Cooking

with model theory, universal algebra and Ramsey theory

in the complexity theory kitchen

Michael Pinsker

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

Yeshiva University, October 2011

Outline

- **Part I**

  Graph-SAT problems

- **Part II**

  Making the finite infinite

  Homogeneous structures

- **Part III**

  Making the infinite finite

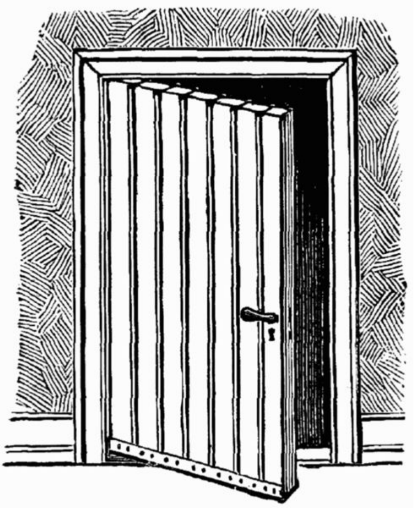
  Ramsey theory

- **Part IV**

  The Graph-SAT dichotomy

- **Part V**

  The future



Part I

Graph-SAT problems

The Boolean satisfiability problem

The Boolean satisfiability problem

Let Ψ be a finite set of propositional formulas.

The Boolean satisfiability problem

Let Ψ be a finite set of propositional formulas.

Computational problem: Boolean-SAT(Ψ)

INPUT:

- A set W of propositional variables, and
- statements $\phi_1, \ldots, \phi_n$ about the variables in W, where each ϕ_i is taken from Ψ.

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

The Boolean satisfiability problem

Let Ψ be a finite set of propositional formulas.

Computational problem: Boolean-SAT(Ψ)

INPUT:

- A set W of propositional variables, and
- statements $\phi_1, \ldots, \phi_n$ about the variables in W, where each ϕ_i is taken from Ψ.

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

Computational complexity depends on Ψ. Always in NP.

The Boolean satisfiability problem

Let Ψ be a finite set of propositional formulas.

Computational problem: Boolean-SAT(Ψ)

INPUT:

- A set W of propositional variables, and
- statements $\phi_1, \ldots, \phi_n$ about the variables in W, where each ϕ_i is taken from Ψ.

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

Computational complexity depends on Ψ. Always in NP.

Theorem (Schaefer '78)

Boolean-SAT(Ψ) is either in P or NP-complete, for all Ψ.

The Boolean satisfiability problem

Let Ψ be a finite set of propositional formulas.

Computational problem: Boolean-SAT(Ψ)

INPUT:

- A set W of propositional variables, and
- statements $\phi_1, \ldots, \phi_n$ about the variables in W, where each ϕ_i is taken from Ψ.

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

Computational complexity depends on Ψ. Always in NP.

Theorem (Schaefer '78) 1133 citations on google scholar

Boolean-SAT(Ψ) is either in P or NP-complete, for all Ψ.

The Graph Satisfiability Problem

The Graph Satisfiability Problem

Let E be a binary relation symbol.

(Imagine: edge relation of an undirected graph.)

Let Ψ 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 Ψ be a finite set of quantifier-free $\{E\}$-formulas.

Computational problem: Graph-SAT(Ψ)

INPUT:

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

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 Ψ be a finite set of quantifier-free $\{E\}$-formulas.

Computational problem: Graph-SAT(Ψ)

INPUT:

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

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

Computational complexity depends on Ψ. Always in NP.

The Graph Satisfiability Problem

Let E be a binary relation symbol.

(Imagine: edge relation of an undirected graph.)

Let Ψ be a finite set of quantifier-free $\{E\}$-formulas.

Computational problem: Graph-SAT(Ψ)

INPUT:

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

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

Computational complexity depends on Ψ. Always in NP.

Question

For which Ψ is Graph-SAT(Ψ) tractable?

Graph-SAT: Examples

Graph-SAT: Examples

Example 1 Let Ψ_1 only contain

$$\begin{aligned}\psi_1(x,y,z) :=& (E(x,y) \wedge \neg E(y,z) \wedge \neg E(x,z)) \\ &\vee (\neg E(x,y) \wedge E(y,z) \wedge \neg E(x,z)) \\ &\vee (\neg E(x,y) \wedge \neg E(y,z) \wedge E(x,z)) \ .\end{aligned}$$

Graph-SAT: Examples

Example 1 Let Ψ_1 only contain

$$\begin{aligned}\psi_1(x, y, z) :=& (E(x, y) \wedge \neg E(y, z) \wedge \neg E(x, z)) \\ & \vee (\neg E(x, y) \wedge E(y, z) \wedge \neg E(x, z)) \\ & \vee (\neg E(x, y) \wedge \neg E(y, z) \wedge E(x, z)) \,.\end{aligned}$$

Graph-SAT(Ψ_1) is NP-complete.

Graph-SAT: Examples

Example 1 Let Ψ_1 only contain

$$\begin{aligned}\psi_1(x,y,z) :=&(E(x,y) \wedge \neg E(y,z) \wedge \neg E(x,z))\\ &\vee (\neg E(x,y) \wedge E(y,z) \wedge \neg E(x,z))\\ &\vee (\neg E(x,y) \wedge \neg E(y,z) \wedge E(x,z))\ .\end{aligned}$$

Graph-SAT(Ψ_1) is NP-complete.

Example 2 Let Ψ_2 only contain

$$\begin{aligned}\psi_2(x,y,z) :=&(E(x,y) \wedge \neg E(y,z) \wedge \neg E(x,z))\\ &\vee (\neg E(x,y) \wedge E(y,z) \wedge \neg E(x,z))\\ &\vee (\neg E(x,y) \wedge \neg E(y,z) \wedge E(x,z))\\ &\vee (E(x,y) \wedge E(y,z) \wedge E(x,z))\ .\end{aligned}$$

Graph-SAT: Examples

Example 1 Let Ψ_1 only contain

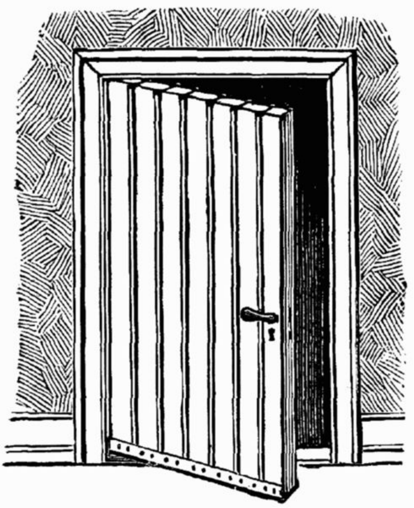
$$\begin{aligned}\psi_1(x,y,z) :=& (E(x,y) \wedge \neg E(y,z) \wedge \neg E(x,z)) \\ & \vee (\neg E(x,y) \wedge E(y,z) \wedge \neg E(x,z)) \\ & \vee (\neg E(x,y) \wedge \neg E(y,z) \wedge E(x,z)) \ .\end{aligned}$$

Graph-SAT(Ψ_1) is NP-complete.

Example 2 Let Ψ_2 only contain

$$\begin{aligned}\psi_2(x,y,z) :=& (E(x,y) \wedge \neg E(y,z) \wedge \neg E(x,z)) \\ & \vee (\neg E(x,y) \wedge E(y,z) \wedge \neg E(x,z)) \\ & \vee (\neg E(x,y) \wedge \neg E(y,z) \wedge E(x,z)) \\ & \vee (E(x,y) \wedge E(y,z) \wedge E(x,z)) \ .\end{aligned}$$

Graph-SAT(Ψ_2) is in P.



Part II

Making the finite infinite

(Homogeneous structures)

Graph formulas and reducts of the random graph

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

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 $\alpha \in \mathrm{Aut}(G)$ 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 $i : A \to B$ there exists $\alpha \in \mathrm{Aut}(G)$ extending i.
- universal, i.e., contains all finite (even countable) graphs.

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

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

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 $\alpha \in \mathrm{Aut}(G)$ 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 Ψ 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 $\alpha \in \mathrm{Aut}(G)$ 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 Ψ of graph formulas, define a structure

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

Γ_Ψ is a *reduct of* the random graph, i.e.,
a structure with a first-order definition in G.

Graph-SAT as CSP

Graph-SAT as CSP

An instance

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

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

Graph-SAT as CSP

An instance

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

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

The decision problem
whether or not a given primitive positive sentence holds in Γ_Ψ
is called the Constraint Satisfaction Problem of Γ_Ψ (or CSP(Γ_Ψ)).

Graph-SAT as CSP

An instance

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

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

The decision problem
whether or not a given primitive positive sentence holds in Γ_Ψ
is called the Constraint Satisfaction Problem of Γ_Ψ (or CSP(Γ_Ψ)).

So Graph-SAT(Ψ) and CSP(Γ_Ψ) are one and the same problem.

Why the random graph?

Why the random graph?

We have seen:

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

Why the random graph?

We have seen:

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

Note:

Could have used any universal graph!

Why the random graph?

We have seen:

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

Note:

Could have used any universal graph!

But:

The random graph is the nicest universal graph.

Why the random graph?

We have seen:

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

Note:

Could have used any universal graph!

But:

The random graph is the nicest universal graph.

Let's study reducts of homogeneous structures!

Reducts of homogeneous structures

Let Δ be a countable relational structure
in a finite language which is *homogeneous*, i.e.,

Reducts of homogeneous structures

Let Δ be a countable relational structure in a finite language which is *homogeneous*, i.e.,

For all $A, B \subseteq \Delta$ finite, for all isomorphisms $i : A \to B$ there exists $\alpha \in \mathrm{Aut}(\Delta)$ extending i.

Reducts of homogeneous structures

Let Δ be a countable relational structure
in a finite language which is *homogeneous*, i.e.,

For all $A, B \subseteq \Delta$ finite, for all isomorphisms $i : A \to B$
there exists $\alpha \in \mathrm{Aut}(\Delta)$ extending i.

Definition

A *reduct* of Δ is a structure with a first-order (fo) definition in Δ.

Reducts of homogeneous structures

Let Δ be a countable relational structure
in a finite language which is *homogeneous*, i.e.,

For all $A, B \subseteq \Delta$ finite, for all isomorphisms $i : A \to B$
there exists $\alpha \in \mathrm{Aut}(\Delta)$ extending i.

Definition

A *reduct* of Δ is a structure with a first-order (fo) definition in Δ.

Problem

Classify the reducts of Δ.

We call Δ the *base structure*.

Classifications up to first-order interdefinability

Classifications up to first-order interdefinability

One possibility of classification:

We can consider two reducts Γ, Γ' of Δ *equivalent* iff
Γ has a fo-definition from Γ' and vice-versa.

One possibility of classification:

We can consider two reducts Γ, Γ' of Δ *equivalent* iff Γ has a fo-definition from Γ' and vice-versa.

We say that Γ and Γ' are fo-interdefinable.

Classifications up to first-order interdefinability

One possibility of classification:

We can consider two reducts Γ, Γ' of Δ *equivalent* iff
Γ has a fo-definition from Γ' and vice-versa.

We say that Γ and Γ' are fo-interdefinable.

The relation "Γ is fo-definable in Γ'" is a quasiorder on the reducts.

Classifications up to first-order interdefinability

One possibility of classification:

We can consider two reducts Γ, Γ' of Δ *equivalent* iff
Γ has a fo-definition from Γ' and vice-versa.

We say that Γ and Γ' are fo-interdefinable.

The relation "Γ is fo-definable in Γ'" is a quasiorder on the reducts.

We factor this quasiorder
by the equivalence relation of fo-interdefinability,
and obtain a complete lattice.

Classifications up to pp interdefinability

Another possibility of classification:

Classifications up to pp interdefinability

Another possibility of classification:

A formula is primitive positive (pp) iff
it is of the form $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$, for atomic ϕ_i.

Classifications up to pp interdefinability

Another possibility of classification:

A formula is primitive positive (pp) iff
it is of the form $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$, for atomic ϕ_i.

Can consider reducts Γ, Γ' equivalent iff
Γ has a *pp*-definition from Γ' and vice-versa.

Classifications up to pp interdefinability

Another possibility of classification:

A formula is primitive positive (pp) iff
it is of the form $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$, for atomic ϕ_i.

Can consider reducts Γ, Γ' equivalent iff
Γ has a *pp*-definition from Γ' and vice-versa.

The relation "Γ is *pp*-definable in Γ'" is a quasiorder on the reducts.

Classifications up to pp interdefinability

Another possibility of classification:

A formula is primitive positive (pp) iff
it is of the form $\exists w_1, \ldots, w_m. \bigwedge_i \phi_i$, for atomic ϕ_i.

Can consider reducts Γ, Γ' equivalent iff
Γ has a *pp*-definition from Γ' and vice-versa.

The relation "Γ is *pp*-definable in Γ'" is a quasiorder on the reducts.

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

Comparing the classifications

Comparing the classifications

Observe:

Primitive positive (pp) interdefinability is finer than first order (fo) interdefinability.

Comparing the classifications

Observe:

Primitive positive (pp) interdefinability is finer than first order (fo) interdefinability.

In fact:

The lattice corresponding to fo-definability is a factor of the lattice corresponding to pp-definability.

What is interesting?

What is interesting?

Which of the two lattices is interesting?

What is interesting?

Which of the two lattices is interesting?

Model theorists: First order!

What is interesting?

Which of the two lattices is interesting?

Model theorists: First order!

Complexity theorists: Primitive positive!

What is interesting?

Which of the two lattices is interesting?

Model theorists: First order!

Complexity theorists: Primitive positive!

Explanation:

- Every reduct defines a computational problem (Constraint Satisfaction Problem).
- Reducts which are pp-interdefinable have polynomial time-equivalent computational complexity.

What is interesting?

Which of the two lattices is interesting?

Model theorists: First order!

Complexity theorists: Primitive positive!

Explanation:

- Every reduct defines a computational problem (Constraint Satisfaction Problem).
- Reducts which are pp-interdefinable have polynomial time-equivalent computational complexity.

This talk: Method for pp. Helps also for fo.

Why is Δ homogeneous in a finite language?

Question makes sense for arbitrary base structure Δ.

Why is Δ homogeneous in a finite language?

Question makes sense for arbitrary base structure Δ.

But homogeneity + finite language imply:

Why is Δ homogeneous in a finite language?

Question makes sense for arbitrary base structure Δ.

But homogeneity + finite language imply:

- *fo-closed* reducts correspond to *closed groups* $\supseteq \mathrm{Aut}(\Delta)$;

Why is Δ homogeneous in a finite language?

Question makes sense for arbitrary base structure Δ.

But homogeneity + finite language imply:

- *fo-closed* reducts correspond to *closed groups* $\supseteq \mathrm{Aut}(\Delta)$;
- *pp-closed reducts* correspond to *closed clones* $\supseteq \mathrm{Aut}(\Delta)$.

Why is Δ homogeneous in a finite language?

Question makes sense for arbitrary base structure Δ.

But homogeneity + finite language imply:

- *fo-closed* reducts correspond to *closed groups* $\supseteq \mathrm{Aut}(\Delta)$;
- *pp-closed reducts* correspond to *closed clones* $\supseteq \mathrm{Aut}(\Delta)$.

For our method, we will need even “more” than homogeneity in a finite language:

The Ramsey property

Example: The dense linear order

Example: The dense linear order

Denote by $(\mathbb{Q}; <)$ be the order of the rationals, and set

$$\begin{aligned}
\text{betw}(x,y,z) :=& \{(x,y,z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < y < x\} \\
\text{cycl}(x,y,z) :=& \{(x,y,z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < x < y \\
& \text{ or } y < z < x\} \\
\text{sep}(x,y,z,w) :=& \{(x,y,z,w) \in \mathbb{Q}^4 : \dots\}
\end{aligned}$$

Example: The dense linear order

Denote by $(\mathbb{Q}; <)$ be the order of the rationals, and set

$$\mathrm{betw}(x, y, z) := \{(x, y, z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < y < x\}$$
$$\mathrm{cycl}(x, y, z) := \{(x, y, z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < x < y \text{ or } y < z < x\}$$
$$\mathrm{sep}(x, y, z, w) := \{(x, y, z, w) \in \mathbb{Q}^4 : \ldots\}$$

Theorem (Cameron '76)

Let Γ be a reduct of $\Delta := (\mathbb{Q}; <)$. Then:

Example: The dense linear order

Denote by $(\mathbb{Q}; <)$ be the order of the rationals, and set

$$\begin{aligned} \text{betw}(x,y,z) &:= \{(x,y,z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < y < x\} \\ \text{cycl}(x,y,z) &:= \{(x,y,z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < x < y \\ &\qquad \text{or } y < z < x\} \\ \text{sep}(x,y,z,w) &:= \{(x,y,z,w) \in \mathbb{Q}^4 : \ldots\} \end{aligned}$$

Theorem (Cameron '76)

Let Γ be a reduct of $\Delta := (\mathbb{Q}; <)$. Then:

1. Γ is first-order interdefinable with $(\mathbb{Q}; <)$, or

Example: The dense linear order

Denote by $(\mathbb{Q}; <)$ be the order of the rationals, and set

$$\begin{aligned}
\text{betw}(x, y, z) :=& \{(x, y, z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < y < x\} \\
\text{cycl}(x, y, z) :=& \{(x, y, z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < x < y \\
& \text{or } y < z < x\} \\
\text{sep}(x, y, z, w) :=& \{(x, y, z, w) \in \mathbb{Q}^4 : \ldots\}
\end{aligned}$$

Theorem (Cameron '76)

Let Γ be a reduct of $\Delta := (\mathbb{Q}; <)$. Then:

1. Γ is first-order interdefinable with $(\mathbb{Q}; <)$, or
2. Γ is first-order interdefinable with $(\mathbb{Q}; \text{betw})$, or

Example: The dense linear order

Denote by $(\mathbb{Q}; <)$ be the order of the rationals, and set

$$\begin{aligned}
\text{betw}(x,y,z) &:= \{(x,y,z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < y < x\} \\
\text{cycl}(x,y,z) &:= \{(x,y,z) \in \mathbb{Q}^3 : x < y < z \text{ or } z < x < y \\
&\qquad \text{or } y < z < x\} \\
\text{sep}(x,y,z,w) &:= \{(x,y,z,w) \in \mathbb{Q}^4 : \ldots\}
\end{aligned}$$

Theorem (Cameron '76)

Let Γ be a reduct of $\Delta := (\mathbb{Q}; <)$. Then:

1. Γ is first-order interdefinable with $(\mathbb{Q}; <)$, or
2. Γ is first-order interdefinable with $(\mathbb{Q}; \text{betw})$, or
3. Γ is first-order interdefinable with $(\mathbb{Q}; \text{cycl})$, or

Example: The dense linear order

Denote by $(\mathbb{Q};<)$ be the order of the rationals, and set

$$\begin{aligned}\text{betw}(x,y,z) :=& \{(x,y,z)\in\mathbb{Q}^3 : x<y<z \text{ or } z<y<x\}\\ \text{cycl}(x,y,z) :=& \{(x,y,z)\in\mathbb{Q}^3 : x<y<z \text{ or } z<x<y\\ & \text{ or } y<z<x\}\\ \text{sep}(x,y,z,w) :=& \{(x,y,z,w)\in\mathbb{Q}^4 : \dots\}\end{aligned}$$

Theorem (Cameron '76)

Let Γ be a reduct of $\Delta := (\mathbb{Q};<)$. Then:

1. Γ is first-order interdefinable with $(\mathbb{Q};<)$, or
2. Γ is first-order interdefinable with $(\mathbb{Q};\text{betw})$, or
3. Γ is first-order interdefinable with $(\mathbb{Q};\text{cycl})$, or
4. Γ is first-order interdefinable with $(\mathbb{Q};\text{sep})$, or

Example: The dense linear order

Denote by $(\mathbb{Q};<)$ be the order of the rationals, and set

$$\begin{aligned}\text{betw}(x,y,z) &:= \{(x,y,z) \in \mathbb{Q}^3 : x<y<z \text{ or } z<y<x\}\\ \text{cycl}(x,y,z) &:= \{(x,y,z) \in \mathbb{Q}^3 : x<y<z \text{ or } z<x<y \\ &\qquad \text{ or } y<z<x\}\\ \text{sep}(x,y,z,w) &:= \{(x,y,z,w) \in \mathbb{Q}^4 : \dots\}\end{aligned}$$

Theorem (Cameron '76)

Let Γ be a reduct of $\Delta := (\mathbb{Q};<)$. Then:

1. Γ is first-order interdefinable with $(\mathbb{Q};<)$, or
2. Γ is first-order interdefinable with $(\mathbb{Q};\text{betw})$, or
3. Γ is first-order interdefinable with $(\mathbb{Q};\text{cycl})$, or
4. Γ is first-order interdefinable with $(\mathbb{Q};\text{sep})$, or
5. Γ is first-order interdefinable with $(\mathbb{Q};=)$.

Example: The random graph

Example: The random graph

Let $G = (V; E)$ be the random graph, and set for all $k \geq 2$

$$R^{(k)} := \{(x_1, \ldots, x_k) \subseteq V^k : x_i \text{ distinct, number of edges odd}\}.$$

Example: The random graph

Let $G = (V; E)$ be the random graph, and set for all $k \geq 2$

$$R^{(k)} := \{(x_1, \ldots, x_k) \subseteq V^k : x_i \text{ distinct, number of edges odd}\}.$$

Theorem (Thomas '91)

Let Γ be a reduct of $\Delta := G = (V; E)$. Then:

Example: The random graph

Let $G = (V; E)$ be the random graph, and set for all $k \geq 2$

$$R^{(k)} := \{(x_1, \ldots, x_k) \subseteq V^k : x_i \text{ distinct, number of edges odd}\}.$$

Theorem (Thomas '91)

Let Γ be a reduct of $\Delta := G = (V; E)$. Then:

1. Γ is first-order interdefinable with $(V; E)$, or

Example: The random graph

Let $G = (V; E)$ be the random graph, and set for all $k \geq 2$

$$R^{(k)} := \{(x_1, \ldots, x_k) \subseteq V^k : x_i \text{ distinct, number of edges odd}\}.$$

Theorem (Thomas '91)

Let Γ be a reduct of $\Delta := G = (V; E)$. Then:

1. Γ is first-order interdefinable with $(V; E)$, or
2. Γ is first-order interdefinable with $(V; R^{(3)})$, or

Example: The random graph

Let $G = (V; E)$ be the random graph, and set for all $k \geq 2$

$$R^{(k)} := \{(x_1, \ldots, x_k) \subseteq V^k : x_i \text{ distinct, number of edges odd}\}.$$

Theorem (Thomas '91)

Let Γ be a reduct of $\Delta := G = (V; E)$. Then:

1. Γ is first-order interdefinable with $(V; E)$, or
2. Γ is first-order interdefinable with $(V; R^{(3)})$, or
3. Γ is first-order interdefinable with $(V; R^{(4)})$, or

Example: The random graph

Let $G = (V; E)$ be the random graph, and set for all $k \geq 2$

$$R^{(k)} := \{(x_1, \ldots, x_k) \subseteq V^k : x_i \text{ distinct, number of edges odd}\}.$$

Theorem (Thomas '91)

Let Γ be a reduct of $\Delta := G = (V; E)$. Then:

1. Γ is first-order interdefinable with $(V; E)$, or
2. Γ is first-order interdefinable with $(V; R^{(3)})$, or
3. Γ is first-order interdefinable with $(V; R^{(4)})$, or
4. Γ is first-order interdefinable with $(V; R^{(5)})$, or

Example: The random graph

Let $G = (V; E)$ be the random graph, and set for all $k \geq 2$

$$R^{(k)} := \{(x_1, \ldots, x_k) \subseteq V^k : x_i \text{ distinct, number of edges odd}\}.$$

Theorem (Thomas '91)

Let Γ be a reduct of $\Delta := G = (V; E)$. Then:

1. Γ is first-order interdefinable with $(V; E)$, or
2. Γ is first-order interdefinable with $(V; R^{(3)})$, or
3. Γ is first-order interdefinable with $(V; R^{(4)})$, or
4. Γ is first-order interdefinable with $(V; R^{(5)})$, or
5. Γ is first-order interdefinable with $(V; =)$.

Further examples

Further examples

Theorem (Thomas '91)

The homogeneous K_n-free graph has 2 reducts up to fo-interdefinability.

Further examples

Theorem (Thomas '91)

The homogeneous K_n-free graph has 2 reducts up to fo-interdefinability.

Theorem (Thomas '96)

The homogeneous k-graph has $2^k + 1$ reducts up to fo-interdefinability.

Further examples

Theorem (Thomas '91)

The homogeneous K_n-free graph has 2 reducts up to fo-interdefinability.

Theorem (Thomas '96)

The homogeneous k-graph has $2^k + 1$ reducts up to fo-interdefinability.

Theorem (Junker, Ziegler '08)

$(\mathbb{Q}; <, 0)$ has 116 reducts up to fo-interdefinability.

Very recent examples

Very recent examples

Theorem (Pách, MP, Pluhár, Pongrácz, Szabó '11)

The random partial order has 5 reducts up to fo-interdefinability.

Very recent examples

Theorem (Pách, MP, Pluhár, Pongrácz, Szabó '11)

The random partial order has 5 reducts up to fo-interdefinability.

Theorem (Pongrácz '11)

The random K_n-free graph plus constant has 13 reducts if $n = 3$, and 16 reducts if $n \geq 4$ up to fo-interdefinability.

Very recent examples

Theorem (Pách, MP, Pluhár, Pongrácz, Szabó '11)

The random partial order has 5 reducts up to fo-interdefinability.

Theorem (Pongrácz '11)

The random K_n-free graph plus constant has 13 reducts if $n = 3$, and 16 reducts if $n \geq 4$ up to fo-interdefinability.

Depressing fact (Horváth, Pongrácz, MP '11)

The random graph with a constant has too many reducts up to fo-interdefinability.

Thomas' conjecture

Conjecture (Thomas '91)

Let Δ be homogeneous in a finite language.

Then Δ has finitely many reducts up to fo-interdefinability.

pp classifications

pp classifications

Theorem (Bodirsky, Chen, MP '08)

For the structure $\Delta := (X; =)$, there exist:

pp classifications

Theorem (Bodirsky, Chen, MP '08)

For the structure $\Delta := (X; =)$, there exist:

- 1 reduct up to first order interdefinability

pp classifications

Theorem (Bodirsky, Chen, MP '08)

For the structure $\Delta := (X; =)$, there exist:

- 1 reduct up to first order interdefinability
- $2^{\aleph_0}$ reducts up to primitive positive interdefinability

Permutation groups - fo

Permutation groups - fo

Theorem (Ryll-Nardzewski)

Let Δ be homogeneous, finite language.

The mapping

$$\Gamma \mapsto \mathrm{Aut}(\Gamma)$$

is a one-to-one correspondence between
the *first-order closed* reducts of Δ and
the *closed permutation groups* containing $\mathrm{Aut}(\Delta)$.

first order closed $=$ contains all fo-definable relations

group called closed iff it is closed in the convergence topology.

Clones - pp

Clones - pp

Theorem (Bodirsky, Nešetřil '03)

Let Δ be homogeneous, finite language. Then

$$\Gamma \mapsto \text{Pol}(\Gamma)$$

is a one-to-one correspondence between
the *primitive positive closed* reducts of Δ and
the *closed clones* containing $\text{Aut}(\Delta)$.

A clone is a set of finitary operations on Δ which

- contains all projections $\pi_i^n(x_1, \ldots, x_n) = x_i$, and
- is closed under composition.

$\text{Pol}(\Gamma)$ is the clone of all homomorphisms from finite powers of Γ to Γ.

A clone C is closed if for each $n \geq 1$, the set of n-ary operations in C is a closed subset of the Baire space Δ^{Δ^n}.

Groups and Clones

For homogeneous Δ in finite language:

Reducts up to **fo-interdefinability** $\leftrightarrow$

closed **permutation groups** $\supseteq \mathrm{Aut}(\Delta)$;

Reducts up to **pp-interdefinability** $\leftrightarrow$

closed **clones** $\supseteq \mathrm{Aut}(\Delta)$.

Groups and Clones

For homogeneous Δ in finite language:

Reducts up to **fo-interdefinability** $\leftrightarrow$

closed **permutation groups** $\supseteq$ Aut(Δ);

Reducts up to **pp-interdefinability** $\leftrightarrow$

closed **clones** $\supseteq$ Aut(Δ).

Larger reducts $\rightarrow$ harder CSP

$\Gamma \leq_{pp} \Gamma' \quad \rightarrow \quad \mathrm{CSP}(\Gamma) \leq_{Poltime} \mathrm{CSP}(\Gamma')$

Groups and Clones

For homogeneous Δ in finite language:

Reducts up to **fo-interdefinability** $\leftrightarrow$

closed **permutation groups** $\supseteq$ Aut(Δ);

Reducts up to **pp-interdefinability** $\leftrightarrow$

closed **clones** $\supseteq$ Aut(Δ).

Larger reducts $\rightarrow$ harder CSP

$\Gamma \leq_{pp} \Gamma' \quad \rightarrow \quad \text{CSP}(\Gamma) \leq_{Poltime} \text{CSP}(\Gamma')$

Larger clones $\rightarrow$ easier CSP

$\text{Pol}(\Gamma) \subseteq \text{Pol}(\Gamma') \quad \rightarrow \quad \text{CSP}(\Gamma') \leq_{Poltime} \text{CSP}(\Gamma)$

The reducts of the random graph, revisited

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

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

Let $- : V \to V$ be an isomorphism between G and $\bar{G}$.

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

Let $- : V \to V$ be an isomorphism between G and $\bar{G}$.

For $c \in V$, let G_c be the graph that arises by switching
all edges and non-edges containing c.

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

Let $- : V \to V$ be an isomorphism between G and $\bar{G}$.

For $c \in V$, let G_c be the graph that arises by switching all edges and non-edges containing c.

Let $\mathrm{sw}_c : V \to V$ be an isomorphism between G and G_c.

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

Let $- : V \to V$ be an isomorphism between G and $\bar{G}$.

For $c \in V$, let G_c be the graph that arises by switching all edges and non-edges containing c.

Let $\mathrm{sw}_c : V \to V$ be an isomorphism between G and G_c.

Theorem (Thomas '91)

The closed groups containing $\mathrm{Aut}(G)$ are the following:

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

Let $- : V \to V$ be an isomorphism between G and $\bar{G}$.

For $c \in V$, let G_c be the graph that arises by switching all edges and non-edges containing c.

Let $\text{sw}_c : V \to V$ be an isomorphism between G and G_c.

Theorem (Thomas '91)

The closed groups containing $\text{Aut}(G)$ are the following:

1. $\text{Aut}(G)$

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

Let $- : V \to V$ be an isomorphism between G and $\bar{G}$.

For $c \in V$, let G_c be the graph that arises by switching all edges and non-edges containing c.

Let $\mathrm{sw}_c : V \to V$ be an isomorphism between G and G_c.

Theorem (Thomas '91)

The closed groups containing $\mathrm{Aut}(G)$ are the following:

1. $\mathrm{Aut}(G)$
2. $\langle \{-\} \cup \mathrm{Aut}(G) \rangle$

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

Let $- : V \to V$ be an isomorphism between G and $\bar{G}$.

For $c \in V$, let G_c be the graph that arises by switching all edges and non-edges containing c.

Let $\mathrm{sw}_c : V \to V$ be an isomorphism between G and G_c.

Theorem (Thomas '91)

The closed groups containing $\mathrm{Aut}(G)$ are the following:

1. $\mathrm{Aut}(G)$
2. $\langle \{-\} \cup \mathrm{Aut}(G) \rangle$
3. $\langle \{\mathrm{sw}_c\} \cup \mathrm{Aut}(G) \rangle$

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

Let $- : V \to V$ be an isomorphism between G and $\bar{G}$.

For $c \in V$, let G_c be the graph that arises by switching all edges and non-edges containing c.

Let $\mathrm{sw}_c : V \to V$ be an isomorphism between G and G_c.

Theorem (Thomas '91)

The closed groups containing $\mathrm{Aut}(G)$ are the following:

1. $\mathrm{Aut}(G)$
2. $\langle \{-\} \cup \mathrm{Aut}(G) \rangle$
3. $\langle \{\mathrm{sw}_c\} \cup \mathrm{Aut}(G) \rangle$
4. $\langle \{-, \mathrm{sw}_c\} \cup \mathrm{Aut}(G) \rangle$

The reducts of the random graph, revisited

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

Let $\bar{G}$ be the graph that arises by switching edges and non-edges.

Let $- : V \to V$ be an isomorphism between G and $\bar{G}$.

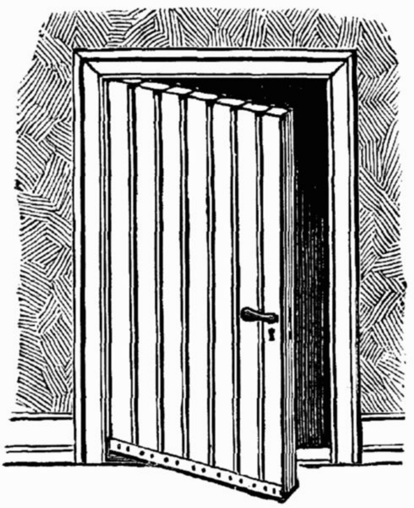
For $c \in V$, let G_c be the graph that arises by switching all edges and non-edges containing c.

Let $\mathrm{sw}_c : V \to V$ be an isomorphism between G and G_c.

Theorem (Thomas '91)

The closed groups containing $\mathrm{Aut}(G)$ are the following:

1. $\mathrm{Aut}(G)$
2. $\langle \{-\} \cup \mathrm{Aut}(G) \rangle$
3. $\langle \{\mathrm{sw}_c\} \cup \mathrm{Aut}(G) \rangle$
4. $\langle \{-, \mathrm{sw}_c\} \cup \mathrm{Aut}(G) \rangle$
5. The full symmetric group S_V.



Part III

Making the infinite finite

(Ramsey theory)

How to classify all reducts up to . . .-interdefinability?

Climb up the lattice!

Canonical functions on the Random graph

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

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

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,

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.

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.

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.

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.

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.

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.

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**:

Finding canonical behaviour

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

For all graphs H
there exists a graph S such that

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 $f : G \to G$, 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 $f : G \to G$, 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.

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).

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.

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:

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)

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)

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 $-$

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

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.

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.

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
- $-$
- sw_c
- One of 9 canonical binary injections.

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

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
- $-$
- sw_c
- One of 9 canonical binary injections.

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

More involved argument: Extend G by a random dense linear order.

Ramsey classes

Ramsey classes

Let S, H, P be structures in the same signature τ.

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

means:

Ramsey classes

Let S, H, P be structures in the same signature τ.

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

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.

Ramsey classes

Let S, H, P be structures in the same signature τ.

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

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 τ-structures is called a *Ramsey class* iff
for all $H, P \in \mathcal{C}$ there exists S in $\mathcal{C}$ such that $S \to (H)^P$.

Canonical functions on Ramsey structures

Let Δ now be an arbitrary structure.

Canonical functions on Ramsey structures

Let Δ 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 Δ 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 Δ is

- Ramsey
- homogeneous in a finite language
- (ordered),

then all finite substructures of Δ have a copy in Δ
on which f is canonical.

Canonical functions on Ramsey structures

Let Δ 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 Δ is

- Ramsey
- homogeneous in a finite language
- (ordered),

then all finite substructures of Δ have a copy in Δ on which f is canonical.

Thus: Any $f : \Delta \to \Delta$ generates a canonical function,

Canonical functions on Ramsey structures

Let Δ 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 Δ is

- Ramsey
- homogeneous in a finite language
- (ordered),

then all finite substructures of Δ have a copy in Δ
on which f is canonical.

Thus: Any $f : \Delta \to \Delta$ generates a canonical function, but it could be the identity.

What we would like to do...

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 Δ.

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 Δ.

Why don't you just do it?

Adding constants to Ramsey structures

Problem

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

Adding constants to Ramsey structures

Problem

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

Canonizing functions on Ramsey structures

Proposition

If Δ 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 Δ
- behaves like f on $\{c_1, \ldots, c_n\}$.

Canonizing functions on Ramsey structures

Proposition (new proof 2011!)

If Δ 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 Δ
- behaves like f on $\{c_1, \ldots, c_n\}$.

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

Canonizing functions on Ramsey structures

Proposition (new proof 2011!)

If Δ 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 Δ
- behaves like f on $\{c_1, \ldots, c_n\}$.

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

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

Canonizing functions on Ramsey structures

Proposition (new proof 2011!)

If Δ 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 Δ
- behaves like f on $\{c_1, \ldots, c_n\}$.

Set $S := \{g : \Delta^k \to \Delta \mid \quad g$ 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 2011!)

If Δ 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 Δ
- behaves like f on $\{c_1, \ldots, c_n\}$.

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

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 2011!)

If Δ 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 Δ
- behaves like f on $\{c_1, \ldots, c_n\}$.

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

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 $[h(x_1, \ldots, x_k)]_\sim$.

Canonizing functions on Ramsey structures

Proposition (new proof 2011!)

If Δ 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 Δ
- behaves like f on $\{c_1, \ldots, c_n\}$.

Set $S := \{g : \Delta^k \to \Delta \mid \quad g$ 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. □

Minimal clones above Ramsey structures

Theorem (Bodirsky, MP, Tsankov '10)

Let Γ be a reduct of a finite language homogeneous ordered Ramsey structure Δ. Then:

Minimal clones above Ramsey structures

Theorem (Bodirsky, MP, Tsankov '10)

Let Γ be a reduct of a finite language homogeneous ordered Ramsey structure Δ. 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 Γ be a reduct of a finite language homogeneous ordered Ramsey structure Δ. Then:

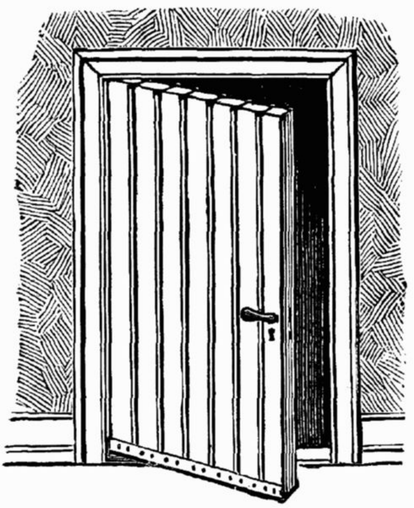
- Every minimal closed superclone of $\text{Pol}(\Gamma)$ is generated by such a canonical function.
- If Γ has a finite language, then there are finitely many minimal closed superclones of $\text{Pol}(\Gamma)$.
  (Arity bound!)

Minimal clones above Ramsey structures

Theorem (Bodirsky, MP, Tsankov '10)

Let Γ be a reduct of a finite language homogeneous ordered Ramsey structure Δ. Then:

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



Part IV

The Graph-SAT dichotomy

The Graph Satisfiability Problem

The Graph Satisfiability Problem

Let Ψ be a finite set of graph formulas.

Computational problem: Graph-SAT(Ψ)

INPUT:

- A set W of variables (vertices), and
- statements $\phi_1, \ldots, \phi_n$ about the elements of W, where each ϕ_i is taken from Ψ.

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

The Graph Satisfiability Problem

Let Ψ be a finite set of graph formulas.

Computational problem: Graph-SAT(Ψ)

INPUT:

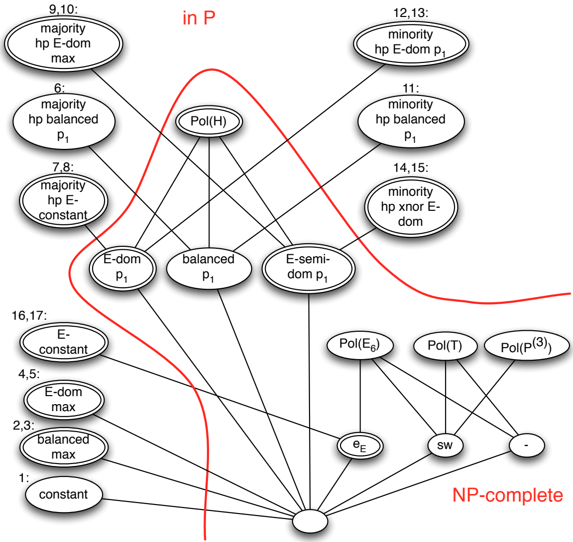
- A set W of variables (vertices), and
- statements $\phi_1, \ldots, \phi_n$ about the elements of W, where each ϕ_i is taken from Ψ.

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

Theorem

Graph-SAT(Ψ) is either in P or NP-complete, for all Ψ.

The Graph-SAT dichotomy visualized



Theorem

The following 17 distinct clones are precisely the minimal tractable closed clones containing 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.

The Meta Problem

The Meta Problem

Meta-Problem of Graph-SAT(Ψ)

INPUT: A finite set Ψ of graph formulas.

QUESTION: Is Graph-SAT(Ψ) in P?

The Meta Problem

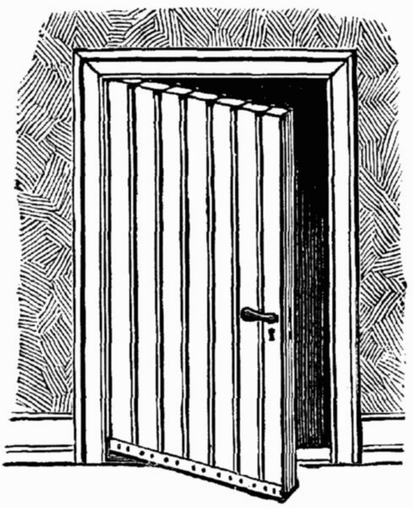
Meta-Problem of Graph-SAT(Ψ)

INPUT: A finite set Ψ of graph formulas.

QUESTION: Is Graph-SAT(Ψ) in P?

Theorem (Bodirsky, MP '10)

The Meta-Problem of Graph-SAT(Ψ) is decidable.



Part V

The future

Other homogeneous structures

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

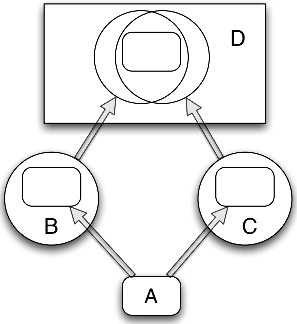
Temp-SAT(Ψ): Is there a linear order such that...

Other homogeneous structures

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

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

The classes of finite graphs and linear orders are *amalgamation classes*.



Amalgamation classes

Further amalgamation classes.

Amalgamation classes

Further amalgamation classes.

- Partial orders

Amalgamation classes

Further amalgamation classes.

- Partial orders
- Lattices

Amalgamation classes

Further amalgamation classes.

- Partial orders
- Lattices
- Distributive lattices

Further amalgamation classes.

- Partial orders
- Lattices
- Distributive lattices
- Metric spaces with rational distances

Amalgamation classes

Further amalgamation classes.

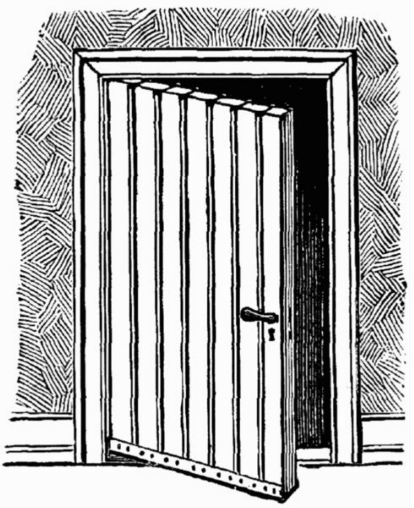
- Partial orders
- Lattices
- Distributive lattices
- Metric spaces with rational distances
- Tournaments

Amalgamation classes

Further amalgamation classes.

- Partial orders
- Lattices
- Distributive lattices
- Metric spaces with rational distances
- Tournaments

Homogeneous digraphs classified by Cherlin.



References

Schaefer's theorem for graphs

by Manuel Bodirsky and Michael Pinsker

Proceedings of STOC, 2011.

Reducts of Ramsey structures

by Manuel Bodirsky and Michael Pinsker

AMS Contemporary Mathematics, 2011.