psi(a_1, \ldots, a_n)\}. \]

For a set \(\Psi\) of \(\{<\}\)-formulas, define a structure

\[ \Gamma_\Psi := (\mathbb{Q}; (R_\psi : \psi \in \Psi)). \]
Temporal formulas and reducts of \((\mathbb{Q}; <)\)

Let \((\mathbb{Q}; <)\) denote the order of the rationals.

For a \(\{<\}\)-formula \(\psi(x_1, \ldots, x_n)\), define a relation

\[
R_\psi := \{(a_1, \ldots, a_n) \in \mathbb{Q}^n : \psi(a_1, \ldots, a_n)\}.
\]

For a set \(\Psi\) of \(\{<\}\)-formulas, define a structure

\[
\Gamma_\Psi := (\mathbb{Q}; (R_\psi : \psi \in \Psi)).
\]

\(\Gamma_\Psi\) is a reduct of \((\mathbb{Q}; <)\).
Let \((\mathbb{Q}; <)\) denote the order of the rationals.

For a \(\{<\}\)-formula \(\psi(x_1, \ldots, x_n)\), define a relation
\[
R_\psi := \{(a_1, \ldots, a_n) \in \mathbb{Q}^n : \psi(a_1, \ldots, a_n)\}.
\]

For a set \(\Psi\) of \(\{<\}\)-formulas, define a structure
\[
\Gamma_\Psi := (\mathbb{Q}; (R_\psi : \psi \in \Psi)).
\]

\(\Gamma_\Psi\) is a reduct of \((\mathbb{Q}; <)\).

Temp-SAT(\(\Psi\)) and CSP(\(\Gamma_\Psi\)) are one and the same problem.
Temporal formulas and reducts of \((\mathbb{Q}; <)\)

Let \((\mathbb{Q}; <)\) denote the order of the rationals.

For a \{<\}-formula \(\psi(x_1, \ldots, x_n)\), define a relation

\[ R_\psi := \{(a_1, \ldots, a_n) \in \mathbb{Q}^n : \psi(a_1, \ldots, a_n)\} . \]

For a set \(\Psi\) of \{<\}-formulas, define a structure

\[ \Gamma_\Psi := (\mathbb{Q}; (R_\psi : \psi \in \Psi)) . \]

\(\Gamma_\Psi\) is a reduct of \((\mathbb{Q}; <)\).

\(\text{Temp-SAT}(\Psi)\) and \(\text{CSP}(\Gamma_\Psi)\) are one and the same problem.

Could have used any infinite linear order, but \((\mathbb{Q}; <)\) is homogeneous.
Three classification theorems

All problems Boolean-SAT($\psi$), Graph-SAT($\psi$), and Temp-SAT($\psi$) are either in P or NP-complete.
Three classification theorems

All problems $\text{Boolean-SAT}(\psi)$, $\text{Graph-SAT}(\psi)$, and $\text{Temp-SAT}(\psi)$ are either in P or NP-complete.

Given $\psi$, we can decide in which class the problem falls.
Three classification theorems

All problems Boolean-SAT($\psi$), Graph-SAT($\psi$), and Temp-SAT($\psi$) are either in P or NP-complete.

Given $\psi$, we can decide in which class the problem falls.

- **Boolean-SAT**: Schaefer (1978)
- **Temp-SAT**: Bodirsky and Kára (2007)
- **Graph-SAT**: Bodirsky and MP (2010)
Homogeneous structures

**Graph-SAT(ψ):** Is there a finite graph such that... (constraints)

**Temp-SAT(ψ):** Is there a linear order such that...
Homogeneous structures

Graph-SAT(\(\psi\)): Is there a finite graph such that... (constraints)

Temp-SAT(\(\psi\)): Is there a linear order such that...

The classes of finite graphs and linear orders are amalgamation classes.
Fraïssé’s theorem

Theorem (Fraïssé)

- If $\mathcal{C}$ is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with age $\mathcal{C}$.
- The age of a homogeneous structure is an amalgamation class.
Fraïssé’s theorem

**Theorem (Fraïssé)**

- If $\mathcal{C}$ is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with age $\mathcal{C}$.
- The age of a homogeneous structure is an amalgamation class.

**Further amalgamation classes.**
Fraïssé’s theorem

Theorem (Fraïssé)

- If $\mathcal{C}$ is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with age $\mathcal{C}$.
- The age of a homogeneous structure is an amalgamation class.

Further amalgamation classes.

- Partial orders
Fraïssé’s theorem

Theorem (Fraïssé)

- If $C$ is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with \textit{age} $C$.
- The age of a homogeneous structure is an amalgamation class.

Further amalgamation classes.

- Partial orders
- Lattices (Jónsson)
Fraïssé’s theorem

Theorem (Fraïssé)
- If $\mathcal{C}$ is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with age $\mathcal{C}$.
- The age of a homogeneous structure is an amalgamation class.

Further amalgamation classes.
- Partial orders
- Lattices (Jónsson)
- Distributive lattices (Pierce)
Fraïssé’s theorem

**Theorem (Fraïssé)**

- If $\mathcal{C}$ is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with age $\mathcal{C}$.
- The age of a homogeneous structure is an amalgamation class.

**Further amalgamation classes.**

- Partial orders
- Lattices (Jónsson)
- Distributive lattices (Pierce)
- Trivial lattices (Day, Ježek)
Theorem (Fraïssé)

- If $\mathcal{C}$ is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with age $\mathcal{C}$.
- The age of a homogeneous structure is an amalgamation class.

Further amalgamation classes.

- Partial orders
- Lattices (Jónsson)
- Distributive lattices (Pierce)
- Trivial lattices (Day, Ježek)
- Metric spaces with rational distances
Fraïssé’s theorem

Theorem (Fraïssé)

- If \( \mathcal{C} \) is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with \( \text{age } \mathcal{C} \).
- The age of a homogeneous structure is an amalgamation class.

Further amalgamation classes.

- Partial orders
- Lattices (Jónsson)
- Distributive lattices (Pierce)
- Trivial lattices (Day, Ježek)
- Metric spaces with rational distances
- Tournaments
**Fraïssé’s theorem**

**Theorem (Fraïssé)**
- If $\mathcal{C}$ is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with age $\mathcal{C}$.
- The age of a homogeneous structure is an amalgamation class.

**Further amalgamation classes.**
- Partial orders
- Lattices (Jónsson)
- Distributive lattices (Pierce)
- Trivial lattices (Day, Ježek)
- Metric spaces with rational distances
- Tournaments
- Henson digraphs (forbidden tournaments)
Fraïssé’s theorem

Theorem (Fraïssé)

- If $\mathcal{C}$ is a countable class of structures closed under substructures which has amalgamation, then there exists a unique homogeneous structure with age $\mathcal{C}$.
- The age of a homogeneous structure is an amalgamation class.

Further amalgamation classes.

- Partial orders
- Lattices (Jónsson)
- Distributive lattices (Pierce)
- Trivial lattices (Day, Ježek)
- Metric spaces with rational distances
- Tournaments
- Henson digraphs (forbidden tournaments)

Homogeneous digraphs classified by Cherlin.
Making the infinite finite

Michael Pinsker (Paris 7)
Making the infinite finite

Michael Pinsker (Paris 7)
Part III

Making the infinite finite
Reducts of homogeneous structures

Let $\Delta$ be a countable homogeneous relational structure in a finite language. We call $\Delta$ the *base structure*. 
Reducts of homogeneous structures

Let $\Delta$ be a countable homogeneous relational structure in a finite language. We call $\Delta$ the *base structure*.

**Definition**

A *reduct* of $\Delta$ is a structure with a first-order definition in $\Delta$.

For us it makes sense to consider two reducts $\Gamma, \Gamma'$ of $\Delta$ *equivalent* iff $\Gamma$ has a pp-definition from $\Gamma'$ and vice-versa.
Let $\Delta$ be a countable homogeneous relational structure in a finite language. We call $\Delta$ the base structure.

**Definition**

A *reduct* of $\Delta$ is a structure with a first-order definition in $\Delta$.

For us it makes sense to consider two reducts $\Gamma, \Gamma'$ of $\Delta$ *equivalent* iff $\Gamma$ has a pp-definition from $\Gamma'$ and vice-versa.

We say that $\Gamma$ and $\Gamma'$ are *pp-interdefinable*. 
Let $\Delta$ be a countable homogeneous relational structure in a finite language. We call $\Delta$ the base structure.

**Definition**

A *reduct* of $\Delta$ is a structure with a first-order definition in $\Delta$.

For us it makes sense to consider two reducts $\Gamma, \Gamma'$ of $\Delta$ equivalent iff $\Gamma$ has a pp-definition from $\Gamma'$ and vice-versa.

We say that $\Gamma$ and $\Gamma'$ are *pp-interdefinable*.

The relation “$\Gamma$ is pp-definable in $\Gamma'$” is a quasiorder on the reducts.
Reducts of homogeneous structures

Let $\Delta$ be a countable homogeneous relational structure in a finite language. We call $\Delta$ the base structure.

**Definition**

A reduct of $\Delta$ is a structure with a first-order definition in $\Delta$.

For us it makes sense to consider two reducts $\Gamma$, $\Gamma'$ of $\Delta$ equivalent iff $\Gamma$ has a pp-definition from $\Gamma'$ and vice-versa.

We say that $\Gamma$ and $\Gamma'$ are pp-interdefinable.

The relation “$\Gamma$ is pp-definable in $\Gamma'$” is a quasiorder on the reducts.

We factor this quasiorder by the equivalence relation of pp-interdefinability, and obtain a complete lattice.
Reducts and closed clones

Problem
Classify the reducts of $\Delta$ up to pp-interdefinability.

Definition
A clone $C$ on $D$ is closed iff for each $n \geq 1$, the set of its $n$-ary functions $C \cap D^n$ is a closed subset of the Baire space $D^n$.

Theorem (Bodirsky, Nešetřil '03)
Let $\Delta$ be $\omega$-categorical (e.g., homogeneous in a finite language). Then $\Gamma \mapsto \text{Pol}(\Gamma)$ is a one-to-one correspondence between the primitive positive closed reducts of $\Delta$ and the closed clones containing $\text{Aut}(\Delta)$.
Problem
Classify the reducts of $\Delta$ up to pp-interdefinability.

Definition
A clone $\mathcal{C}$ on $D$ is \textit{closed} iff for each $n \geq 1$, the set of its $n$-ary functions $\mathcal{C} \cap D^{Dn}$ it is a closed subset of the Baire space $D^{Dn}$. 
Reducts and closed clones

Problem
Classify the reducts of $\Delta$ up to pp-interdefinability.

Definition
A clone $\mathcal{C}$ on $D$ is closed iff for each $n \geq 1$, the set of its $n$-ary functions $\mathcal{C} \cap D^n$ it is a closed subset of the Baire space $D^n$.

Theorem (Bodirsky, Nešetřil ’03)
Let $\Delta$ be $\omega$-categorical (e.g., homogeneous in a finite language). Then
$$\Gamma \mapsto \text{Pol}(\Gamma)$$
is a one-to-one correspondence between the primitive positive closed reducts of $\Delta$ and the closed clones containing $\text{Aut}(\Delta)$.
We thus have to understand the closed clones \( \supseteq Aut(\Delta) \).
We thus have to understand the closed clones $\supseteq \text{Aut}(\Delta)$.

Theorem (Bodirsky, Chen, MP ’08)

The structure $\Delta := (D; =)$ has $2^{\aleph_0}$ reducts up to primitive positive interdefinability.
We thus have to understand the closed clones $\supseteq \text{Aut}(\Delta)$.

**Theorem (Bodirsky, Chen, MP ’08)**

The structure $\Delta := (D; =)$ has $2^{\aleph_0}$ reducts up to primitive positive interdefinability.

Where is the border between NP-completeness and tractability?
We thus have to understand the closed clones $\supseteq \text{Aut}(\Delta)$.

**Theorem (Bodirsky, Chen, MP ’08)**

The structure $\Delta := (D; =)$ has $2^{\aleph_0}$ reducts up to primitive positive interdefinability.

Where is the border between NP-completeness and tractability?

Are we in NP at all?
Finite boundedness

There exist $2^{\aleph_0}$ non-isomorphic homogeneous digraphs.
Finite boundedness

There exist $2^\aleph_0$ non-isomorphic homogeneous digraphs. Easy to see that they have distinct CSPs.
Finite boundedness

There exist $2^{\aleph_0}$ non-isomorphic homogeneous digraphs.
Easy to see that they have distinct CSPs.
Thus there exist homogeneous digraphs with **undecidable** CSP.
Finite boundedness

There exist $2^\aleph_0$ non-isomorphic homogeneous digraphs.
Easy to see that they have distinct CSPs.
Thus there exist homogeneous digraphs with **undecidable** CSP.

**Definition**

A class $\mathcal{C}$ of $\tau$-structures is **finitely bounded** iff

there exists a finite set $\mathcal{F}$ of $\tau$-structures such that

for all $\tau$-structures $A$ ($A \in \mathcal{C}$ iff no $F \in \mathcal{F}$ embeds into $\mathcal{C}$).

$\mathcal{F}$... set of “forbidden substructures”
Finite boundedness

There exist $2^{|\omega|}$ non-isomorphic homogeneous digraphs. Easy to see that they have distinct CSPs. Thus there exist homogeneous digraphs with undecidable CSP.

**Definition**

A class $\mathcal{C}$ of $\tau$-structures is *finitely bounded* iff there exists a finite set $\mathcal{F}$ of $\tau$-structures such that for all $\tau$-structures $A$ ($A \in \mathcal{C}$ iff no $F \in \mathcal{F}$ embeds into $\mathcal{C}$).

$\mathcal{F}$ ... set of “forbidden substructures”

**Examples**
Finite boundedness

There exist $2^\aleph_0$ non-isomorphic homogeneous digraphs. Easy to see that they have distinct CSPs. Thus there exist homogeneous digraphs with **undecidable** CSP.

**Definition**

A class $\mathcal{C}$ of $\tau$-structures is *finitely bounded* iff there exists a finite set $\mathcal{F}$ of $\tau$-structures such that for all $\tau$-structures $A$ ($A \in \mathcal{C}$ iff no $F \in \mathcal{F}$ embeds into $\mathcal{C}$).

$\mathcal{F}$... set of “forbidden substructures”

**Examples**

- Partial orders
Finite boundedness

There exist \(2^{\aleph_0}\) non-isomorphic homogeneous digraphs. Easy to see that they have distinct CSPs. Thus there exist homogeneous digraphs with **undecidable CSP**.

**Definition**

A class \(\mathcal{C}\) of \(\tau\)-structures is *finitely bounded* iff there exists a finite set \(\mathcal{F}\) of \(\tau\)-structures such that for all \(\tau\)-structures \(A\) (\(A \in \mathcal{C}\) iff no \(F \in \mathcal{F}\) embeds into \(\mathcal{C}\)).

\(\mathcal{F}\) ... set of “forbidden substructures”

**Examples**

- Partial orders
- Lattices
Finite boundedness

There exist $2^{\aleph_0}$ non-isomorphic homogeneous digraphs. Easy to see that they have distinct CSPs. Thus there exist homogeneous digraphs with undecidable CSP.

Definition

A class $\mathcal{C}$ of $\tau$-structures is finitely bounded iff there exists a finite set $\mathcal{F}$ of $\tau$-structures such that for all $\tau$-structures $A$ ($A \in \mathcal{C}$ iff no $F \in \mathcal{F}$ embeds into $\mathcal{C}$).

$\mathcal{F}$... set of “forbidden substructures”

Examples

- Partial orders
- Lattices
- Graphs
Finite boundedness

There exist $2^{\aleph_0}$ non-isomorphic homogeneous digraphs. Easy to see that they have distinct CSPs. Thus there exist homogeneous digraphs with undecidable CSP.

Definition

A class $\mathcal{C}$ of $\tau$-structures is **finitely bounded** iff there exists a finite set $\mathcal{F}$ of $\tau$-structures such that for all $\tau$-structures $A$ ($A \in \mathcal{C}$ iff no $F \in \mathcal{F}$ embeds into $\mathcal{C}$).

$\mathcal{F}$... set of “forbidden substructures”

Examples

- Partial orders
- Lattices
- Graphs
- $K_n$-free graphs
Observation

If a homogeneous structure in a finite language is finitely bounded, then the CSP of its reducts is in NP.
Observation

If a homogeneous structure in a finite language is finitely bounded, then the CSP of its reducts is in NP.

Still, how to cope with infinite polymorphisms?
Observation

If a homogeneous structure in a finite language is finitely bounded, then the CSP of its reducts is in NP.

Still, how to cope with infinite polymorphisms?

Use Ramsey theory to make them finite.
Canonical functions on the Random graph

Let $G = (V; E)$ be the random graph.

**Definition.** $f : G \to G$ is *canonical* iff

\[
\text{if } (x, y) \text{ and } (u, v) \text{ have the same type in } G, \text{ then } (f(x), f(y)) \text{ and } (f(u), f(v)) \text{ have the same type in } G.
\]

**Examples.** Automorphisms / Embeddings are canonical. Constant functions are canonical. Homomorphisms are not necessarily canonical. $- : E \to E$ and $e : N \to N$ are canonical.
Canonical functions on the Random graph

Let $G = (V; E)$ be the random graph.

**Definition.** $f : G \rightarrow G$ is *canonical* iff for all $x, y, u, v \in V$, if $(x, y)$ and $(u, v)$ have the same type in $G$, then $(f(x), f(y))$ and $(f(u), f(v))$ have the same type in $G$. 

**Examples.** Automorphisms / Embeddings are canonical. Constant functions are canonical. $- : E \rightarrow V$ and $\varepsilon : N \rightarrow V$ are canonical.
Canonical functions on the Random graph

Let $G = (V; E)$ be the random graph.

**Definition.** $f : G \rightarrow G$ is *canonical* iff for all $x, y, u, v \in V$, if $(x, y)$ and $(u, v)$ have the same type in $G$, then $(f(x), f(y))$ and $(f(u), f(v))$ have the same type in $G$. 

Canonical functions on the Random graph

Let $G = (V; E)$ be the random graph.

**Definition.** $f : G \to G$ is *canonical* iff
for all $x, y, u, v \in V$,
if $(x, y)$ and $(u, v)$ have the same type in $G$,
then $(f(x), f(y))$ and $(f(u), f(v))$ have the same type in $G$.

**Examples.**

Automorphisms / Embeddings are canonical.
Constant functions are canonical.
Homomorphisms are not necessarily canonical.
$- : E$ and $e : N$ are canonical.
Let $G = (V; E)$ be the random graph.

**Definition.** $f : G \rightarrow G$ is *canonical* iff for all $x, y, u, v \in V$,

if $(x, y)$ and $(u, v)$ have the same type in $G$,
then $(f(x), f(y))$ and $(f(u), f(v))$ have the same type in $G$.

**Examples.**

- Automorphisms / Embeddings are canonical.
Canonical functions on the Random graph

Let $G = (V; E)$ be the random graph.

**Definition.** $f : G \rightarrow G$ is *canonical* iff for all $x, y, u, v \in V$, if $(x, y)$ and $(u, v)$ have the same type in $G$, then $(f(x), f(y))$ and $(f(u), f(v))$ have the same type in $G$.

**Examples.**

- Automorphisms / Embeddings are canonical.
- Constant functions are canonical.
Canonical functions on the Random graph

Let $G = (V; E)$ be the random graph.

**Definition.** $f : G \rightarrow G$ is *canonical* iff for all $x, y, u, v \in V$, if $(x, y)$ and $(u, v)$ have the same type in $G$, then $(f(x), f(y))$ and $(f(u), f(v))$ have the same type in $G$.

**Examples.**

- Automorphisms / Embeddings are canonical.
- Constant functions are canonical.
- Homomorphisms are not necessarily canonical.
Canonical functions on the Random graph

Let $G = (V; E)$ be the random graph.

**Definition.** $f : G \rightarrow G$ is canonical iff for all $x, y, u, v \in V$, if $(x, y)$ and $(u, v)$ have the same type in $G$, then $(f(x), f(y))$ and $(f(u), f(v))$ have the same type in $G$.

**Examples.**

- Automorphisms / Embeddings are canonical.
- Constant functions are canonical.
- Homomorphisms are not necessarily canonical.
- $-e$ is canonical.
Canonical functions on the Random graph

Let $G = (V; E)$ be the random graph.

**Definition.** $f : G \to G$ is *canonical* iff for all $x, y, u, v \in V$, if $(x, y)$ and $(u, v)$ have the same type in $G$, then $(f(x), f(y))$ and $(f(u), f(v))$ have the same type in $G$.

**Examples.**

- Automorphisms / Embeddings are canonical.
- Constant functions are canonical.
- Homomorphisms are not necessarily canonical.
- $-$ is canonical.
- $e_E$ and $e_N$ are canonical.
Finding canonical behaviour

The class of finite graphs has the following Ramsey property:

Given $f: G \rightarrow G$, color the edges of $G$ according to the type of their image: 3 possibilities. Same for non-edges. Conclusion: Every finite graph has a copy in $G$ on which $f$ is canonical.
Finding canonical behaviour

The class of finite graphs has the following **Ramsey property**:

For all graphs $H$
there exists a graph $S$ such that

Given $f: G \to G$, color the edges of $G$ according to the type of their image: 3 possibilities. Same for non-edges.

Conclusion: Every finite graph has a copy in $G$ on which $f$ is canonical.
Finding canonical behaviour

The class of finite graphs has the following **Ramsey property**:

For all graphs $H$
there exists a graph $S$ such that
if the edges of $S$ are colored with 3 colors,
Finding canonical behaviour

The class of finite graphs has the following Ramsey property:

For all graphs $H$
there exists a graph $S$ such that
if the edges of $S$ are colored with 3 colors,
then there is a copy of $H$ in $S$
on which the coloring is constant.
Finding canonical behaviour

The class of finite graphs has the following **Ramsey property**: 

For all graphs $H$
there exists a graph $S$ such that
if the edges of $S$ are colored with 3 colors,
then there is a copy of $H$ in $S$
on which the coloring is constant.

Given $f : G \rightarrow G$, color the edges of $G$
according to the type of their image: 3 possibilities.

Same for non-edges.
Finding canonical behaviour

The class of finite graphs has the following **Ramsey property**:

For all graphs $H$
there exists a graph $S$ such that
if the edges of $S$ are colored with 3 colors,
then there is a copy of $H$ in $S$
on which the coloring is constant.

Given $f : G \rightarrow G$, color the edges of $G$
according to the type of their image: 3 possibilities.

Same for non-edges.

**Conclusion:** Every finite graph has a copy in $G$ on which $f$ is canonical.
A canonical function $f : G \to G$ induces a function $f' : \{E, N, =\} \to \{E, N, =\}$ (i.e., a function on the 2-types of $G$).
Patterns in functions on the random graph

A canonical function $f : G \rightarrow G$ induces a function $f' : \{E, N, =\} \rightarrow \{E, N, =\}$ (i.e., a function on the 2-types of $G$).

Converse does not hold.
A canonical function $f : G \to G$ induces a function $f' : \{E, N, =\} \to \{E, N, =\}$ (i.e., a function on the 2-types of $G$).

Converse does not hold.

The following are all possibilities of canonical functions:
Patterns in functions on the random graph

A canonical function $f : G \rightarrow G$ induces a function $f' : \{E, N, =\} \rightarrow \{E, N, =\}$ (i.e., a function on the 2-types of $G$). Converse does not hold.

The following are all possibilities of canonical functions:

- Turning everything into edges ($e_E$)
A canonical function \( f : G \to G \) induces a function \( f' : \{E, N, =\} \to \{E, N, =\} \) (i.e., a function on the 2-types of \( G \)).

Converse does not hold.

The following are all possibilities of canonical functions:

- Turning everything into edges \( (e_E) \)
- Turning everything into non-edges \( (e_N) \)
A canonical function \( f : G \to G \) induces a function \( f' : \{E, N, =\} \to \{E, N, =\} \) (i.e., a function on the 2-types of \( G \)).

Converse does not hold.

The following are all possibilities of canonical functions:

- Turning everything into edges (\( e_E \))
- turning everything into non-edges (\( e_N \))
- behaving like –
A canonical function $f : G \to G$ induces a function $f' : \{E, N, =\} \to \{E, N, =\}$ (i.e., a function on the 2-types of $G$).

Converse does not hold.

The following are all possibilities of canonical functions:

- Turning everything into edges ($e_E$)
- turning everything into non-edges ($e_N$)
- behaving like –
- being constant
A canonical function $f : G \rightarrow G$ induces a function $f' : \{E, N, =\} \rightarrow \{E, N, =\}$ (i.e., a function on the 2-types of $G$).

Converse does not hold.

The following are all possibilities of canonical functions:

- Turning everything into edges ($e_E$)
- turning everything into non-edges ($e_N$)
- behaving like $-$
- being constant
- behaving like an automorphism.
Patterns in functions on the random graph

A canonical function $f : G \rightarrow G$ induces a function $f' : \{E, N, =\} \rightarrow \{E, N, =\}$ (i.e., a function on the 2-types of $G$).

Converse does not hold.

The following are all possibilities of canonical functions:

- Turning everything into edges ($e_E$)
- turning everything into non-edges ($e_N$)
- behaving like $-$
- being constant
- behaving like an automorphism.

Given any $f : G \rightarrow G$, we know that one of these behaviors appears for arbitrary finite subgraphs of $G$. 

Patterns in functions on the random graph

A canonical function $f : G \to G$ induces a function $f' : \{E, N, =\} \to \{E, N, =\}$ (i.e., a function on the 2-types of $G$).

Converse does not hold.

The following are all possibilities of canonical functions:

- Turning everything into edges ($e_E$)
- turning everything into non-edges ($e_N$)
- behaving like $-$
- being constant
- behaving like an automorphism.

Given any $f : G \to G$, we know that one of these behaviors appears for arbitrary finite subgraphs of $G$.

Problem: Keeping some information on $f$ when canonizing.
Adding constants

Let $f : G \to G$.

If $f$ violates a relation $R$, then there are $c_1, \ldots, c_n \in V$ witnessing this.
Adding constants

Let $f : G \to G$.
If $f$ violates a relation $R$, then there are $c_1, \ldots, c_n \in V$ witnessing this.

**Fact.**
The structure $(V; E, c_1, \ldots, c_n)$ has that Ramsey property, too.
Adding constants

Let \( f : G \to G \).

If \( f \) violates a relation \( R \), then there are \( c_1, \ldots, c_n \in V \) witnessing this.

**Fact.**
The structure \((V; E, c_1, \ldots, c_n)\) has that Ramsey property, too.

Consider \( f \) as a function from \((V; E, c_1, \ldots, c_n)\) to \((V; E)\).
Adding constants

Let $f : G \to G$.

If $f$ violates a relation $R$, then there are $c_1, \ldots, c_n \in V$ witnessing this.

**Fact.**
The structure $(V; E, c_1, \ldots, c_n)$ has that Ramsey property, too.

Consider $f$ as a function from $(V; E, c_1, \ldots, c_n)$ to $(V; E)$.
Again, $f$ is canonical on arbitrarily large finite substructures of $(V; E, c_1, \ldots, c_n)$. 
Adding constants

Let $f : G \to G$.

If $f$ violates a relation $R$, then there are $c_1, \ldots, c_n \in V$ witnessing this.

**Fact.**
The structure $(V; E, c_1, \ldots, c_n)$ has that Ramsey property, too.

Consider $f$ as a function from $(V; E, c_1, \ldots, c_n)$ to $(V; E)$.

Again, $f$ is canonical on arbitrarily large finite substructures of $(V; E, c_1, \ldots, c_n)$.

We can assume that it shows the *same* behavior on all these substructures.
Adding constants

Let $f : G \to G$.

If $f$ violates a relation $R$, then there are $c_1, \ldots, c_n \in V$ witnessing this.

**Fact.**
The structure $(V; E, c_1, \ldots, c_n)$ has that Ramsey property, too.

Consider $f$ as a function from $(V; E, c_1, \ldots, c_n)$ to $(V; E)$.

Again, $f$ is canonical on arbitrarily large finite substructures of $(V; E, c_1, \ldots, c_n)$.

We can assume that it shows the *same* behavior on all these substructures.

By topological closure, $f$ generates a function which:

- behaves like $f$ on $\{c_1, \ldots, c_n\}$, and
- is canonical as a function from $(V; E, c_1, \ldots, c_n)$ to $(V; E)$. 

The minimal clones on the random graph

Theorem (Bodirsky, MP ’10)

Let \( f \) be a finitary operation on \( G \) which “is” not an automorphism. Then \( f \) generates one of the following:

- A constant operation
- \( e_E \)
- \( e_N \)
- \(-\)
- \( sw_c \)
- One of 9 canonical binary injections.
The minimal clones on the random graph

Theorem (Bodirsky, MP ’10)

Let $f$ be a finitary operation on $G$ which "is" not an automorphism. Then $f$ generates one of the following:

- A constant operation
- $e_E$
- $e_N$
- $-$
- $\text{sw}_c$
- One of 9 canonical binary injections.

We thus know the minimal closed clones containing $\text{Aut}(G)$.
The minimal clones on the random graph

**Theorem (Bodirsky, MP ’10)**

Let $f$ be a finitary operation on $G$ which “is” not an automorphism. Then $f$ generates one of the following:

- A constant operation
- $e_E$
- $e_N$
- $\bar{1}$
- $\text{sw}_c$
- One of 9 canonical binary injections.

We thus know the *minimal closed clones* containing $\text{Aut}(G)$.

**More involved argument:** Extend $G$ by a random dense linear order.
Let $S, H, P$ be structures in the same signature $\tau$.

$S \rightarrow (H)P$ means:

For any coloring of the copies of $P$ in $S$ with 2 colors there exists a copy of $H$ in $S$ such that the copies of $P$ in $H$ all have the same color.

Definition

A class $C$ of $\tau$-structures is called a Ramsey class iff for all $H, P \in C$ there exists $S \in C$ such that $S \rightarrow (H)P$.
Let $S, H, P$ be structures in the same signature $\tau$.

$$S \rightarrow (H)^P$$

means:
Ramsey classes

Let $S, H, P$ be structures in the same signature $\tau$.

\[ S \rightarrow (H)^P \]

means:

For any coloring of the copies of $P$ in $S$ with 2 colors there exists a copy of $H$ in $S$ such that the copies of $P$ in $H$ all have the same color.
Ramsey classes

Let $S, H, P$ be structures in the same signature $\tau$. 

$$S \to (H)^P$$

means:

For any coloring of the copies of $P$ in $S$ with 2 colors there exists a copy of $H$ in $S$ such that the copies of $P$ in $H$ all have the same color.

Definition

A class $\mathcal{C}$ of $\tau$-structures is called a Ramsey class iff for all $H, P \in \mathcal{C}$ there exists $S$ in $\mathcal{C}$ such that $S \to (H)^P$. 
Let $\Delta$ now be an arbitrary structure.

Definition: $f : \Delta \to \Delta$ is canonical iff for all tuples $(x_1, \ldots, x_n)$, $(y_1, \ldots, y_n)$ of the same type, $(f(x_1), \ldots, f(x_n))$ and $(f(y_1), \ldots, f(y_n))$ have the same type too.

Observation. If $\Delta$ is Ramsey ordered $\omega$-categorical, then all finite substructures of $\Delta$ have a copy in $\Delta$ on which $f$ is canonical.

Thus: If $\Delta$ is in addition homogeneous in a finite language, then any $f : \Delta \to \Delta$ generates a canonical function, but it could be the identity.
Canonical functions on Ramsey structures

Let $\Delta$ now be an arbitrary structure.

**Definition**

$f : \Delta \to \Delta$ is *canonical* iff

for all tuples $(x_1, \ldots, x_n), (y_1, \ldots, y_n)$ of the same type

$(f(x_1), \ldots, f(x_n))$ and $(f(y_1), \ldots, f(y_n))$ have the same type too.
Canonical functions on Ramsey structures

Let $\Delta$ now be an arbitrary structure.

Definition

$f : \Delta \to \Delta$ is canonical iff for all tuples $(x_1, \ldots, x_n), (y_1, \ldots, y_n)$ of the same type $(f(x_1), \ldots, f(x_n))$ and $(f(y_1), \ldots, f(y_n))$ have the same type too.

Observation. If $\Delta$ is

- Ramsey
- ordered
- $\omega$-categorical,

then all finite substructures of $\Delta$ have a copy in $\Delta$ on which $f$ is canonical.
Let $\Delta$ now be an arbitrary structure.

**Definition**

$f : \Delta \to \Delta$ is *canonical* iff for all tuples $(x_1, \ldots, x_n), (y_1, \ldots, y_n)$ of the same type $(f(x_1), \ldots, f(x_n))$ and $(f(y_1), \ldots, f(y_n))$ have the same type too.

**Observation.** If $\Delta$ is

- Ramsey
- ordered
- $\omega$-categorical,

then all finite substructures of $\Delta$ have a copy in $\Delta$ on which $f$ is canonical.

**Thus:** If $\Delta$ is in addition *homogeneous* in a finite language, then any $f : \Delta \to \Delta$ generates a canonical function,
Canonical functions on Ramsey structures

Let $\Delta$ now be an arbitrary structure.

**Definition**

$f : \Delta \to \Delta$ is *canonical* iff for all tuples $(x_1, \ldots, x_n), (y_1, \ldots, y_n)$ of the same type $(f(x_1), \ldots, f(x_n))$ and $(f(y_1), \ldots, f(y_n))$ have the same type too.

**Observation.** If $\Delta$ is

- Ramsey
- ordered
- $\omega$-categorical,

then all finite substructures of $\Delta$ have a copy in $\Delta$ on which $f$ is canonical.

**Thus:** If $\Delta$ is in addition homogeneous in a finite language, then any $f : \Delta \to \Delta$ generates a canonical function, but it could be the identity.
What we would like to do...

We would like to fix $c_1, ..., c_n \in \Delta$ witnessing that $f$ does something interesting (e.g., violate a certain relation), and have canonical behavior of $f$ as a function from $(\Delta, c_1, ..., c_n)$ to $\Delta$.

Why don't you just do it?
What we would like to do...

We would like to fix $c_1, \ldots, c_n \in \Delta$ witnessing that $f$ does something interesting (e.g., violate a certain relation), and have canonical behavior of $f$ as a function from $(\Delta, c_1, \ldots, c_n)$ to $\Delta$. 

Why don't you just do it?

Michael Pinsker (Paris 7)
What we would like to do...

We would like to fix $c_1, \ldots, c_n \in \Delta$ witnessing that $f$ does something interesting (e.g., violate a certain relation), and have canonical behavior of $f$ as a function from $(\Delta, c_1, \ldots, c_n)$ to $\Delta$.

Why don’t you just do it?
Adding constants to Ramsey structures

Problem

If $\Delta$ is Ramsey, is $(\Delta, c_1, \ldots, c_n)$ still Ramsey?

Theorem (Kechris, Pestov, Todorcevic '05)

An ordered homogeneous structure is Ramsey iff its automorphism group is extremely amenable, i.e., it has a fixed point whenever it acts on a compact Hausdorff space.

Observation

Every open subgroup of an extremely amenable group is extremely amenable.

Corollary

If $\Delta$ is ordered, homogeneous, and Ramsey, then so is $(\Delta, c_1, \ldots, c_n)$.
Adding constants to Ramsey structures

**Problem**
If $\Delta$ is Ramsey, is $(\Delta, c_1, \ldots, c_n)$ still Ramsey?

**Theorem (Kechris, Pestov, Todorcevic ’05)**
An ordered homogeneous structure is Ramsey iff its automorphism group is *extremely amenable*, i.e., it has a fixed point whenever it acts on a compact Hausdorff space.
Adding constants to Ramsey structures

Problem
If $\Delta$ is Ramsey, is $(\Delta, c_1, \ldots, c_n)$ still Ramsey?

Theorem (Kechris, Pestov, Todorcevic ’05)
An ordered homogeneous structure is Ramsey iff its automorphism group is extremely amenable, i.e., it has a fixed point whenever it acts on a compact Hausdorff space.

Observation
Every open subgroup of an extremely amenable group is extremely amenable.
Adding constants to Ramsey structures

Problem
If $\Delta$ is Ramsey, is $(\Delta, c_1, \ldots, c_n)$ still Ramsey?

Theorem (Kechris, Pestov, Todorcevic ’05)
An ordered homogeneous structure is Ramsey iff its automorphism group is extremely amenable, i.e., it has a fixed point whenever it acts on a compact Hausdorff space.

Observation
Every open subgroup of an extremely amenable group is extremely amenable.

Corollary
If $\Delta$ is ordered, homogeneous, and Ramsey, then so is $(\Delta, c_1, \ldots, c_n)$.
Proposition

If $\Delta$ is ordered Ramsey homogeneous finite language, $f : \Delta^k \to \Delta$, and $c_1, \ldots, c_n \in \Delta$, then $f$ generates a function which

- is canonical as a function from $(\Delta, c_1, \ldots, c_n)^k$ to $\Delta$
- behaves like $f$ on $\{c_1, \ldots, c_n\}$.
Proposition (new proof at Fields, July 2011!)

If $\Delta$ is ordered Ramsey homogeneous finite language, $f : \Delta^k \rightarrow \Delta$, and $c_1, \ldots, c_n \in \Delta$, then $f$ generates a function which

- is canonical as a function from $(\Delta, c_1, \ldots, c_n)^k$ to $\Delta$
- behaves like $f$ on $\{c_1, \ldots, c_n\}$.

Set $S := \{g : \Delta^k \rightarrow \Delta \mid g \text{ agrees with } f \text{ on } \{c_1, \ldots, c_n\}\}$. 

\[ \text{Set } S := \{g : \Delta^k \rightarrow \Delta \mid g \text{ agrees with } f \text{ on } \{c_1, \ldots, c_n\}\}. \]
Proposition (new proof at Fields, July 2011!)

If $\Delta$ is ordered Ramsey homogeneous finite language, $f : \Delta^k \to \Delta$, and $c_1, \ldots, c_n \in \Delta$, then $f$ generates a function which

- is canonical as a function from $(\Delta, c_1, \ldots, c_n)^k$ to $\Delta$
- behaves like $f$ on $\{c_1, \ldots, c_n\}$.

Set $S := \{g : \Delta^k \to \Delta \mid g$ agrees with $f$ on $\{c_1, \ldots, c_n\}\}$. Set $g \sim h$ iff there is $\alpha \in \text{Aut}(\Delta)$ such that $g = \alpha h$. 

□
Proposition (new proof at Fields, July 2011!)

If $\Delta$ is ordered Ramsey homogeneous finite language, $f : \Delta^k \rightarrow \Delta$, and $c_1, \ldots, c_n \in \Delta$, then $f$ generates a function which

- is canonical as a function from $(\Delta, c_1, \ldots, c_n)^k$ to $\Delta$
- behaves like $f$ on $\{c_1, \ldots, c_n\}$.

Set $S := \{g : \Delta^k \rightarrow \Delta \mid g$ agrees with $f$ on $\{c_1, \ldots, c_n\}\}$. Set $g \sim h$ iff there is $\alpha \in \text{Aut}(\Delta)$ such that $g = \alpha h$.

Fact. $S/\sim$ is compact.
Canonizing functions on Ramsey structures

Proposition (new proof at Fields, July 2011!)
If $\Delta$ is ordered Ramsey homogeneous finite language, $f: \Delta^k \to \Delta$, and $c_1, \ldots, c_n \in \Delta$, then $f$ generates a function which
- is canonical as a function from $(\Delta, c_1, \ldots, c_n)^k$ to $\Delta$
- behaves like $f$ on $\{c_1, \ldots, c_n\}$.

Set $S := \{g: \Delta^k \to \Delta \mid g$ agrees with $f$ on $\{c_1, \ldots, c_n\}\}$. 

Set $g \sim h$ iff there is $\alpha \in \text{Aut}(\Delta)$ such that $g = \alpha h$.

Fact. $S/ \sim$ is compact.

Let $\text{Aut}(\Delta, c_1, \ldots, c_n)^k$ act on $S/ \sim$ by
\[
(\beta_1, \ldots, \beta_k)([g(x_1, \ldots, x_k)]\sim) := [g(\beta_1(x_1), \ldots, \beta_k(x_k))]\sim
\]
Canonizing functions on Ramsey structures

**Proposition (new proof at Fields, July 2011!)**

If $\Delta$ is ordered Ramsey homogeneous finite language, $f : \Delta^k \to \Delta$, and $c_1, \ldots, c_n \in \Delta$, then $f$ generates a function which

- is canonical as a function from $(\Delta, c_1, \ldots, c_n)^k$ to $\Delta$
- behaves like $f$ on $\{c_1, \ldots, c_n\}$.

Set $S := \{g : \Delta^k \to \Delta \mid g$ agrees with $f$ on $\{c_1, \ldots, c_n\}\}$. 

Set $g \sim h$ iff there is $\alpha \in \text{Aut}(\Delta)$ such that $g = \alpha h$.

**Fact.** $S/ \sim$ is compact.

Let $\text{Aut}(\Delta, c_1, \ldots, c_n)^k$ act on $S/ \sim$ by

$$(\beta_1, \ldots, \beta_k)([g(x_1, \ldots, x_k)]_{\sim}) := [g(\beta_1(x_1), \ldots, \beta_k(x_k))]_{\sim}$$

The continuous action has a fixed point $[h(x_1, \ldots, x_k)]_{\sim}$. 

Michael Pinsker (Paris 7)
Canonizing functions on Ramsey structures

Proposition (new proof at Fields, July 2011!)

If $\Delta$ is ordered Ramsey homogeneous finite language, $f : \Delta^k \rightarrow \Delta$, and $c_1, \ldots, c_n \in \Delta$, then $f$ generates a function which

- is canonical as a function from $(\Delta, c_1, \ldots, c_n)^k$ to $\Delta$
- behaves like $f$ on $\{c_1, \ldots, c_n\}$.

Set $S := \{g : \Delta^k \rightarrow \Delta \mid g$ agrees with $f$ on $\{c_1, \ldots, c_n\}\}$.

Set $g \sim h$ iff there is $\alpha \in \text{Aut}(\Delta)$ such that $g = \alpha h$.

Fact. $S/ \sim$ is compact.

Let $\text{Aut}(\Delta, c_1, \ldots, c_n)^k$ act on $S/ \sim$ by

$$(\beta_1, \ldots, \beta_k)([g(x_1, \ldots, x_k)] \sim) := [g(\beta_1(x_1), \ldots, \beta_k(x_k))] \sim$$

The continuous action has a fixed point $[h(x_1, \ldots, x_k)] \sim$.

Any element of the fixed point is canonical. \qed
Theorem (Bodirsky, MP, Tsankov ’10)

Let $\Gamma$ be a reduct of a finite language homogeneous ordered Ramsey structure $\Delta$. Then:

- Every minimal closed superclone of $\text{Pol}(\Gamma)$ is generated by such a canonical function.
- If $\Gamma$ has a finite language, then there are finitely many minimal closed superclones of $\text{Pol}(\Gamma)$.
- Every closed superclone of $\text{Pol}(\Gamma)$ contains a minimal closed superclone of $\text{Pol}(\Gamma)$. 
Minimal clones above Ramsey structures

Theorem (Bodirsky, MP, Tsankov ’10)

Let $\Gamma$ be a reduct of a finite language homogeneous ordered Ramsey structure $\Delta$. Then:

- Every minimal closed superclone of $\text{Pol}(\Gamma)$ is generated by such a canonical function.
Minimal clones above Ramsey structures

Theorem (Bodirsky, MP, Tsankov ’10)

Let $\Gamma$ be a reduct of a finite language homogeneous ordered Ramsey structure $\Delta$. Then:

- Every minimal closed superclone of $\text{Pol}(\Gamma)$ is generated by such a canonical function.
- If $\Gamma$ has a finite language, then there are finitely many minimal closed superclones of $\text{Pol}(\Gamma)$. (Arity bound!)
Theorem (Bodirsky, MP, Tsankov ’10)

Let \( \Gamma \) be a reduct of a finite language homogeneous ordered Ramsey structure \( \Delta \). Then:

- Every minimal closed superclone of \( \text{Pol}(\Gamma) \) is generated by such a canonical function.
- If \( \Gamma \) has a finite language, then there are finitely many minimal closed superclones of \( \text{Pol}(\Gamma) \).
  (Arity bound!)
- Every closed superclone of \( \text{Pol}(\Gamma) \) contains a minimal closed superclone of \( \text{Pol}(\Gamma) \).
The Graph-SAT dichotomy visualized

Making the infinite finite

Michael Pinsker  (Paris 7)
The Graph-SAT dichotomy in more detail

Theorem (Bodirsky, MP '10)
Let $\Gamma$ be a reduct of the random graph. Then:
Either $\Gamma$ has one out of 17 canonical polymorphisms, and CSP($\Gamma$) is tractable,
or CSP($\Gamma$) is NP-complete.

Theorem (Bodirsky, MP '10)
Let $\Gamma$ be a reduct of the random graph. Then:
Either $\Gamma$ pp-defines one out of 4 hard relations, and CSP($\Gamma$) is NP-complete,
or CSP($\Gamma$) is tractable.
Theorem (Bodirsky, MP ’10)

Let $\Gamma$ be a reduct of the random graph. Then:

- Either $\Gamma$ has one out of 17 canonical polymorphisms, and $\text{CSP}(\Gamma)$ is tractable,
- or $\text{CSP}(\Gamma)$ is NP-complete.
The Graph-SAT dichotomy in more detail

Theorem (Bodirsky, MP ’10)
Let \( \Gamma \) be a reduct of the random graph. Then:

- Either \( \Gamma \) has one out of 17 canonical polymorphisms, and CSP(\( \Gamma \)) is tractable,
- or CSP(\( \Gamma \)) is NP-complete.

Theorem (Bodirsky, MP ’10)
Let \( \Gamma \) be a reduct of the random graph. Then:

- Either \( \Gamma \) pp-defines one out of 4 hard relations, and CSP(\( \Gamma \)) is NP-complete,
- or CSP(\( \Gamma \)) is tractable.
Theorem

The following 17 distinct clones are precisely the minimal tractable local clones containing $\text{Aut}(G)$:

1. The clone generated by a constant operation.
2. The clone generated by a balanced binary injection of type max.
3. The clone generated by a balanced binary injection of type min.
4. The clone generated by an $E$-dominated binary injection of type max.
5. The clone generated by an $N$-dominated binary injection of type min.
6. The clone generated by a function of type majority which is hyperplanely balanced and of type projection.
7. The clone generated by a function of type majority which is hyperplanely $E$-constant.
8. The clone generated by a function of type majority which is hyperplanely $N$-constant.
9. The clone generated by a function of type majority which is hyperplanely of type max and $E$-dominated.
10. The clone generated by a function of type majority which is hyperplanely of type min and $N$-dominated.
11. The clone generated by a function of type minority which is hyperplanely balanced and of type projection.
12. The clone generated by a function of type minority which is hyperplanely of type projection and $E$-dominated.
13. The clone generated by a function of type minority which is hyperplanely of type projection and $N$-dominated.
14. The clone generated by a function of type minority which is hyperplanely of type $\text{xnor}$ and $E$-dominated.
15. The clone generated by a function of type minority which is hyperplanely of type $\text{xor}$ and $N$-dominated.
16. The clone generated by a binary injection which is $E$-constant.
17. The clone generated by a binary injection which is $N$-constant.
The Meta Problem

Meta-Problem of Graph-SAT$(\Psi)$

**INPUT:** A finite set $\Psi$ of graph formulas.

**QUESTION:** Is Graph-SAT$(\Psi)$ in P?

**Theorem (Bodirsky, MP '10)**

The Meta-Problem of Graph-SAT$(\Psi)$ is decidable.

Making the infinite finite

Michael Pinsker (Paris 7)
The Meta Problem

Meta-Problem of Graph-SAT(ψ)

INPUT: A finite set $\psi$ of graph formulas.

QUESTION: Is Graph-SAT($\psi$) in P?

Theorem (Bodirsky, MP '10)

The Meta-Problem of Graph-SAT($\psi$) is decidable.

Making the infinite finite

Michael Pinsker (Paris 7)
The Meta Problem

Meta-Problem of Graph-SAT(\(\Psi\))

INPUT: A finite set \(\Psi\) of graph formulas.

QUESTION: Is Graph-SAT(\(\Psi\)) in P?

Theorem (Bodirsky, MP ’10)

The Meta-Problem of Graph-SAT(\(\Psi\)) is decidable.
Part IV
The past and the future
The Past: What we can do

Climb up the clone lattice

Violate (hard) relations canonically

Decide pp definability:

Theorem (Bodirsky, MP, Tsankov '10)

Let $\Delta$ be ordered Ramsey homogeneous with finite language finitely bounded. Then the following problem is decidable:

INPUT: Two finite language reducts $\Gamma_1, \Gamma_2$ of $\Delta$.

QUESTION: Is $\Gamma_1$ primitive positive definable in $\Gamma_2$?
The Past: What we can do

- Climb up the clone lattice

Theorem (Bodirsky, MP, Tsankov ’10)

Let $\Delta$ be ordered Ramsey homogeneous with finite language finitely bounded. Then the following problem is decidable:

INPUT: Two finite language reducts $\Gamma_1, \Gamma_2$ of $\Delta$.

QUESTION: Is $\Gamma_1$ primitive positive definable in $\Gamma_2$?
The Past: What we can do

- Climb up the clone lattice
- Violate (hard) relations canonically

Theorem (Bodirsky, MP, Tsankov '10)

Let $\Delta$ be ordered Ramsey homogeneous with finite language finitely bounded. Then the following problem is decidable:

**INPUT:** Two finite language reducts $\Gamma_1, \Gamma_2$ of $\Delta$.

**QUESTION:** Is $\Gamma_1$ primitive positive definable in $\Gamma_2$?
The Past: What we can do

- Climb up the clone lattice
- Violate (hard) relations canonically
- Decide pp definability:

Theorem (Bodirsky, MP, Tsankov '10)

Let $\Delta$ be ordered Ramsey homogeneous with finite language finitely bounded.

Then the following problem is decidable:

INPUT: Two finite language reducts $\Gamma_1, \Gamma_2$ of $\Delta$.

QUESTION: Is $\Gamma_1$ primitive positive definable in $\Gamma_2$?
The Past: What we can do

- Climb up the clone lattice
- Violate (hard) relations canonically
- Decide pp definability:

Theorem (Bodirsky, MP, Tsankov ’10)

Let $\Delta$ be

- ordered Ramsey
- homogeneous
- with finite language
- finitely bounded.

Then the following problem is decidable:

INPUT: Two finite language reducts $\Gamma_1, \Gamma_2$ of $\Delta$.

QUESTION: Is $\Gamma_1$ primitive positive definable in $\Gamma_2$?
The Future

Making the infinite finite

Michael Pinsker (Paris 7)
The Future

Generalize setting of method

Is every structure $\Delta$ which is

- homogeneous
- with finite language
- finitely bounded

a reduct of a structure $\Delta'$ which is

- ordered Ramsey
- homogeneous
- with finite language
- finitely bounded.

?
Making the infinite finite

Michael Pinsker  (Paris 7)
The Future

Apply method

- Random partial order
- Random tournament
- Random $K_n$-free graph
- Atomless Boolean algebra
- Random lattice
The Future

Making the infinite finite

Michael Pinsker (Paris 7)
Develop method

Abstract cloning → Manuel’s talk
Making the infinite finite

Michael Pinsker  (Paris 7)
Making the infinite finite
Michael Pinsker (Paris 7)
Making the infinite finite

Michael Pinsker (Paris 7)
Making the infinite finite

Michael Pinsker (Paris 7)
Making the infinite finite

Michael Pinsker  (Paris 7)
Making the infinite finite

Michael Pinsker (Paris 7)
| Making the infinite finite | Michael Pinsker  (Paris 7) |
THANK YOU