LIX, Ecole Polytechnique
Technische Universität Dresden

The Countable Boolean Vector Space and Bit Vector CSPs

PhD Defence

September 17, 2015

François Bossière

Jury Members

Advisor: Manuel Bodirsky
Reviewers: Florent Madelaine
Csaba Szabó

Examiners: Olivier Bournez
Arnaud Durand

Constraint Satisfaction Problems

Informal definition of CSPs

- A CSP is a computational problem.
- The input consists of a finite set of variables and a finite set of constraints imposed on those variables.
- The task is to decide whether there is an assignment of values to the variables such that all the constraints are simultaneously satisfied.

Examples

- Is a propositional formula in CNF with at most three literals per clause satisfiable on $\{0, 1\}$?
- Is there a solution to a finite set of linear equations over $\mathbb{F}_2$?

Constraint Satisfaction Problems

Informal definition of CSPs

- A CSP is a computational problem.
- The input consists of a finite set of variables and a finite set of constraints imposed on those variables.
- The task is to decide whether there is an assignment of values to the variables such that all the constraints are simultaneously satisfied.

Examples

- Is a propositional formula in CNF with at most three literals per clause satisfiable on $\{0, 1\}$?
- Is there a solution to a finite set of linear equations over $\mathbb{F}_2$?

Formalisation of CSPs

Preliminaries

- Given a relational signature τ, an atomic formula is of the form $R(\overline{x})$ with R a relation in τ.
- A primitive positive (pp) formula on τ is of the form $\exists x_1 \ldots x_n(\phi_1(\overline{x}) \wedge \cdots \wedge \phi_k(\overline{x}))$ where all ϕ_i are atomic formulas.

Formal definition of *CSPs*

Given a structure Γ on a finite relational signature τ, we define the computational problem CSP(Γ):

- **Input**: a primitive positive sentence ϕ.
- **Question**: $\Gamma \models \phi$?

Natural question: what is the complexity of CSP(Γ) for a given Γ?
Proposition: it does not change when adding pp-definable relations to Γ.

Formalisation of CSPs

Preliminaries

- Given a relational signature τ, an atomic formula is of the form $R(\overline{x})$ with R a relation in τ.
- A primitive positive (pp) formula on τ is of the form $\exists x_1 \ldots x_n(\phi_1(\overline{x}) \wedge \cdots \wedge \phi_k(\overline{x}))$ where all ϕ_i are atomic formulas.

Formal definition of *CSPs*

Given a structure Γ on a finite relational signature τ, we define the computational problem CSP(Γ):

◇ **Input**: a primitive positive sentence ϕ.

◇ **Question**: $\Gamma \models \phi$?

Natural question: what is the complexity of CSP(Γ) for a given Γ?
Proposition: it does not change when adding pp-definable relations to Γ.

Formalisation of CSPs

Preliminaries

- Given a relational signature τ, an atomic formula is of the form $R(\overline{x})$ with R a relation in τ.
- A primitive positive (pp) formula on τ is of the form $\exists x_1 \dots x_n(\phi_1(\overline{x}) \wedge \cdots \wedge \phi_k(\overline{x}))$ where all ϕ_i are atomic formulas.

Formal definition of *CSPs*

Given a structure Γ on a finite relational signature τ, we define the computational problem CSP(Γ):

- ◇ **Input**: a primitive positive sentence ϕ.
- ◇ **Question**: $\Gamma \models \phi$?

Natural question: what is the complexity of CSP(Γ) for a given Γ?

Proposition: it does not change when adding pp-definable relations to Γ.

Formalisation of CSPs

Preliminaries

- Given a relational signature τ, an atomic formula is of the form $R(\overline{x})$ with R a relation in τ.
- A primitive positive (pp) formula on τ is of the form $\exists x_1 \dots x_n (\phi_1(\overline{x}) \wedge \cdots \wedge \phi_k(\overline{x}))$ where all ϕ_i are atomic formulas.

Formal definition of *CSPs*

Given a structure Γ on a finite relational signature τ, we define the computational problem CSP(Γ):

- **Input**: a primitive positive sentence ϕ.
- **Question**: $\Gamma \models \phi$?

Natural question: what is the complexity of CSP(Γ) for a given Γ?
Proposition: it does not change when adding pp-definable relations to Γ.

Dichotomy for finite Structures

- Schaefer'77: for any 2-element structure Γ, CSP(Γ) is either polynomially solvable or NP-complete.

Conjecture (Feder-Vardi'93)

This dichotomy holds for every finite structure Γ.

- Bulatov'03: confirmed Feder-Vardi's conjecture for domains of size 3.
- Markovic'12: confirmed for domains of size 4 (announced but not published yet).
- The conjecture is already open for domains of size ≥ 5.

What about infinite structures?

Infinite Structures

Non-Dichotomy

- Ladner'75: if $P \neq NP$, there are NP-intermediate computational decision problems, i.e., problems in NP that are neither polynomial-time tractable nor NP-complete.
- Bodirsky-Grohe'08: Every computational decision problem is polynomial-time equivalent to a CSP with an infinite template.
- Consequently: no dichotomy for CSPs on infinite structures.

Question

Can we identify large natural classes of CSPs on infinite structures whose complexity can be classified?

Infinite Structures

Non-Dichotomy

- Ladner'75: if P $\neq$ NP, there are NP-intermediate computational decision problems, i.e., problems in NP that are neither polynomial-time tractable nor NP-complete.
- Bodirsky-Grohe'08: Every computational decision problem is polynomial-time equivalent to a CSP with an infinite template.
- Consequently: no dichotomy for CSPs on infinite structures.

Question

Can we identify large natural classes of CSPs on infinite structures whose complexity can be classified?

The Countable Boolean Vector Space

Definition

There is up to isomorphism a unique countably infinite vector space over the field $\mathbb{F}_2$. We denote it by $(V; +)$.

Characteristics:

- fundamental structure in Model Theory
- Fraïssé limit of the class of finite $\mathbb{F}_2$-vector spaces
- **homogeneous**, i.e., any partial isomorphism between finite substructures of $(V; +)$ can be extended to an automorphism of $(V; +)$

Reducts of $(V;+)$

Definition

A reduct of a structure Δ is a relational structure with the same domain as Δ whose **relations are definable with first-order formulas over Δ**.

Examples of relations definable over $(V;+)$: let $n \geq 3$ be an integer,

- $x = 0 :\Leftrightarrow x + x = x$
- $\mathrm{Eq}_n(x_1, \ldots, x_n) :\Leftrightarrow \Sigma_{i \leq n} x_i = 0$
- $\mathrm{Ind}_n(x_1, \ldots, x_n) :\Leftrightarrow x_1, \ldots, x_n$ are linearly independent
- $\mathrm{Ieq}_n(x_1, \ldots, x_n) :\Leftrightarrow \mathrm{Eq}_n(x_1, \ldots, x_n)$ and every subfamily of size $n-1$ of $x_1, \ldots, x_n$ is linearly independent

Reducts of $(V;+)$

Definition

A reduct of a structure Δ is a relational structure with the same domain as Δ whose **relations are definable with first-order formulas over** Δ.

Examples of relations definable over $(V;+)$: let $n \geq 3$ be an integer,

- $x = 0 :\Leftrightarrow x + x = x$
- $\mathrm{Eq}_n(x_1, \ldots, x_n) :\Leftrightarrow \Sigma_{i \leq n} x_i = 0$
- $\mathrm{Ind}_n(x_1, \ldots, x_n) :\Leftrightarrow x_1, \ldots, x_n$ are linearly independent
- $\mathrm{Ieq}_n(x_1, \ldots, x_n) :\Leftrightarrow \mathrm{Eq}_n(x_1, \ldots, x_n)$ and every subfamily of size $n - 1$ of $x_1, \ldots, x_n$ is linearly independent

Reducts of $(V;+)$

Definition

A reduct of a structure Δ is a relational structure with the same domain as Δ whose **relations are definable with first-order formulas over** Δ.

Examples of relations definable over $(V;+)$: let $n \geq 3$ be an integer,

- $x = 0 :\Leftrightarrow x + x = x$
- $\mathrm{Eq}_n(x_1, \ldots, x_n) :\Leftrightarrow \Sigma_{i \leq n} x_i = 0$
- $\mathrm{Ind}_n(x_1, \ldots, x_n) :\Leftrightarrow x_1, \ldots, x_n$ are linearly independent
- $\mathrm{Ieq}_n(x_1, \ldots, x_n) :\Leftrightarrow \mathrm{Eq}_n(x_1, \ldots, x_n)$ and every subfamily of size $n-1$ of $x_1, \ldots, x_n$ is linearly independent

Reducts of $(V;+)$

Definition

A reduct of a structure Δ is a relational structure with the same domain as Δ whose **relations are definable with first-order formulas over** Δ.

Examples of relations definable over $(V;+)$: let $n \geq 3$ be an integer,

- $x = 0 :\Leftrightarrow x + x = x$
- $\mathrm{Eq}_n(x_1, \dots, x_n) :\Leftrightarrow \Sigma_{i \leq n} x_i = 0$
- $\mathrm{Ind}_n(x_1, \dots, x_n) :\Leftrightarrow x_1, \dots, x_n$ are linearly independent
- $\mathrm{Ieq}_n(x_1, \dots, x_n) :\Leftrightarrow \mathrm{Eq}_n(x_1, \dots, x_n)$ and every subfamily of size $n-1$ of $x_1, \dots, x_n$ is linearly independent

Reducts of $(V; +)$

Definition

A reduct of a structure Δ is a relational structure with the same domain as Δ whose **relations are definable with first-order formulas over** Δ.

Examples of relations definable over $(V; +)$: let $n \geq 3$ be an integer,

- $x = 0 :\Leftrightarrow x + x = x$
- $\mathrm{Eq}_n(x_1, \ldots, x_n) :\Leftrightarrow \Sigma_{i \leq n} x_i = 0$
- $\mathrm{Ind}_n(x_1, \ldots, x_n) :\Leftrightarrow x_1, \ldots, x_n$ are linearly independent
- $\mathrm{Ieq}_n(x_1, \ldots, x_n) :\Leftrightarrow \mathrm{Eq}_n(x_1, \ldots, x_n)$ and every subfamily of size $n-1$ of $x_1, \ldots, x_n$ is linearly independent

Bit Vector CSPs

Definition

A Bit Vector CSP is a problem $\mathrm{CSP}(\Gamma)$ where Γ is a reduct of $(V; +)$.

Examples:

1. $\mathrm{CSP}(V; \mathrm{Eq}_3, \neq)$
2. $\mathrm{CSP}(V; \mathrm{leq}_4, \mathrm{leq}_8, Z_1 \cup Z_2 \cup \mathrm{Ind}_4)$ where:
   - $Z_1(x, y, z, t) :\Leftrightarrow x = 0 \wedge \mathrm{leq}_3(y, z, t)$, and
   - $Z_2(x, y, z, t) :\Leftrightarrow x \notin \{0, y, z, t\} \wedge \mathrm{leq}_3(y, z, t)\}$
3. $\mathrm{CSP}(V; \mathrm{leq}_5, Q)$ where:
   - $Q(x, y, z, t_1, t_2, t_3) :\Leftrightarrow \mathrm{leq}_4(x, y, z, t_1) \vee \mathrm{leq}_5(x, y, z, t_2, t_3)$

Remark: 1. is in P by Gaussian elimination, but classifying the complexity of Examples 2. and 3. is not that easy.

Bit Vector CSPs

Definition

A Bit Vector CSP is a problem CSP(Γ) where Γ is a reduct of $(V;+)$.

Examples:

1. CSP$(V;\mathrm{Eq}_3,\neq)$
2. CSP$(V;\mathrm{Ieq}_4,\mathrm{Ieq}_8,Z_1\cup Z_2\cup\mathrm{Ind}_4)$ where:
   - $Z_1(x,y,z,t) :\Leftrightarrow x=0\wedge \mathrm{Ieq}_3(y,z,t)$, and
   - $Z_2(x,y,z,t) :\Leftrightarrow x\notin\{0,y,z,t\}\wedge \mathrm{Ieq}_3(y,z,t)\}$
3. CSP$(V;\mathrm{Ieq}_5,Q)$ where:
   - $Q(x,y,z,t_1,t_2,t_3) :\Leftrightarrow \mathrm{Ieq}_4(x,y,z,t_1)\vee \mathrm{Ieq}_5(x,y,z,t_2,t_3)$

Remark: 1. is in P by Gaussian elimination, but classifying the complexity of Examples 2. and 3. is not that easy.

Bit Vector CSPs

Definition

A Bit Vector CSP is a problem $\mathrm{CSP}(\Gamma)$ where Γ is a reduct of $(V;+)$.

Examples:

1. $\mathrm{CSP}(V;\mathrm{Eq}_3,\neq)$
2. $\mathrm{CSP}(V;\mathrm{leq}_4,\mathrm{leq}_8,Z_1\cup Z_2\cup\mathrm{Ind}_4)$ where:
   - $Z_1(x,y,z,t) :\Leftrightarrow x=0\wedge\mathrm{leq}_3(y,z,t)$, and
   - $Z_2(x,y,z,t) :\Leftrightarrow x\notin\{0,y,z,t\}\wedge\mathrm{leq}_3(y,z,t))\}$
3. $\mathrm{CSP}(V;\mathrm{leq}_5,Q)$ where:
   - $Q(x,y,z,t_1,t_2,t_3) :\Leftrightarrow \mathrm{leq}_4(x,y,z,t_1)\vee\mathrm{leq}_5(x,y,z,t_2,t_3)$

Remark: 1. is in P by Gaussian elimination, but classifying the complexity of Examples 2. and 3. is not that easy.

Bit Vector CSPs

Definition

A Bit Vector CSP is a problem CSP(Γ) where Γ is a reduct of $(V; +)$.

Examples:

1. $\mathrm{CSP}(V; \mathrm{Eq}_3, \neq)$
2. $\mathrm{CSP}(V; \mathrm{Ieq}_4, \mathrm{Ieq}_8, Z_1 \cup Z_2 \cup \mathrm{Ind}_4)$ where:
   - $Z_1(x, y, z, t) :\Leftrightarrow x = 0 \wedge \mathrm{Ieq}_3(y, z, t)$, and
   - $Z_2(x, y, z, t) :\Leftrightarrow x \notin \{0, y, z, t\} \wedge \mathrm{Ieq}_3(y, z, t))\}$
3. $\mathrm{CSP}(V; \mathrm{Ieq}_5, Q)$ where:
   - $Q(x, y, z, t_1, t_2, t_3) :\Leftrightarrow \mathrm{Ieq}_4(x, y, z, t_1) \vee \mathrm{Ieq}_5(x, y, z, t_2, t_3)$

Remark: 1. is in P by Gaussian elimination, but classifying the complexity of Examples 2. and 3. is not that easy.

Bit Vector CSPs

Definition

A Bit Vector CSP is a problem CSP(Γ) where Γ is a reduct of $(V; +)$.

Examples:

1. $\mathrm{CSP}(V; \mathrm{Eq}_3, \neq)$
2. $\mathrm{CSP}(V; \mathrm{leq}_4, \mathrm{leq}_8, Z_1 \cup Z_2 \cup \mathrm{Ind}_4)$ where:
   - $Z_1(x, y, z, t) :\Leftrightarrow x = 0 \wedge \mathrm{leq}_3(y, z, t)$, and
   - $Z_2(x, y, z, t) :\Leftrightarrow x \notin \{0, y, z, t\} \wedge \mathrm{leq}_3(y, z, t)\}$
3. $\mathrm{CSP}(V; \mathrm{leq}_5, Q)$ where:
   - $Q(x, y, z, t_1, t_2, t_3) :\Leftrightarrow \mathrm{leq}_4(x, y, z, t_1) \vee \mathrm{leq}_5(x, y, z, t_2, t_3)$

Remark: 1. is in P by Gaussian elimination, but classifying the complexity of Examples 2. and 3. is not that easy.

ω-categoricity

Conjecture

A Bit Vector CSP is either in P, or NP-complete.

How can we classify the complexity of CSP(Γ)?

- For finite structures, we can use a **universal algebraic approach**.
- To adapt it for an infinite Γ, we need a strong property on Γ: *ω-categoricity*.

Definition (Ryll-Nardzewski's form)

A countable structure of domain D is ω-categorical iff it has finitely many orbits w.r.t. the natural action of its automorphism group on D^n, for all n.

Facts:

- A reduct of an ω-categorical structure is ω-categorical.
- $(V; +)$ is ω-categorical.

ω-categoricity

Conjecture

A Bit Vector CSP is either in P, or NP-complete.

How can we classify the complexity of CSP(Γ)?

- For finite structures, we can use a **universal algebraic approach**.
- To adapt it for an infinite Γ, we need a strong property on Γ: *ω-categoricity*.

Definition (Ryll-Nardzewski's form)

A countable structure of domain D is ω-categorical iff it has finitely many orbits w.r.t. the natural action of its automorphism group on D^n, for all n.

Facts:

- A reduct of an ω-categorical structure is ω-categorical.
- $(V; +)$ is ω-categorical.

ω-categoricity

Conjecture

A Bit Vector CSP is either in P, or NP-complete.

How can we classify the complexity of CSP(Γ)?

- For finite structures, we can use a **universal algebraic approach**.
- To adapt it for an infinite Γ, we need a strong property on Γ: *ω-categoricity*.

Definition (Ryll-Nardzewski's form)

A countable structure of domain D is ω-categorical iff it has finitely many orbits w.r.t. the natural action of its automorphism group on D^n, for all n.

Facts:

- A reduct of an ω-categorical structure is ω-categorical.
- $(V;+)$ is ω-categorical.

ω-categoricity

Conjecture

A Bit Vector CSP is either in P, or NP-complete.

How can we classify the complexity of CSP(Γ)?

- For finite structures, we can use a **universal algebraic approach**.
- To adapt it for an infinite Γ, we need a strong property on Γ: ω-*categoricity*.

Definition (Ryll-Nardzewski's form)

A countable structure of domain D is ω-categorical iff it has finitely many orbits w.r.t. the natural action of its automorphism group on D^n, for all n.

Facts:

- A reduct of an ω-categorical structure is ω-categorical.
- $(V; +)$ is ω-categorical.

Polymorphisms

- A m-ary operation f preserves a n-ary relation R if for all n-tuples $\overline{x_1}, \ldots, \overline{x_m}$ in R, the n-tuple $(f(x_{1,i}, \ldots, x_{m,i}))_{1 \leq i \leq n}$ is again in R.
- f is called a polymorphism of a relational structure Γ if it preserves every relation of Γ.
- A unary polymorphism of Γ is called an endomorphism of Γ.

$$
\begin{array}{ccccccccc}
 & & \overset{f}{\frown} & \overset{f}{\frown} & \overset{f}{\frown} & & \overset{f}{\frown} & \\
\overline{x_1} & = & x_{1,1} & x_{1,2} & x_{1,3} & \cdots & x_{1,n} & \in R \\
\overline{x_2} & = & x_{2,1} & x_{2,2} & x_{2,3} & \cdots & x_{2,n} & \in R \\
\vdots & & \vdots & \vdots & \vdots & & \vdots & \\
\overline{x_m} & = & x_{m,1} & x_{m,2} & x_{m,3} & \cdots & x_{m,n} & \in R \\
 & & \smile & \smile & \smile & & \smile & \\
 & & \parallel & \parallel & \parallel & & \parallel & \\
 & & z_1 & z_2 & z_3 & \cdots & z_n & \in R
\end{array}
$$

Universal Algebraic Approach and ω-categoricity

- Pol(Γ) (resp. End(Γ)): the set of all polymorphisms (resp. endomorphisms) of Γ.
- Inv(F): the set of all relations preserved by a set F of operations.
- $\langle\Gamma\rangle_{\text{pp}}$: the set of all relations which are **definable with a primitive positive formula** over Γ.

Theorem (Geiger'68 & Bodirsky,Nesetril'03)

For every countably infinite ω-categorical or finite structure Γ:

$$\text{Inv}(\text{Pol}(\Gamma)) = \langle\Gamma\rangle_{\text{pp}}$$

Consequently, **the complexity of** CSP(Γ) **is determined by** Pol(Γ).
We first focus on **understanding** End(Γ).

Universal Algebraic Approach and ω-categoricity

- $\mathrm{Pol}(\Gamma)$ (resp. $\mathrm{End}(\Gamma)$): the set of all polymorphisms (resp. endomorphisms) of Γ.
- $\mathrm{Inv}(F)$: the set of all relations preserved by a set F of operations.
- $\langle\Gamma\rangle_{\mathrm{pp}}$: the set of all relations which are **definable with a primitive positive formula** over Γ.

Theorem (Geiger'68 & Bodirsky,Nesetril'03)

For every countably infinite ω-categorical or finite structure Γ:

$$\mathrm{Inv}(\mathrm{Pol}(\Gamma)) = \langle\Gamma\rangle_{\mathrm{pp}}$$

Consequently, **the complexity of** $\mathrm{CSP}(\Gamma)$ **is determined by** $\mathrm{Pol}(\Gamma)$.
We first focus on **understanding** $\mathrm{End}(\Gamma)$.

Classification of Endomorphism monoids of Reducts

Theorem

Let Γ be a reduct of $(V;+)$ which is not homomorphically equivalent to a reduct of $(V;0)$. Then $\mathrm{End}(\Gamma)$ belongs to a list of **27 monoids**.

Remarks:

- Homomorphic equivalence preserves the complexity of the CSP.
- CSPs of reducts of $(V;0)$ are fully classified in the thesis.

What method do we use?

Classification of Endomorphism monoids of Reducts

Theorem

Let Γ be a reduct of $(V; +)$ which is not homomorphically equivalent to a reduct of $(V; 0)$. Then $\mathrm{End}(\Gamma)$ belongs to a list of **27 monoids**.

Remarks:

- Homomorphic equivalence preserves the complexity of the CSP.
- CSPs of reducts of $(V; 0)$ are fully classified in the thesis.

What method do we use?

Bodirsky and Pinsker's Method using Ramsey Theory

Goal: classify End(Γ) for reducts Γ of structures with a strong combinatorial property called *Ramsey Property*.

Notation: Let $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}$ be structures with same signature, and $r \in \mathbb{N}$.

- $\binom{\mathfrak{B}}{\mathfrak{A}}$ denotes the set of substructures of $\mathfrak{B}$ isomorphic to $\mathfrak{A}$.
- we write $\mathfrak{C} \to (\mathfrak{B})^{\mathfrak{A}}_r$ if for all colouring $\chi\colon \binom{\mathfrak{C}}{\mathfrak{A}} \to \{1, \ldots, r\}$ there exists $\mathfrak{B}' \in \binom{\mathfrak{C}}{\mathfrak{B}}$ such that χ is monochromatic on $\binom{\mathfrak{B}'}{\mathfrak{A}}$.

Ramsey Property

A structure Γ has the Ramsey property if for all finite substructures $\mathfrak{A}, \mathfrak{B}$ of Γ and all $k \in \mathbb{N}$, we have: $\Gamma \to (\mathfrak{B})^{\mathfrak{A}}_r$.

Graham-Leeb-Rothschild'71: $(V; +)$ is Ramsey.

Bodirsky and Pinsker's Method using Ramsey Theory

Goal: classify End(Γ) for reducts Γ of structures with a strong combinatorial property called *Ramsey Property*.

Notation: Let $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}$ be structures with same signature, and $r \in \mathbb{N}$.

- $\binom{\mathfrak{B}}{\mathfrak{A}}$ denotes the set of substructures of $\mathfrak{B}$ isomorphic to $\mathfrak{A}$.
- we write $\mathfrak{C} \to (\mathfrak{B})^{\mathfrak{A}}_r$ if for all colouring $\chi\colon \binom{\mathfrak{C}}{\mathfrak{A}} \to \{1, \dots, r\}$ there exists $\mathfrak{B}' \in \binom{\mathfrak{C}}{\mathfrak{B}}$ such that χ is monochromatic on $\binom{\mathfrak{B}'}{\mathfrak{A}}$.

Ramsey Property

A structure Γ has the Ramsey property if for all finite substructures $\mathfrak{A}, \mathfrak{B}$ of Γ and all $k \in \mathbb{N}$, we have: $\Gamma \to (\mathfrak{B})^{\mathfrak{A}}_r$.

Graham-Leeb-Rothschild'71: $(V; +)$ is Ramsey.

Bodirsky and Pinsker's Method using Ramsey Theory

Goal: classify End(Γ) for reducts Γ of structures with a strong combinatorial property called *Ramsey Property*.

Notation: Let $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}$ be structures with same signature, and $r \in \mathbb{N}$.

- $\binom{\mathfrak{B}}{\mathfrak{A}}$ denotes the set of substructures of $\mathfrak{B}$ isomorphic to $\mathfrak{A}$.
- we write $\mathfrak{C} \to (\mathfrak{B})^{\mathfrak{A}}_r$ if for all colouring $\chi \colon \binom{\mathfrak{C}}{\mathfrak{A}} \to \{1, \ldots, r\}$ there exists $\mathfrak{B}' \in \binom{\mathfrak{C}}{\mathfrak{B}}$ such that χ is monochromatic on $\binom{\mathfrak{B}'}{\mathfrak{A}}$.

Ramsey Property

A structure Γ has the Ramsey property if for all finite substructures $\mathfrak{A}, \mathfrak{B}$ of Γ and all $k \in \mathbb{N}$, we have: $\Gamma \to (\mathfrak{B})^{\mathfrak{A}}_r$.

Graham-Leeb-Rothschild'71: $(V; +)$ is Ramsey.

Bodirsky and Pinsker's Method using Ramsey Theory

Goal: classify End(Γ) for reducts Γ of structures with a strong combinatorial property called *Ramsey Property*.

Notation: Let $\mathfrak{A}, \mathfrak{B}, \mathfrak{C}$ be structures with same signature, and $r \in \mathbb{N}$.

- $\binom{\mathfrak{B}}{\mathfrak{A}}$ denotes the set of substructures of $\mathfrak{B}$ isomorphic to $\mathfrak{A}$.
- we write $\mathfrak{C} \to (\mathfrak{B})^{\mathfrak{A}}_r$ if for all colouring $\chi \colon \binom{\mathfrak{C}}{\mathfrak{A}} \to \{1, \ldots, r\}$ there exists $\mathfrak{B}' \in \binom{\mathfrak{C}}{\mathfrak{B}}$ such that χ is monochromatic on $\binom{\mathfrak{B}'}{\mathfrak{A}}$.

Ramsey Property

A structure Γ has the Ramsey property if for all finite substructures $\mathfrak{A}, \mathfrak{B}$ of Γ and all $k \in \mathbb{N}$, we have: $\Gamma \to (\mathfrak{B})^{\mathfrak{A}}_r$.

Graham-Leeb-Rothschild'71: $(V; +)$ is Ramsey.

Generating Canonical Functions

The Ramsey property allows us to prove the existence of *canonical functions*. They play a crucial role in the classification of $\mathrm{End}(\Gamma)$.

- $f : \Delta_1 \to \Delta_2$ is canonical if the orbit of the image of a tuple $\bar{a}$ only depends on the orbit of $\bar{a}$.
- $f : \Delta \to \Delta$ generates g if g belongs to the closure of $\{f\} \cup \mathrm{Aut}(\Delta)$ under composition and pointwise convergence.
- if an endomorphism of a reduct Γ of Δ generates g, then $g \in \mathrm{End}(\Gamma)$.

Theorem (Bodirsky,Pinsker,Tsankov'11)

Let Δ be a homogeneous Ramsey structure **with finite relational signature**. For all $f : \Delta \to \Delta$ and all $C := \{c_1, \ldots, c_n\}$, f generates a canonical function g from $(\Delta, c_1, \ldots, c_n)$ to Δ such that $f{\restriction}C = g{\restriction}C$.

Fact: finitely many canonical functions from Δ_1 to Δ_2 for **finite relational signatures**.

Generating Canonical Functions

The Ramsey property allows us to prove the existence of *canonical functions*. They play a crucial role in the classification of $\mathrm{End}(\Gamma)$.

- $f : \Delta_1 \to \Delta_2$ is canonical if the orbit of the image of a tuple $\bar{a}$ only depends on the orbit of $\bar{a}$.
- $f : \Delta \to \Delta$ generates g if g belongs to the closure of $\{f\} \cup \mathrm{Aut}(\Delta)$ under composition and pointwise convergence.
- if an endomorphism of a reduct Γ of Δ generates g, then $g \in \mathrm{End}(\Gamma)$.

Theorem (Bodirsky,Pinsker,Tsankov'11)

Let Δ be a homogeneous Ramsey structure **with finite relational signature**. For all $f : \Delta \to \Delta$ and all $C := \{c_1, \ldots, c_n\}$, f generates a canonical function g from $(\Delta, c_1, \ldots, c_n)$ to Δ such that $f{\restriction}C = g{\restriction}C$.

Fact: finitely many canonical functions from Δ_1 to Δ_2 for **finite relational signatures**.

Generating Canonical Functions

The Ramsey property allows us to prove the existence of *canonical functions*. They play a crucial role in the classification of $\mathrm{End}(\Gamma)$.

- $f : \Delta_1 \to \Delta_2$ is canonical if the orbit of the image of a tuple $\bar{a}$ only depends on the orbit of $\bar{a}$.
- $f : \Delta \to \Delta$ generates g if g belongs to the closure of $\{f\} \cup \mathrm{Aut}(\Delta)$ under composition and pointwise convergence.
- if an endomorphism of a reduct Γ of Δ generates g, then $g \in \mathrm{End}(\Gamma)$.

Theorem (Bodirsky,Pinsker,Tsankov'11)

Let Δ be a homogeneous Ramsey structure **with finite relational signature**. For all $f : \Delta \to \Delta$ and all $C := \{c_1, \ldots, c_n\}$, f generates a canonical function g from $(\Delta, c_1, \ldots, c_n)$ to Δ such that $f{\restriction}C = g{\restriction}C$.

Fact: finitely many canonical functions from Δ_1 to Δ_2 for **finite relational signatures**.

Generating Canonical Functions

The Ramsey property allows us to prove the existence of *canonical functions*. They play a crucial role in the classification of $\mathrm{End}(\Gamma)$.

- $f : \Delta_1 \to \Delta_2$ is canonical if the orbit of the image of a tuple $\bar{a}$ only depends on the orbit of $\bar{a}$.
- $f : \Delta \to \Delta$ generates g if g belongs to the closure of $\{f\} \cup \mathrm{Aut}(\Delta)$ under composition and pointwise convergence.
- if an endomorphism of a reduct Γ of Δ generates g, then $g \in \mathrm{End}(\Gamma)$.

Theorem (Bodirsky,Pinsker,Tsankov'11)

Let Δ be a homogeneous Ramsey structure **with finite relational signature**. For all $f : \Delta \to \Delta$ and all $C := \{c_1, \ldots, c_n\}$, f generates a canonical function g from $(\Delta, c_1, \ldots, c_n)$ to Δ such that $f{\restriction}C = g{\restriction}C$.

Fact: finitely many canonical functions from Δ_1 to Δ_2 for **finite relational signatures**.

Dealing with a Functional Signature

Proposition

$(V;+)$ is not first-order interdefinable with any homogeneous structure with finite relational signature.

Difficulties:

- we have to adapt Bodirsky-Pinsker's theorem to use it on $(V;+)$
- potentially infinitely many canonical functions from $(V;+)$ to $(V;+)$

Fact

Bodirsky-Pinsker's theorem can be adapted for ω-categorical homogeneous structures **with functional signatures** which have the Ramsey property.

We first study canonical functions from $(V;+)$ to $(V;+)$.

Dealing with a Functional Signature

Proposition

$(V;+)$ is not first-order interdefinable with any homogeneous structure with finite relational signature.

Difficulties:

- we have to adapt Bodirsky-Pinsker's theorem to use it on $(V;+)$
- potentially infinitely many canonical functions from $(V;+)$ to $(V;+)$

Fact

Bodirsky-Pinsker's theorem can be adapted for ω-categorical homogeneous structures **with functional signatures** which have the Ramsey property.

We first study canonical functions from $(V;+)$ to $(V;+)$.

Dealing with a Functional Signature

Proposition

$(V; +)$ is not first-order interdefinable with any homogeneous structure with finite relational signature.

Difficulties:

- we have to adapt Bodirsky-Pinsker's theorem to use it on $(V; +)$
- potentially infinitely many canonical functions from $(V; +)$ to $(V; +)$

Fact

Bodirsky-Pinsker's theorem can be adapted for ω-categorical homogeneous structures **with functional signatures** which have the Ramsey property.

We first study canonical functions from $(V; +)$ to $(V; +)$.

Canonical functions of $(V; +)$ without Constants

Theorem

Let f be a unary canonical function from $(V; +)$ to $(V; +)$. There exists $h \in \mathrm{End}(V; +, \neq)$ and $a \notin h(V)$ s.t. one of the following applies:

- (id-function) $f(x) = h(x)$ for all $x \neq 0$;
- (af-function) $f(x) = h(x) + a$ for all $x \neq 0$;
- (gen-function) f sends any family of pairwise distinct elements of $V \setminus \{0\}$ to a linearly independent family;
- Degenerated case: f has an image of size at most 2.

Translation of vector $d \neq 0$: $t_d(x) := x + d$ for all x.
Fact: Translations preserve Eq_4 and are not canonical.

Useful properties

- id-functions and af-functions preserve Ieq_4 but gen-functions and translations do not.
- Let f be an injection violating Ieq_4. Then f generates a gen-function or a translation.
- $\mathrm{End}(V; \mathrm{Eq}_4, \neq)$ is generated by t_d.
- t_d together with any injection violating Eq_4 generates a gen-function.

Towards a Description of Injective Endomorphism Monoids

Translation of vector $d \neq 0$: $t_d(x) := x + d$ for all x.
Fact: Translations preserve Eq_4 and are not canonical.

Useful properties

- id-functions and af-functions preserve Ieq_4 but gen-functions and translations do not.
- Let f be an injection violating Ieq_4. Then f generates a gen-function or a translation.
- $\mathrm{End}(V; \mathrm{Eq}_4, \neq)$ is generated by t_d.
- t_d together with any injection violating Eq_4 generates a gen-function.

Classification of Injective Endomorphism Monoids of Reducts

Theorem

Let Γ be a reduct of $(V;+)$ with only injective endomorphisms. Then one of the following holds:

- $\mathrm{End}(\Gamma) = \mathrm{End}(V; \mathrm{Eq}_4, \neq)$;
- $\mathrm{End}(\Gamma)$ is contained in $\mathrm{End}(V; \mathrm{Ieq}_4, \neq)$;
- $\mathrm{End}(\Gamma)$ contains a gen-function.

Why stopping when $\mathrm{End}(\Gamma)$ contains a gen-function?

If a gen-function belongs to $\mathrm{End}(\Gamma)$, then Γ is homomorphically equivalent to a reduct of $(V; 0)$.

Classification of Injective Endomorphism Monoids of Reducts

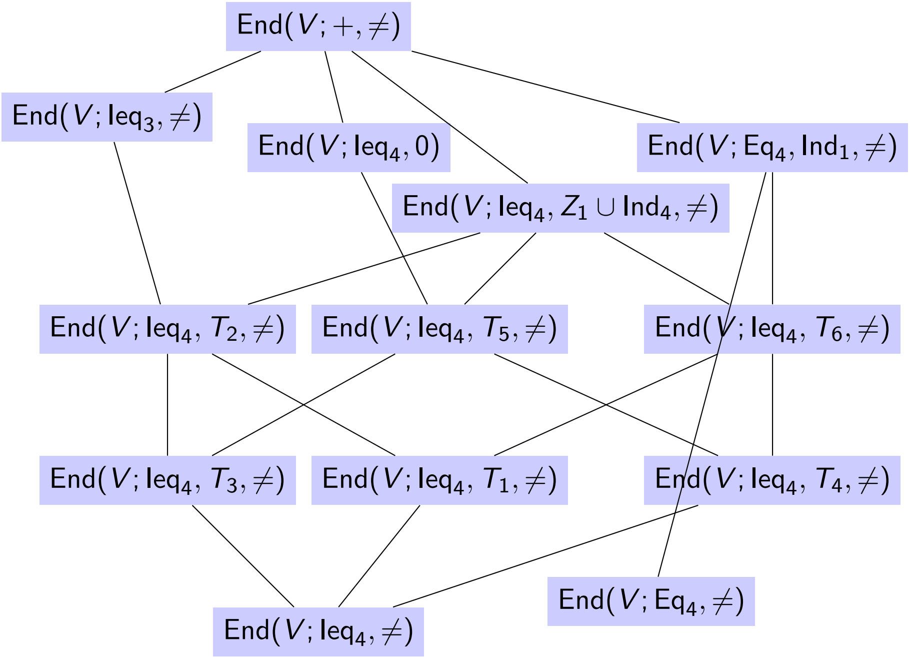
Theorem

Let Γ be a reduct of $(V; +)$ with only injective endomorphisms. Then one of the following holds:

- $\mathrm{End}(\Gamma) = \mathrm{End}(V; \mathrm{Eq}_4, \neq)$;
- $\mathrm{End}(\Gamma)$ is contained in $\mathrm{End}(V; \mathrm{Ieq}_4, \neq)$;
- $\mathrm{End}(\Gamma)$ contains a gen-function.

Why stopping when $\mathrm{End}(\Gamma)$ contains a gen-function?

If a gen-function belongs to $\mathrm{End}(\Gamma)$, then Γ is homomorphically equivalent to a reduct of $(V; 0)$.


$\mathrm{End}(V; +, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_3, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_4, 0)$
$\mathrm{End}(V; \mathrm{Eq}_4, \mathrm{Ind}_1, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_4, Z_1 \cup \mathrm{Ind}_4, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_4, T_2, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_4, T_5, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_4, T_6, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_4, T_3, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_4, T_1, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_4, T_4, \neq)$
$\mathrm{End}(V; \mathrm{Eq}_4, \neq)$
$\mathrm{End}(V; \mathrm{Ieq}_4, \neq)$

To further simplify the study, we need a new idea!

Model-complete Cores

$\overline{\mathrm{Aut}(\Delta)}$: topological closure of $\mathrm{Aut}(\Delta)$ under pointwise convergence.

Definition

A model-complete core of a reduct Γ is a structure Δ homomorphically equivalent to Γ, and such that:

$$\mathrm{End}(\Delta) = \overline{\mathrm{Aut}(\Delta)}$$

Note that $\mathrm{CSP}(\Delta)$ and $\mathrm{CSP}(\Gamma)$ are equal by homomorphic equivalence.

Theorem (Bodirsky'06)

Every reduct Γ of an ω-categorical structure has a model-complete core Δ. All model-complete cores of Γ are isomorphic to Δ.

Model-complete Cores

$\overline{\mathrm{Aut}(\Delta)}$: topological closure of $\mathrm{Aut}(\Delta)$ under pointwise convergence.

Definition

A model-complete core of a reduct Γ is a structure Δ homomorphically equivalent to Γ, and such that:

$$\mathrm{End}(\Delta) = \overline{\mathrm{Aut}(\Delta)}$$

Note that $\mathrm{CSP}(\Delta)$ and $\mathrm{CSP}(\Gamma)$ are equal by homomorphic equivalence.

Theorem (Bodirsky'06)

Every reduct Γ of an ω-categorical structure has a model-complete core Δ. All model-complete cores of Γ are isomorphic to Δ.

Classification of Model-Complete Cores

Theorem

Let Γ be a reduct of $(V;+)$. Exactly one of the following holds:

1. $\mathrm{End}(\Gamma) = \mathrm{End}(V;+,\neq)$,
2. $\mathrm{End}(\Gamma) = \mathrm{End}(V;\mathrm{Ieq}_4,0)$, or
3. the model-complete core of Γ is isomorphic to a structure Γ' s.t.:
   a) $\mathrm{End}(\Gamma') = \mathrm{End}(V \setminus \{0\};\mathrm{Ieq}_3)$,
   b) $\mathrm{End}(\Gamma') = \mathrm{End}(V;\mathrm{Eq}_4,\neq)$,
   c) Γ' is a reduct of $(V;0)$, or
   d) Γ' is a 2-element structure.

Pathway to Bit Vector CSPs

Corollary

Let Γ be a reduct of $(V;+)$. There exists a structure Γ' with same CSP as Γ and s.t. one of the following holds:

1. $\mathrm{End}(\Gamma') = \mathrm{End}(V;+,\neq)$;
2. $\mathrm{End}(\Gamma') = \mathrm{End}(V \setminus \{0\};\mathrm{Ieq}_3)$;
3. $\mathrm{End}(\Gamma') = \mathrm{End}(V;\mathrm{Eq}_4,\neq)$;
4. $\mathrm{End}(\Gamma') = \mathrm{End}(V;\mathrm{Ieq}_4,0)$;
5. Γ' is a reduct of $(V;0)$;
6. Γ' is a 2-element structure.

The study of the polymorphisms is now strongly simplified.

Pathway to Bit Vector CSPs

Corollary

Let Γ be a reduct of $(V;+)$. There exists a structure Γ' with same CSP as Γ and s.t. one of the following holds:

1. $\mathrm{End}(\Gamma') = \mathrm{End}(V;+,\neq)$;
2. $\mathrm{End}(\Gamma') = \mathrm{End}(V \setminus \{0\};\mathrm{Ieq}_3)$;
3. $\mathrm{End}(\Gamma') = \mathrm{End}(V;\mathrm{Eq}_4,\neq)$;
4. $\mathrm{End}(\Gamma') = \mathrm{End}(V;\mathrm{Ieq}_4,0)$;
5. Γ' is a reduct of $(V;0)$;
6. Γ' is a 2-element structure.

The study of the polymorphisms is now strongly simplified.

Case distinction of the previous corollary:

1. Γ' is a 2-element structure: P/NPc dichotomy (Schaefer'77)
2. Γ' is a reduct of $(V; 0)$: P/NPc dichotomy
   Poly. algos: **Schaefer** + ad-hoc routines
3. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Eq}_4, \neq)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
4. $\mathrm{End}(\Gamma') = \mathrm{End}(V \setminus \{0\}; \mathrm{leq}_3)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
5. $\mathrm{End}(\Gamma') = \mathrm{End}(V; +, \neq)$: partial proof for P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ???
6. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{leq}_4, 0)$: P/NPc dichotomy can be proved accordingly to Case 5.

NB: Case 1 is a joint work with Antoine Mottet and contains Bodirsky and Kara's classification of CSPs for reducts of $(\mathbb{N}; =)$ as a subcase.

Case distinction of the previous corollary:

1. Γ' is a 2-element structure: P/NPc dichotomy (Schaefer'77)
2. Γ' is a reduct of $(V; 0)$: P/NPc dichotomy
   Poly. algos: **Schaefer** + ad-hoc routines
3. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Eq}_4, \neq)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
4. $\mathrm{End}(\Gamma') = \mathrm{End}(V \setminus \{0\}; \mathrm{Ieq}_3)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
5. $\mathrm{End}(\Gamma') = \mathrm{End}(V; +, \neq)$: partial proof for P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ???
6. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Ieq}_4, 0)$: P/NPc dichotomy can be proved accordingly to Case 5.

NB: Case 1 is a joint work with Antoine Mottet and contains Bodirsky and Kara's classification of CSPs for reducts of $(\mathbb{N}; =)$ as a subcase.

Case distinction of the previous corollary:

1. Γ' is a 2-element structure: P/NPc dichotomy (Schaefer'77)
2. Γ' is a reduct of $(V; 0)$: P/NPc dichotomy
   Poly. algos: **Schaefer** + ad-hoc routines
3. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Eq}_4, \neq)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
4. $\mathrm{End}(\Gamma') = \mathrm{End}(V \setminus \{0\}; \mathrm{Ieq}_3)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
5. $\mathrm{End}(\Gamma') = \mathrm{End}(V; +, \neq)$: partial proof for P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ???
6. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Ieq}_4, 0)$: P/NPc dichotomy can be proved accordingly to Case 5.

NB: Case 1 is a joint work with Antoine Mottet and contains Bodirsky and Kara's classification of CSPs for reducts of $(\mathbb{N}; =)$ as a subcase.

Case distinction of the previous corollary:

1. Γ' is a 2-element structure: P/NPc dichotomy (Schaefer'77)
2. Γ' is a reduct of $(V; 0)$: P/NPc dichotomy
   Poly. algos: **Schaefer** + ad-hoc routines
3. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Eq}_4, \neq)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
4. $\mathrm{End}(\Gamma') = \mathrm{End}(V \setminus \{0\}; \mathrm{Ieq}_3)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
5. $\mathrm{End}(\Gamma') = \mathrm{End}(V; +, \neq)$: partial proof for P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ???
6. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Ieq}_4, 0)$: P/NPc dichotomy can be proved accordingly to Case 5.

NB: Case 1 is a joint work with Antoine Mottet and contains Bodirsky and Kara's classification of CSPs for reducts of $(\mathbb{N}; =)$ as a subcase.

Case distinction of the previous corollary:

1. Γ' is a 2-element structure: P/NPc dichotomy (Schaefer'77)
2. Γ' is a reduct of $(V; 0)$: P/NPc dichotomy
   Poly. algos: **Schaefer** + ad-hoc routines
3. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Eq}_4, \neq)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
4. $\mathrm{End}(\Gamma') = \mathrm{End}(V \setminus \{0\}; \mathrm{Ieq}_3)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
5. $\mathrm{End}(\Gamma') = \mathrm{End}(V; +, \neq)$: partial proof for P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ???
6. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Ieq}_4, 0)$: P/NPc dichotomy can be proved accordingly to Case 5.

NB: Case 1 is a joint work with Antoine Mottet and contains Bodirsky and Kara's classification of CSPs for reducts of $(\mathbb{N}; =)$ as a subcase.

Case distinction of the previous corollary:

1. Γ' is a 2-element structure: P/NPc dichotomy (Schaefer'77)
2. Γ' is a reduct of $(V; 0)$: P/NPc dichotomy
   Poly. algos: **Schaefer** + ad-hoc routines
3. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Eq}_4, \neq)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
4. $\mathrm{End}(\Gamma') = \mathrm{End}(V \setminus \{0\}; \mathrm{Ieq}_3)$: P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ad-hoc routines.
5. $\mathrm{End}(\Gamma') = \mathrm{End}(V; +, \neq)$: partial proof for P/NPc dichotomy
   Poly. algos: **Schaefer + Gauss Pivot** + ???
6. $\mathrm{End}(\Gamma') = \mathrm{End}(V; \mathrm{Ieq}_4, 0)$: P/NPc dichotomy can be proved accordingly to Case 5.

NB: Case 1 is a joint work with Antoine Mottet and contains Bodirsky and Kara's classification of CSPs for reducts of $(\mathbb{N}; =)$ as a subcase.

Conclusion and Perspectives

What we have achieved:

- adapt Bodirsky-Pinsker's method for **functional signatures**;
- classification of **canonical functions** from $(V; +)$ to $(V; +)$;
- classification of **endomorphism monoids of reducts** of $(V; +)$ which are not homorphically equivalent to reducts of $(V; 0)$;
- classification of **model-complete cores of reducts** of $(V; +)$ up to existential positive interdefinability. There are **finitely many**;
- **P/NPc dichotomy for Bit Vector CSPs** in 4 out of 6 listed cases.

What remains to be done:

- prove dichotomy when $\mathrm{End}(\Gamma) = \mathrm{End}(V; +, \neq)$;
- generalize the dichotomy for any vector space over **a finite field**; the **automorphism groups classification** is already established in this setting (Bodor-Kalina-Szabó'15)
- study reducts of the **atomless Boolean Algebra**;
  NB: $(V; +)$ is one of its reducts, as $x + y := (x \cup y) \setminus (x \cap y)$.

Conclusion and Perspectives

What we have achieved:

- adapt Bodirsky-Pinsker's method for **functional signatures**;
- classification of **canonical functions** from $(V;+)$ to $(V;+)$;
- classification of **endomorphism monoids of reducts** of $(V;+)$ which are not homorphically equivalent to reducts of $(V;0)$;
- classification of **model-complete cores of reducts** of $(V;+)$ up to existential positive interdefinability. There are **finitely many**;
- **P/NPc dichotomy for Bit Vector CSPs** in 4 out of 6 listed cases.

What remains to be done:

- prove dichotomy when $\mathrm{End}(\Gamma) = \mathrm{End}(V;+,\neq)$;
- generalize the dichotomy for any vector space over **a finite field**; the **automorphism groups classification** is already established in this setting (Bodor-Kalina-Szabó'15)
- study reducts of the **atomless Boolean Algebra**;
  NB: $(V;+)$ is one of its reducts, as $x+y := (x \cup y) \setminus (x \cap y)$.

Thank you!