Computable Model Theory and Primitive Recursiveness

Kalimullin I.Sh.

Kazan Federal University
e-mail:ikalimul@gmail.com

Mathematical Logic Seminar
Moscow State University, October 9, 2019

Computable structures

- A structure $\mathcal{A}$ is computable if the $|\mathcal{A}| \subseteq \mathbb{N}$ is computable and all relations and operations from $\mathcal{A}$ are computable.

Computable structures

- A structure $\mathcal{A}$ is computable if the $|\mathcal{A}| \subseteq \mathbb{N}$ is computable and all relations and operations from $\mathcal{A}$ are computable.
- A computable presentation of a countable structure $\mathcal{A}$ is any computable isomorphic copy of $\mathcal{A}$.

Computable structures

- A structure $\mathcal{A}$ is computable if the $|\mathcal{A}| \subseteq \mathbb{N}$ is computable and all relations and operations from $\mathcal{A}$ are computable.
- A computable presentation of a countable structure $\mathcal{A}$ is any computable isomorphic copy of $\mathcal{A}$.

Examples.

- $(\mathbb{N}, +, \times)$ is computable;

Computable structures

- A structure $\mathcal{A}$ is computable if the $|\mathcal{A}| \subseteq \mathbb{N}$ is computable and all relations and operations from $\mathcal{A}$ are computable.
- A computable presentation of a countable structure $\mathcal{A}$ is any computable isomorphic copy of $\mathcal{A}$.

Examples.

- $(\mathbb{N}, +, \times)$ is computable;
- $(\mathbb{Q}, +, -, \times, \div)$ has a computable presentation.

Primitive recursive structures

- A structure $\mathcal{A}$ is primitive recursive if the domain $|\mathcal{A}|$ is primitive recursive and all relations and operations in $\mathcal{A}$ are primitive recursive.

Primitive recursive structures

- A structure $\mathcal{A}$ is primitive recursive if the domain $|\mathcal{A}|$ is primitive recursive and all relations and operations in $\mathcal{A}$ are primitive recursive.
- (from Kleene's Normal Form). Every computable structure in a relational language is isomorphic to some primitive recursive structure.

Primitive recursive structures

- A structure $\mathcal{A}$ is primitive recursive if the domain $|\mathcal{A}|$ is primitive recursive and all relations and operations in $\mathcal{A}$ are primitive recursive.
- (from Kleene's Normal Form). Every computable structure in a relational language is isomorphic to some primitive recursive structure.
- The same holds in locally finite structures (Alaev, 2018).

Punctual structures

- A structure $\mathcal{A}$ is punctual (fully primitive recursive) $|\mathcal{A}| = \mathbb{N}$ and all relations and operations in $\mathcal{A}$ are primitive recursive.

Punctual structures

- A structure $\mathcal{A}$ is punctual (fully primitive recursive) $|\mathcal{A}| = \mathbb{N}$ and all relations and operations in $\mathcal{A}$ are primitive recursive.
- A punctual presentation of a countable structure $\mathcal{A}$ is any punctual isomorphic copy of $\mathcal{A}$.

Punctual structures

- A structure $\mathcal{A}$ is punctual (fully primitive recursive) $|\mathcal{A}| = \mathbb{N}$ and all relations and operations in $\mathcal{A}$ are primitive recursive.
- A punctual presentation of a countable structure $\mathcal{A}$ is any punctual isomorphic copy of $\mathcal{A}$.

Examples.

- $(\mathbb{N}, +, \times)$ is punctual;

Punctual structures

- A structure $\mathcal{A}$ is punctual (fully primitive recursive) $|\mathcal{A}| = \mathbb{N}$ and all relations and operations in $\mathcal{A}$ are primitive recursive.
- A punctual presentation of a countable structure $\mathcal{A}$ is any punctual isomorphic copy of $\mathcal{A}$.

Examples.

- $(\mathbb{N}, +, \times)$ is punctual;
- $(\mathbb{Q}, +, -, \times, \div)$ has a punctual presentation;

Punctual structures

- A structure $\mathcal{A}$ is punctual (fully primitive recursive) $|\mathcal{A}| = \mathbb{N}$ and all relations and operations in $\mathcal{A}$ are primitive recursive.
- A punctual presentation of a countable structure $\mathcal{A}$ is any punctual isomorphic copy of $\mathcal{A}$.

Examples.

- $(\mathbb{N}, +, \times)$ is punctual;
- $(\mathbb{Q}, +, -, \times, \div)$ has a punctual presentation;
- $(\mathbb{N}, +1, P(x))$ has a punctual presentation iff $P(x)$ is primitive recursive.

Existence of punctual presentations

Theorem. Every computable

- Equivalence structure (Cenzer, Remmel)
- Linear orders (Grigorieff)
- Torsion-free abelian groups (KMN)
- Boolean algebras (KMN)
- Abelian p-groups (KMN)

has a punctual presentation.

Structures without punctual presentations

Theorem. There are computable

- F.g. groups (Cannonito) and f.p. groups (Gatterdam)
- Torsion abelian groups (Cenzer, Remmel)
- Archimedean ordered abelian groups (KMN)
- Undirected graphs (KMN)

which have no punctual presentations.

Index set of structures with punctual presentations

Let $\{\mathcal{C}_e\}_{e\in\omega}$ be the Gödel numbering of (partial) computable structures.

Index set of structures with punctual presentations

Let $\{\mathcal{C}_e\}_{e\in\omega}$ be the Gödel numbering of (partial) computable structures.

Theorem. ($\mathrm{BH_TKMN}$). The sets of indices

$$I_{Pr} = \{e \in \omega : \mathcal{C}_e \text{ has a punctual presentation}\}$$

and

$$I_{Au} = \{e \in \omega : \mathcal{C}_e \text{ has an automatic presentation}\}.$$

are Σ^1_1 complete.

Game $G^{\infty}(\mathbb{K})$ (Montalban)

Let $\mathbb{K}$ be a class of infinite structures in a fixed language.

- Player C builds an infinite sequence of structures $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2, \ldots$.
- Player D builds a structure $\mathcal{D}$.

Game $G^\infty(\mathbb{K})$ (Montalban)

Let $\mathbb{K}$ be a class of infinite structures in a fixed language.

- Player C builds an infinite sequence of structures $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2, \ldots$.
- Player D builds a structure $\mathcal{D}$.

The sequence of moves: $\mathcal{C}_0[0]$, $\mathcal{D}[0]$, $\mathcal{C}_0[1]$, $\mathcal{C}_1[1]$, $\mathcal{D}[1]$, $\mathcal{C}_0[2]$, $\mathcal{C}_1[2]$, $\mathcal{C}_2[2]$, $\mathcal{D}[2]$, ...

Game $G^{\infty}(\mathbb{K})$ (Montalban)

Let $\mathbb{K}$ be a class of infinite structures in a fixed language.

- Player C builds an infinite sequence of structures $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2, \dots$.
- Player D builds a structure $\mathcal{D}$.

The sequence of moves: $\mathcal{C}_0[0]$, $\mathcal{D}[0]$, $\mathcal{C}_0[1]$, $\mathcal{C}_1[1]$, $\mathcal{D}[1]$, $\mathcal{C}_0[2]$, $\mathcal{C}_1[2]$, $\mathcal{C}_2[2]$, $\mathcal{D}[2]$, ...

The rules:

1. If $\mathcal{C}_i \notin \mathbb{K}$ for some i, then D wins;
2. If $\mathcal{C}_i \in \mathbb{K}$ for every i and $\mathcal{D} \notin \mathbb{K}$, then C wins;
3. If $\mathcal{D}, \mathcal{C}_0, \mathcal{C}_1, \dots \in \mathbb{K}$ and $\mathcal{C}_i \not\cong \mathcal{D}$ for every i then D wins;
4. If $\mathcal{D}, \mathcal{C}_0, \mathcal{C}_1, \dots \in \mathbb{K}$ and $\mathcal{C}_i \cong \mathcal{D}$ for some i then C wins.

∞-Copyable and ∞-diagonalizable classes

If C has a computable winning strategy in $G^\infty(\mathbb{K})$ then $\mathbb{K}$ is ∞-copyable.
If D has a computable winning strategy in $G^\infty(\mathbb{K})$ then $\mathbb{K}$ is ∞-diagonalizable.

∞-Copyable and ∞-diagonalizable classes

If C has a computable winning strategy in $G^{\infty}(\mathbb{K})$ then $\mathbb{K}$ is ∞-copyable.
If D has a computable winning strategy in $G^{\infty}(\mathbb{K})$ then $\mathbb{K}$ is ∞-diagonalizable.

Examples. (Montalban)

1. The class of infinite linear orderings is ∞-copyable.
2. The class of infinite undirected graphs is ∞-diagonalizable.

∞-Copyable classes

Corollary. The classes of infinite

- equivalence structures
- linear orders
- torsion-free abelian groups
- Boolean algebras
- Abelian p-groups

are ∞-copyable.

∞-Diagonalizable classes

Corollary. The classes of infinite

- Torsion abelian groups
- Archimedean ordered abelian groups
- Undirected graphs

are ∞-digonalizable.

An ∞-diagonolizable class with punctual presentations.

Let $\mathbb{LS}$ be the class of infinite linear orderings with the successor relation

$$S(x,y) \iff x < y \,\&\, \neg\exists z[x < z \,\&\, z < y]$$

Then $\mathbb{LS}$ is ∞-diagonolizable (Montalban).

An ∞-diagonolizable class with punctual presentations.

Let $\mathbb{LS}$ be the class of infinite linear orderings with the successor relation

$$S(x, y) \iff x < y \ \& \ \neg\exists z[x < z \ \& \ z < y]$$

Then $\mathbb{LS}$ is ∞-diagonolizable (Montalban).

But every computable member of $\mathbb{LS}$ has a punctual presentation by almost the same proof as for usual linear orderings.

The game $P(\mathbb{K})$

The player C builds an infinite sequence $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2, \ldots$. On the s-th move C works only with $\mathcal{C}_i$, $i \leq s$.

The player D builds $\mathcal{D}$.

The rules:

1. If $\mathcal{C}_i$ is finite for some i, then D wins;
2. If $\mathcal{C}_0, \mathcal{C}_1, \ldots$ are infinite for every i and $\mathcal{D} \notin \mathbb{K}$, then C wins;
3. If $\mathcal{D} \in \mathbb{K}$, $\mathcal{C}_0, \mathcal{C}_1, \ldots$ are infinite, and $\mathcal{C}_i \ncong \mathcal{D}$ for every i then D wins;
4. If $\mathcal{D} \in \mathbb{K}$, $\mathcal{C}_0, \mathcal{C}_1, \ldots$ are infinite, and $\mathcal{C}_i \cong \mathcal{D}$ for some i then C wins.

P-Copyable and P-diagonalizable classes

If C has a computable winning strategy in $P(\mathbb{K})$ then $\mathbb{K}$ is P-copyable.
If D has a computable winning strategy in $P(\mathbb{K})$ then $\mathbb{K}$ is P-diagonalizable.

P-Copyable and P-diagonalizable classes

If C has a computable winning strategy in $P(\mathbb{K})$ then $\mathbb{K}$ is P-copyable.
If D has a computable winning strategy in $P(\mathbb{K})$ then $\mathbb{K}$ is P-diagonalizable.

Observation.

1. If $\mathbb{K}$ is P-diagonalizable then $\mathbb{K}$ is ∞-diagonalizable.

P-Copyable and P-diagonalizable classes

If C has a computable winning strategy in $P(\mathbb{K})$ then $\mathbb{K}$ is P-copyable.
If D has a computable winning strategy in $P(\mathbb{K})$ then $\mathbb{K}$ is P-diagonalizable.

Observation.

1. If $\mathbb{K}$ is P-diagonalizable then $\mathbb{K}$ is ∞-diagonalizable.
2. If $\mathbb{K}$ is ∞-copyable then $\mathbb{K}$ is P-copyable.

P-Copyable and P-diagonalizable classes

Observation.

1. If $\mathbb{K}$ is P-diagonalizable then there is a computable member of $\mathbb{K}$ which has no punctual presentation.

P-Copyable and P-diagonalizable classes

Observation.

1. If $\mathbb{K}$ is P-diagonalizable then there is a computable member of $\mathbb{K}$ which has no punctual presentation.
2. If $\mathbb{K}$ is P-copyable then there is a computable function F such that every computable member of $\mathbb{K}$ has a presentation which is punctual relative to F.

P-Copyable and P-diagonalizable classes

Observation.

1. If $\mathbb{K}$ is P-diagonalizable then there is a computable member of $\mathbb{K}$ which has no punctual presentation.
2. If $\mathbb{K}$ is P-copyable then there is a computable function F such that every computable member of $\mathbb{K}$ has a presentation which is punctual relative to F.

Question. Is there a natural P-copyable class which has a computable member without a punctual copy?

Uniformity in building punctual presentations

Theorem. (FKMZ).

- There is an effective procedure which transforms a computable index of a Boolean algebra with infinitely many atoms to a punctual index of its isomorphic copy.

Uniformity in building punctual presentations

Theorem. (FKMZ).

- There is an effective procedure which transforms a computable index of a Boolean algebra with infinitely many atoms to a punctual index of its isomorphic copy.
- There is a $\emptyset''$-computable procedure which does the same for infinite Abelian p-groups.

Uniformity in building punctual presentations

Theorem. (FKMZ).

- There is an effective procedure which transforms a computable index of a Boolean algebra with infinitely many atoms to a punctual index of its isomorphic copy.
- There is a $\emptyset''$-computable procedure which does the same for infinite Abelian p-groups. Such procedure can not be $\emptyset'$-computable.
- There is a $\emptyset'''$-computable procedure which does the same for infinite linear orders. Such procedure can not be $\emptyset''$-computable.

No upper bound for all structures in general

Theorem. (FKMZ). Every hyperarithmetic procedure ψ on indices fails to find a punctual presentation of some computable structure for which a punctual presentation exists

The game $G^{\alpha}(\mathbb{K})$ (Montalban, 2013)

Here α is a constructive ordinal.

Now the player C again builds an infinite sequence $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2, \ldots$ but also pick from the beginning an index e of a Turing operator φ_e.

The game $G^\alpha(\mathbb{K})$ (Montalban, 2013)

Here α is a constructive ordinal.

Now the player C again builds an infinite sequence $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2, \ldots$ but also pick from the beginning an index e of a Turing operator φ_e.

The rules:

1. If $\mathcal{C}_i \notin \mathbb{K}$ for some i, then D wins;
2. If $\mathcal{C}_i \in \mathbb{K}$ for every i and $\mathcal{D} \notin \mathbb{K}$, then C wins;

The game $G^\alpha(\mathbb{K})$ (Montalban, 2013)

Here α is a constructive ordinal.

Now the player C again builds an infinite sequence $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2, \dots$ but also pick from the beginning an index e of a Turing operator φ_e.

The rules:

1. If $\mathcal{C}_i \notin \mathbb{K}$ for some i, then D wins;
2. If $\mathcal{C}_i \in \mathbb{K}$ for every i and $\mathcal{D} \notin \mathbb{K}$, then C wins;
3. If $\mathcal{D}, \mathcal{C}_0, \mathcal{C}_1, \dots \in \mathbb{K}$ and $\varphi_e(\mathcal{D}^{(\alpha)}; 0)\uparrow$ then D wins.

The game $G^\alpha(\mathbb{K})$ (Montalban, 2013)

Here α is a constructive ordinal.

Now the player C again builds an infinite sequence $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2, \dots$ but also pick from the beginning an index e of a Turing operator φ_e.

The rules:

1. If $\mathcal{C}_i \notin \mathbb{K}$ for some i, then D wins;
2. If $\mathcal{C}_i \in \mathbb{K}$ for every i and $\mathcal{D} \notin \mathbb{K}$, then C wins;
3. If $\mathcal{D}, \mathcal{C}_0, \mathcal{C}_1, \dots \in \mathbb{K}$ and $\varphi_e(\mathcal{D}^{(\alpha)}; 0)\uparrow$ then D wins. Otherwise, let $i_0 = \varphi_e(\mathcal{D}^{(\alpha)}; 0)$.

The game $G^\alpha(\mathbb{K})$ (Montalban, 2013)

Here α is a constructive ordinal.

Now the player C again builds an infinite sequence $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2, \dots$ but also pick from the beginning an index e of a Turing operator φ_e.

The rules:

1. If $\mathcal{C}_i \notin \mathbb{K}$ for some i, then D wins;
2. If $\mathcal{C}_i \in \mathbb{K}$ for every i and $\mathcal{D} \notin \mathbb{K}$, then C wins;
3. If $\mathcal{D}, \mathcal{C}_0, \mathcal{C}_1, \dots \in \mathbb{K}$ and $\varphi_e(\mathcal{D}^{(\alpha)}; 0)\uparrow$ then D wins. Otherwise, let $i_0 = \varphi_e(\mathcal{D}^{(\alpha)}; 0)$.
4. If $\mathcal{D}, \mathcal{C}_0, \mathcal{C}_1, \dots \in \mathbb{K}$ and $\mathcal{C}_{i_0} \ncong \mathcal{D}$ then D wins;
5. If $\mathcal{D}, \mathcal{C}_0, \mathcal{C}_1, \dots \in \mathbb{K}$ and $\mathcal{C}_{i_0} \cong \mathcal{D}$ then C wins.

α-Copyable and α-diagonalizable classes

If C has a computable winning strategy in $G^{\alpha}(\mathbb{K})$ then $\mathbb{K}$ is α-copyable.
If D has a computable winning strategy in $G^{\infty}(\mathbb{K})$ then $\mathbb{K}$ is α-diagonalizable.

α-Copyable and α-diagonalizable classes

If C has a computable winning strategy in $G^{\alpha}(\mathbb{K})$ then $\mathbb{K}$ is α-copyable.
If D has a computable winning strategy in $G^{\infty}(\mathbb{K})$ then $\mathbb{K}$ is α-diagonalizable.

Observation.
0-copyable $\implies$ 1-copyable $\implies$ 2-copyable $\implies \cdots \implies$ ∞-copyable.

α-Copyable and α-diagonalizable classes

If C has a computable winning strategy in $G^{\alpha}(\mathbb{K})$ then $\mathbb{K}$ is α-copyable.
If D has a computable winning strategy in $G^{\infty}(\mathbb{K})$ then $\mathbb{K}$ is α-diagonalizable.

Observation.
0-copyable $\implies$ 1-copyable $\implies$ 2-copyable $\implies \cdots \implies$ ∞-copyable.

∞-diagonalizable $\implies \cdots \implies$ 2-diagonalizable $\implies$ 1-diagonalizable $\implies$ 0-diagonolizable.

Corollaries

- The class of Boolean algebras with infinitely many atoms is $\mathbf{0}$-copyable.
- The class of infinite Abelian p-groups is $\mathbf{2}$-copyable and $\mathbf{1}$-diagonalizable.
- The class of infinite linear orders $\mathbf{3}$-copyable (Harrison-Trainor) and $\mathbf{2}$-diagonalizable (Montalban).

Corollaries

- The class of Boolean algebras with infinitely many atoms is $\mathbf{0}$-copyable.
- The class of infinite Abelian p-groups is $\mathbf{2}$-copyable and $\mathbf{1}$-diagonalizable.
- The class of infinite linear orders $\mathbf{3}$-copyable (Harrison-Trainor) and $\mathbf{2}$-diagonalizable (Montalban).
- There is a $\forall\exists\forall$-axiomatizable class of structures which is ∞-copyable and α-diagonalizable for every constructive α.

R.i.c.e. relations

Theorem. (AKMS). A relation $R(\vec{x})$ on $\mathcal{A}$ is computable in every copy of $\mathcal{A}$ iff there are some $\vec{a}$ from $\mathcal{A}$ and a computable mappings

$$i \mapsto \Phi_i(\vec{a}, \vec{b}_i, \vec{x})$$

$$i \mapsto \Psi_i(\vec{a}, \vec{c}_i, \vec{x})$$

into the quantifier-free formulae, such that

$$R(\vec{x}) \iff \mathcal{A} \models \bigvee_i (\exists \vec{b}_i) \Phi_i(\vec{a}, \vec{b}_i, \vec{x});$$

$$\neg R(\vec{x}) \iff \mathcal{A} \models \bigvee_i (\exists \vec{c}_i) \Psi_i(\vec{a}, \vec{c}_i, \vec{x}).$$

R.i.p. relations

Theorem??? (AKMS) A relation $R(\vec{x})$ on $\mathcal{A}$ is primitive recursive in every copy of $\mathcal{A}$ iff for some $\vec{a}$ and a quantifier-free formula $\Phi(\vec{a}, \vec{x})$ we have

$$R(\vec{x}) \iff \mathcal{A} \models \Phi(\vec{a}, \vec{x}).$$

Turing content coded into structures

Folklore Theorem. Suppose $f \in \omega^\omega$ and $f \leq_T \mathcal{U}$ for every $\mathcal{U} \cong \mathcal{A}$. Then there is a tuple $\vec{a} \in \mathcal{A}^{<\omega}$ and a computable mapping

$$x, y, i \mapsto \Phi_{x,y,i}(\vec{a}, \vec{b}_i)$$

into the quantifier-free formulae, such that

$$f(x) = y \iff \mathcal{A} \models \bigvee_i (\exists \vec{b}_i) \Phi_{x,y,i}(\vec{a}, \vec{b}_i).$$

Punctual content coded into structures

Theorem. (KMM). Suppose $f \in \omega^\omega$ and $f \leq_{PR} \mathcal{U}$ for every $\mathcal{U} \cong \mathcal{A}$. Then
1) f is bounded by a primitive recursive function, and
2) there is a tuple $\vec{a} \in \mathcal{A}^{<\omega}$ and a primitive recursive mapping

$$x, y \mapsto \Phi_{x,y}(\vec{a}, \vec{b})$$

into the quantifier-free formulae, such that

$$f(x) = y \iff \mathcal{A} \models (\exists b_1, \dots, b_n)[\&_{i<j}(b_i \neq b_j) \;\&\; \Phi_{x,y}(\vec{a}, b_1, \dots, b_n)],$$

and

$$f(x) = y \iff \mathcal{A} \models (\forall b_1, \dots, b_n)[\&_{i<j}(b_i \neq b_j) \to \Phi_{x,y}(\vec{a}, b_1, \dots, b_n)].$$

Relational languages are not universal

Corollary. (KMM). Suppose $f \in \omega^\omega$, $\mathcal{A}$ is relational, and $f \leq_{PR} \mathcal{U}$ for every $\mathcal{U} \cong \mathcal{A}$. Then f is primitive recursive.

Relational languages are not universal

Corollary. (KMM). Suppose $f \in \omega^\omega$, $\mathcal{A}$ is relational, and $f \leq_{PR} \mathcal{U}$ for every $\mathcal{U} \cong \mathcal{A}$. Then f is primitive recursive.

Proof. The use of constants $\vec{a}$ in formulae can be eliminated by new relations. By Ramsey Theorem every relational structure contains an infinite substructure having punctual presentation, in fact, isomorphic to a quantifier-free interpretation in $(\omega, <)$.

Relational languages are not universal

Corollary. (KMM). Suppose $f \in \omega^\omega$, $\mathcal{A}$ is relational, and $f \leq_{PR} \mathcal{U}$ for every $\mathcal{U} \cong \mathcal{A}$. Then f is primitive recursive.

Proof. The use of constants $\vec{a}$ in formulae can be eliminated by new relations. By Ramsey Theorem every relational structure contains an infinite substructure having punctual presentation, in fact, isomorphic to a quantifier-free interpretation in $(\omega, <)$.

Corollary. (KMM). Relational structures are not punctually universal.

Note that one binary operation is universal ($\mathrm{DH_TKMT}$).

Computably categorical structures

- Computable structures $\mathcal{A}$ and $\mathcal{B}$ are computably isomorphic if there is a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.

Computably categorical structures

- Computable structures $\mathcal{A}$ and $\mathcal{B}$ are computably isomorphic if there is a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.
- A computable structure $\mathcal{B}$ is computably categorical if $\mathcal{A}$ is computably isomorphic to each of its computable presentations.

Computably categorical structures

- Computable structures $\mathcal{A}$ and $\mathcal{B}$ are computably isomorphic if there is a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.
- A computable structure $\mathcal{B}$ is computably categorical if $\mathcal{A}$ is computably isomorphic to each of its computable presentations.

Examples.

- $(\mathbb{N}, +, \times)$ and $(\mathbb{Q}, +, -, \times, \div)$ are computably categorical;

Computably categorical structures

- Computable structures $\mathcal{A}$ and $\mathcal{B}$ are computably isomorphic if there is a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.
- A computable structure $\mathcal{B}$ is computably categorical if $\mathcal{A}$ is computably isomorphic to each of its computable presentations.

Examples.

- $(\mathbb{N}, +, \times)$ and $(\mathbb{Q}, +, -, \times, \div)$ are computably categorical;
- $(\mathbb{Q}, <)$ is computably categorical;

Computably categorical structures

- Computable structures $\mathcal{A}$ and $\mathcal{B}$ are computably isomorphic if there is a computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.
- A computable structure $\mathcal{B}$ is computably categorical if $\mathcal{A}$ is computably isomorphic to each of its computable presentations.

Examples.

- $(\mathbb{N}, +, \times)$ and $(\mathbb{Q}, +, -, \times, \div)$ are computably categorical;
- $(\mathbb{Q}, <)$ is computably categorical;
- $(\mathbb{N}, <)$ is not computably categorical.

Relatively computably categorical structures

- A computable structure $\mathcal{A}$ is relatively computably categorical if for any $\mathcal{B} \cong \mathcal{A}$ there is an isomorphism $f \leq_T \mathcal{B}$ from $\mathcal{A}$ onto $\mathcal{B}$.

Relatively computably categorical structures

- A computable structure $\mathcal{A}$ is relatively computably categorical if for any $\mathcal{B} \cong \mathcal{A}$ there is an isomorphism $f \leq_T \mathcal{B}$ from $\mathcal{A}$ onto $\mathcal{B}$.
- (Goncharov) A computable structure $\mathcal{A}$ is relatively computably categorical if and only if there is a computable sequence of $\exists$-formulae defining all possible automorphism orbits of tuples in $(\mathcal{A}, \vec{a})$.

Relatively computably categorical structures

- A computable structure $\mathcal{A}$ is relatively computably categorical if for any $\mathcal{B} \cong \mathcal{A}$ there is an isomorphism $f \leq_T \mathcal{B}$ from $\mathcal{A}$ onto $\mathcal{B}$.
- (Goncharov) A computable structure $\mathcal{A}$ is relatively computably categorical if and only if there is a computable sequence of ∃-formulae defining all possible automorphism orbits of tuples in $(\mathcal{A}, \vec{a})$.
- For many natural classes of structures (equivalence structures, linear orderings, boolean algebras, abelian p-groups, torsion-free abelian groups)

computable categoricity = relative computable categoricity.

Triviality of punctually categorical structures

- Punctual structures $\mathcal{A}$ and $\mathcal{B}$ are punctually isomorphic if there is an isomorphism p from $\mathcal{A}$ onto $\mathcal{B}$ such that both p and p^{-1} are primitively recursive.

Triviality of punctually categorical structures

- Punctual structures $\mathcal{A}$ and $\mathcal{B}$ are punctually isomorphic if there is an isomorphism p from $\mathcal{A}$ onto $\mathcal{B}$ such that both p and p^{-1} are primitively recursive.
- A punctual structure $\mathcal{A}$ is punctually categorical if $\mathcal{A}$ is punctually isomorphic to every its punctual presentation.

Triviality of punctually categorical structures

- Punctual structures $\mathcal{A}$ and $\mathcal{B}$ are punctually isomorphic if there is an isomorphism p from $\mathcal{A}$ onto $\mathcal{B}$ such that both p and p^{-1} are primitively recursive.
- A punctual structure $\mathcal{A}$ is punctually categorical if $\mathcal{A}$ is punctually isomorphic to every its punctual presentation.

Examples.

- The structure in empty signature is punctually categorical.

Triviality of punctually categorical structures

- Punctual structures $\mathcal{A}$ and $\mathcal{B}$ are punctually isomorphic if there is an isomorphism p from $\mathcal{A}$ onto $\mathcal{B}$ such that both p and p^{-1} are primitively recursive.
- A punctual structure $\mathcal{A}$ is punctually categorical if $\mathcal{A}$ is punctually isomorphic to every its punctual presentation.

Examples.

- The structure in empty signature is punctually categorical.
- $(\mathbb{N}, +1)$, $(\mathbb{N}, +, \times)$ and $(\mathbb{Q}, +, -, \times, \div)$ are not punctually categorical;

Triviality of punctually categorical structures

- Punctual structures $\mathcal{A}$ and $\mathcal{B}$ are punctually isomorphic if there is an isomorphism p from $\mathcal{A}$ onto $\mathcal{B}$ such that both p and p^{-1} are primitively recursive.
- A punctual structure $\mathcal{A}$ is punctually categorical if $\mathcal{A}$ is punctually isomorphic to every its punctual presentation.

Examples.

- The structure in empty signature is punctually categorical.
- $(\mathbb{N}, +1)$, $(\mathbb{N}, +, \times)$ and $(\mathbb{Q}, +, -, \times, \div)$ are not punctually categorical;
- $(\mathbb{Q}, <)$ is not punctually categorical;

Existence of categorical structures

Theorem. (KMN).

- There is no rigid punctually categorical structure in relational language.

Existence of categorical structures

Theorem. (KMN).

- There is no rigid punctually categorical structure in relational language.
- There is no punctually categorical linear ordering.

Existence of categorical structures

Theorem. (KMN).

- There is no rigid punctually categorical structure in relational language.
- There is no punctually categorical linear ordering.
- There is no punctually categorical Boolean algebra.

Existence of categorical structures

Theorem. (KMN).

- There is no rigid punctually categorical structure in relational language.
- There is no punctually categorical linear ordering.
- There is no punctually categorical Boolean algebra.
- There is no punctually torsion-free abelian group.

Existence of categorical structures

Theorem. (KMN).

- There is no rigid punctually categorical structure in relational language.
- There is no punctually categorical linear ordering.
- There is no punctually categorical Boolean algebra.
- There is no punctually torsion-free abelian group.
- ($\mathrm{DH_TKMT}$). An undirected graph $\mathcal{G}$ is punctually categorical iff it is is a clique or an anti-clique up to finitely many vertices each of which is either connected, or disconnected with the (anti-)clique.

Existence of categorical structures

Theorem. (KMN).

- There is no rigid punctually categorical structure in relational language.
- There is no punctually categorical linear ordering.
- There is no punctually categorical Boolean algebra.
- There is no punctually torsion-free abelian group.
- (DH_TKMT). An undirected graph $\mathcal{G}$ is punctually categorical iff it is is a clique or an anti-clique up to finitely many vertices each of which is either connected, or disconnected with the (anti-)clique.
- An abelian p-group is punctually categorical iff it has the form $\bigoplus_i \mathbb{Z}_{p^{n_i}}$, where $n_i = 1$ for almost all i.

Relatively computably categorical structures

- A computable structure $\mathcal{A}$ is relatively punctually categorical if for any $\mathcal{B} \cong \mathcal{A}$ there is an isomorphism $f \leq_{PR} \mathcal{B}$ from $\mathcal{A}$ onto $\mathcal{B}$ such that $f^{-1} \leq_{PR} \mathcal{B}$.

Relatively computably categorical structures

- A computable structure $\mathcal{A}$ is relatively punctually categorical if for any $\mathcal{B} \cong \mathcal{A}$ there is an isomorphism $f \leq_{PR} \mathcal{B}$ from $\mathcal{A}$ onto $\mathcal{B}$ such that $f^{-1} \leq_{PR} \mathcal{B}$.
- (KMM) A punctual structure $\mathcal{A}$ in a relational language is relatively punctually categorical iff $\mathcal{A}$ is automorphically trivial, i.e., there is some $\vec{a}$ s.t. $(\mathcal{A}, \vec{a}, \vec{b}) \cong (\mathcal{A}, \vec{a}, \vec{c})$ for all $\vec{b}, \vec{c}$

Relatively computably categorical structures

- A computable structure $\mathcal{A}$ is relatively punctually categorical if for any $\mathcal{B} \cong \mathcal{A}$ there is an isomorphism $f \leq_{PR} \mathcal{B}$ from $\mathcal{A}$ onto $\mathcal{B}$ such that $f^{-1} \leq_{PR} \mathcal{B}$.
- (KMM) A punctual structure $\mathcal{A}$ in a relational language is relatively punctually categorical iff $\mathcal{A}$ is automorphically trivial, i.e., there is some $\vec{a}$ s.t. $(\mathcal{A}, \vec{a}, \vec{b}) \cong (\mathcal{A}, \vec{a}, \vec{c})$ for all $\vec{b}, \vec{c}$ (we expect that this holds all punctually categorical structures in a relational language).

Relatively computably categorical structures

- A computable structure $\mathcal{A}$ is relatively punctually categorical if for any $\mathcal{B} \cong \mathcal{A}$ there is an isomorphism $f \leq_{PR} \mathcal{B}$ from $\mathcal{A}$ onto $\mathcal{B}$ such that $f^{-1} \leq_{PR} \mathcal{B}$.
- (KMM) A punctual structure $\mathcal{A}$ in a relational language is relatively punctually categorical iff $\mathcal{A}$ is automorphically trivial, i.e., there is some $\vec{a}$ s.t. $(\mathcal{A}, \vec{a}, \vec{b}) \cong (\mathcal{A}, \vec{a}, \vec{c})$ for all $\vec{b}, \vec{c}$ (we expect that this holds all punctually categorical structures in a relational language).
- (KMM) Punctually categorical structures are countably categorical

Relatively computably categorical structures

- A computable structure $\mathcal{A}$ is relatively punctually categorical if for any $\mathcal{B} \cong \mathcal{A}$ there is an isomorphism $f \leq_{PR} \mathcal{B}$ from $\mathcal{A}$ onto $\mathcal{B}$ such that $f^{-1} \leq_{PR} \mathcal{B}$.
- (KMM) A punctual structure $\mathcal{A}$ in a relational language is relatively punctually categorical iff $\mathcal{A}$ is automorphically trivial, i.e., there is some $\vec{a}$ s.t. $(\mathcal{A}, \vec{a}, \vec{b}) \cong (\mathcal{A}, \vec{a}, \vec{c})$ for all $\vec{b}, \vec{c}$ (we expect that this holds all punctually categorical structures in a relational language).
- (KMM) Punctually categorical structures are countably categorical (we expect that they are totally categorical).

A non-trivial punctually categorical structure

Theorem. (KMN) There is a finitely generated punctually categorical structure $(\mathcal{A}, a^0, s^1, t^1)$ and a primitive recursive function f such that $f(2n) = 1$ and

1. every $a \in \mathcal{A}$ is uniquely presented in the form $a = t^m(s^n(a))$ for $m < f(n)$;
2. $s(t^m(s^n(a))) = s^{n+1}(a)$;
3. $t^{f(n)}(s^n(a)) = s^n(a)$.

A punctually categorical semigroup

Corollary. For some primitive recursive function f the free semigroup presented by a, b, s, t and

1. $st^m s^n a = s^{n+1} a$;
2. $t^{f(n)} s^n a = s^n a$;
3. $ts^m t^n b = t^{n+1} b$;
4. $s^{f(n)} t^n b = t^n b$;
5. $aa = ab = as = at = a$; $ba = bb = bs = bt = b$

is punctually categorical.

Appendix: A preorder on punctual presentations

- $\mathcal{A} \leq_{pr} \mathcal{B}$ if there is a primitive recursive isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.

Appendix: A preorder on punctual presentations

- $\mathcal{A} \leq_{pr} \mathcal{B}$ if there is a primitive recursive isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.
- Let $\mathbf{P}(\mathcal{A})$ be the class of all punctual presentations of $\mathcal{A}$ modulo $\leq_{pr}$.

Appendix: A preorder on punctual presentations

- $\mathcal{A} \leq_{pr} \mathcal{B}$ if there is a primitive recursive isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.
- Let $\mathbf{P}(\mathcal{A})$ be the class of all punctual presentations of $\mathcal{A}$ modulo $\leq_{pr}$.
- $\mathbf{P}(\mathbb{N}, +1)$ has $\leq_{pr}$-least element but has no $\leq_{pr}$-greatest element. The same for every finitely generated punctual structure.

Appendix: A preorder on punctual presentations

- $\mathcal{A} \leq_{pr} \mathcal{B}$ if there is a primitive recursive isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.
- Let $\mathbf{P}(\mathcal{A})$ be the class of all punctual presentations of $\mathcal{A}$ modulo $\leq_{pr}$.
- $\mathbf{P}(\mathbb{N}, +1)$ has $\leq_{pr}$-least element but has no $\leq_{pr}$-greatest element. The same for every finitely generated punctual structure.
- $\mathbf{P}(\mathbb{Q}, <)$ has $\leq_{pr}$-greatest element.

Appendix: A preorder on punctual presentations

- $\mathcal{A} \leq_{pr} \mathcal{B}$ if there is a primitive recursive isomorphism from $\mathcal{A}$ onto $\mathcal{B}$.
- Let $\mathbf{P}(\mathcal{A})$ be the class of all punctual presentations of $\mathcal{A}$ modulo $\leq_{pr}$.
- $\mathbf{P}(\mathbb{N}, +1)$ has $\leq_{pr}$-least element but has no $\leq_{pr}$-greatest element. The same for every finitely generated punctual structure.
- $\mathbf{P}(\mathbb{Q}, <)$ has $\leq_{pr}$-greatest element.
- $\mathbf{P}$(random graph) has no $\leq_{pr}$-greatest element but has maximal elements (Melnikov, Ng).

Appendix: Punctual dimension

Theorem. (Melnikov, Turetsky, ...). There is a structure with two punctual copies $\mathcal{A} \cong \mathcal{B}$ such that $\mathcal{A} \,|_{pr}\, \mathcal{B}$ and any other punctual copy $\mathcal{C}$ is primitive recursively isomorphic either to $\mathcal{A}$, or to $\mathcal{C}$.