On Computability-Theoretic Properties of Heyting Algebras

Nikolay Bazhenov

Sobolev Institute of Mathematics, Novosibirsk, Russia, and
Kazakh-British Technical University, Almaty, Kazakhstan

9th Indian Conference on Logic and its Applications
March 5, 2021

Computable structures

Here we work only with at most countable structures in finite signatures.

A structure $\mathcal{S}$ in the signature

$$\{P_0^{n_0}, P_1^{n_1}, \ldots, P_k^{n_k}; f_0^{m_0}, f_1^{m_1}, \ldots, f_\ell^{m_\ell}; c_0, c_1, \ldots, c_p\}$$

is **computable** (or recursive) if:

- the domain of $\mathcal{S}$ is a (Turing) computable subset of $\mathbb{N}$;
- the predicates $P_i^{\mathcal{S}}$ and the operations $f_j^{\mathcal{S}}$ are computable.

Example 1. The semiring of natural numbers $(\mathbb{N}; +, \cdot)$ is a computable structure.

Computable structures

Example 2. Consider the ordered field of rationals $\mathcal{Q} = (\mathbb{Q}; +, \cdot, \leq)$. Since the domain of $\mathcal{Q}$ is not a subset of $\mathbb{N}$, the field $\mathcal{Q}$ itself is not a computable structure.

Nevertheless, by effectively encoding irreducible fractions $\frac{m}{n}$ (where $m \in \mathbb{Z}$, $n \in \mathbb{N}$, and $n \neq 0$) via natural numbers, one can define a *computable isomorphic copy* of $\mathcal{Q}$.

Computable structures

Example 2. Consider the ordered field of rationals $\mathcal{Q} = (\mathbb{Q}; +, \cdot, \leq)$. Since the domain of $\mathcal{Q}$ is not a subset of $\mathbb{N}$, the field $\mathcal{Q}$ itself is not a computable structure.

Nevertheless, by effectively encoding irreducible fractions $\frac{m}{n}$ (where $m \in \mathbb{Z}$, $n \in \mathbb{N}$, and $n \neq 0$) via natural numbers, one can define a *computable isomorphic copy* of $\mathcal{Q}$.

Example 3. A finitely presented group G has a computable isomorphic copy if and only if the word problem of G is decidable.

Example 4. Let $\mathcal{B}$ be a superatomic Boolean algebra.

Goncharov (1973) proved that $\mathcal{B}$ has a computable copy if and only if the Cantor–Bendixson rank of the Stone space $\mathrm{Ult}(\mathcal{B})$ is a computable ordinal.

Heyting algebras

Consider a signature $L_{HA} = \{\vee, \wedge, \to, 0, 1\}$.

An L_{HA}-structure $\mathcal{H} = (H; \vee, \wedge, \to, 0, 1)$ is a *Heyting algebra* if $(H; \vee, \wedge, 0, 1)$ is a bounded distributive lattice, and for every $a, b \in H$, the element $a \to b$ is the greatest element in the set $\{d : a \wedge d \leq b\}$.

Examples:

- every finite distributive lattice is a Heyting algebra;
- every Boolean algebra is a Heyting algebra: $a \to b = \overline{(a)} \vee b$;
- every linear order with 0 and 1 is a Heyting algebra:

$$a \to b = \begin{cases} 1, & \text{if } a \leq b, \\ b, & \text{if } a > b. \end{cases}$$

Since 1970s, there have been a lot of works on computable linear orders and computable Boolean algebras.

Computable Heyting algebras (not Boolean and not linearly ordered) were studied in [Turlington 2010] and [B. 2017].

Motivation

As a rule of thumb, one can say the following:

(i) The class of graphs provides the richest computability-theoretic environment: every interesting example of a computable structure can be realized as a computable graph.

This property can be explicitly formalized — e.g., [Hirschfeldt, Khoussainov, Shore, and Slinko 2002].

(ii) Boolean algebras have a lot more restrictions on their computability-theoretic properties: for example,

- the characterization of computable categoricity (to be discussed);
- the computable dimension of a Boolean algebra is either 1 or ω [Goncharov and Dzgoev 1980; Remmel 1981];
- every low_4 Boolean algebra has a computable copy [Knight and Stob 2000].

Motivation

As a rule of thumb, one can say the following:

(i) The class of graphs provides the richest computability-theoretic environment: every interesting example of a computable structure can be realized as a computable graph.

This property can be explicitly formalized — e.g., [Hirschfeldt, Khoussainov, Shore, and Slinko 2002].

(ii) Boolean algebras have a lot more restrictions on their computability-theoretic properties: for example,

- the characterization of computable categoricity (to be discussed);
- the computable dimension of a Boolean algebra is either 1 or ω [Goncharov and Dzgoev 1980; Remmel 1981];
- every low_4 Boolean algebra has a computable copy [Knight and Stob 2000].

Main Problem

Within the computability-theoretic framework, we consider the class HA of all Heyting algebras. Is the behavior of HA "closer" to graphs or to Boolean algebras?

In order to attack the problem, we work with *computable categoricity*.

Isomorphic structures, different algorithmic properties

From the point of view of the classical algebra, isomorphic copies of the same structure $\mathcal{S}$ have *the same* algebraic properties.

In computable structure theory, the situation is quite different.

(1) For the linear order $\mathcal{L} = (\mathbb{N}; \leq)$, its adjacency relation

$$\mathrm{Adj}(\mathcal{L}) = \{(x,y) : (x <_{\mathcal{L}} y) \text{ and } \neg\exists z(x <_{\mathcal{L}} z <_{\mathcal{L}} y)\}$$

is decidable.

On the other hand, one can build a computable copy $\mathcal{M} \cong \mathcal{L}$ with undecidable adjacency relation $\mathrm{Adj}(\mathcal{M})$.

Isomorphic structures, different algorithmic properties

From the point of view of the classical algebra, isomorphic copies of the same structure $\mathcal{S}$ have *the same* algebraic properties.

In computable structure theory, the situation is quite different.

(1) For the linear order $\mathcal{L} = (\mathbb{N}; \leq)$, its adjacency relation

$$\mathrm{Adj}(\mathcal{L}) = \{(x,y) : (x <_{\mathcal{L}} y) \text{ and } \neg\exists z(x <_{\mathcal{L}} z <_{\mathcal{L}} y)\}$$

is decidable.

On the other hand, one can build a computable copy $\mathcal{M} \cong \mathcal{L}$ with undecidable adjacency relation $\mathrm{Adj}(\mathcal{M})$.

(2) Let F be a computable field. The field F has a splitting algorithm if there is a computable procedure which given a polynomial $p(x) \in F[x]$, splits $p(x)$ into its irreducible factors in $F[x]$.

Theorem (Fröhlich and Shepherdson 1956)

There are two isomorphic computable fields F and G such that F has a splitting algorithm, but G has no splitting algorithm.

Computable categoricity

Definition (Mal'tsev)

A computable structure $\mathcal{S}$ is **computably categorical** (or autostable) if for any computable structure $\mathcal{A}$ isomorphic to $\mathcal{S}$, there is a computable isomorphism f from $\mathcal{A}$ onto $\mathcal{S}$ (i.e. f is an isomorphism, which is also a computable function).

Roughly speaking, all computable copies of a computably categorical structure $\mathcal{S}$ have the same algorithmic properties.

- The order $(\mathbb{N}; \leq)$ is not computably categorical.
- The field constructed by Fröhlich and Shepherdson is not computably categorical.
- Every finitely generated computable structure is computably categorical [Mal'tsev 1961].

Computable categoricity in familiar classes

Problem

Consider a familiar class of algebraic structures K. Provide a characterization of computably categorical members of K.

Computable categoricity in familiar classes

Problem

Consider a familiar class of algebraic structures K. Provide a characterization of computably categorical members of K.

Some of known characterizations:

- A computable Boolean algebra is computably categorical if and only if its set of atoms is finite [Goncharov and Dzgoev 1980; Remmel 1981].
- A computable linear order $\mathcal{L}$ is computably categorical if and only if the set $\mathrm{Adj}(\mathcal{L})$ is finite [Goncharov and Dzgoev 1980; Remmel 1981].
- An algebraically closed field is computably categorical if and only if it has finite transcendence degree over its prime subfield [Metakides and Nerode 1979].

Computable categoricity and Heyting algebras

Theorem (Downey, Kach, Lempp, Lewis-Pye, Montalbán, and Turetsky 2015)

The index set of computably categorical graphs is m-complete Π^1_1.

In other words, (in general) there is no simpler way to syntactically describe computable categoricity than the original definition given by Mal'tsev.

Computable categoricity and Heyting algebras

Theorem (Downey, Kach, Lempp, Lewis-Pye, Montalbán, and Turetsky 2015)

The index set of computably categorical graphs is m-complete Π^1_1.

In other words, (in general) there is no simpler way to syntactically describe computable categoricity than the original definition given by Mal'tsev.

How one can deal with Main Problem

Main Problem can be attacked as follows:

- If one can prove that Heyting algebras satisfy an analogue of the theorem above, then the class HA is "closer" to graphs.
- If one can obtain a nice algebraic characterization of computably categorical Heyting algebras, then HA is "closer" to Boolean algebras.

Completeness with respect to effective dimensions

Let $\mathcal{S}$ be a computable structure. For a Turing degree $\mathbf{d}$, the $\mathbf{d}$-*computable dimension* of $\mathcal{S}$, denoted by $\dim_{\mathbf{d}}(\mathcal{S})$, is the number of computable copies of $\mathcal{S}$, up to $\mathbf{d}$-computable isomorphisms.

For example, $\dim_{\mathbf{d}}(\mathcal{S}) = 2$ iff there are computable structures $\mathcal{A}$ and $\mathcal{B}$ such that:

- there is no $\mathbf{d}$-computable isomorphism from $\mathcal{A}$ onto $\mathcal{B}$, and
- every computable copy of $\mathcal{S}$ is either $\mathbf{d}$-computably isomorphic to $\mathcal{A}$, or $\mathbf{d}$-computably isomorphic to $\mathcal{B}$.

A class of structures K is **complete with respect to effective dimensions** if for any computable structure $\mathcal{S}$, there is a computable structure $\mathcal{A}_{\mathcal{S}} \in K$ such that for every Turing degree $\mathbf{d}$, we have:

$$\dim_{\mathbf{d}}(\mathcal{A}_{\mathcal{S}}) = \dim_{\mathbf{d}}(\mathcal{S}).$$

Main result

We treat *Heyting algebras with distinguished atoms and coatoms* as structures in the signature $L_{HA} \cup \{\mathrm{At}, \mathrm{Coat}\}$, where At and Coat are unary predicates.

Let HA_{AtCoat} be the class of all Heyting algebras with distinguished atoms and coatoms.

Theorem 1

The class HA_{AtCoat} is complete with respect to effective dimensions.

Proof Idea. We introduce a new encoding procedure, which given a computable undirected graph G, produces a computable structure $\mathcal{H}_{\mathcal{G}} \in HA_{\mathrm{AtCoat}}$. □

Main result

We treat *Heyting algebras with distinguished atoms and coatoms* as structures in the signature $L_{HA} \cup \{\mathrm{At}, \mathrm{Coat}\}$, where At and Coat are unary predicates.

Let HA_{AtCoat} be the class of all Heyting algebras with distinguished atoms and coatoms.

Theorem 1

The class HA_{AtCoat} is complete with respect to effective dimensions.

Proof Idea. We introduce a new encoding procedure, which given a computable undirected graph G, produces a computable structure $\mathcal{H}_{\mathcal{G}} \in HA_{\mathrm{AtCoat}}$. □

We note that the same encoding allows to show that the class HA_{AtCoat} is complete in the sense of [Hirschfeldt, Khoussainov, Shore, and Slinko 2002].

Main result

We treat *Heyting algebras with distinguished atoms and coatoms* as structures in the signature $L_{HA} \cup \{\mathrm{At}, \mathrm{Coat}\}$, where At and Coat are unary predicates.

Let HA_{AtCoat} be the class of all Heyting algebras with distinguished atoms and coatoms.

Theorem 1

The class HA_{AtCoat} is complete with respect to effective dimensions.

Proof Idea. We introduce a new encoding procedure, which given a computable undirected graph G, produces a computable structure $\mathcal{H}_{\mathcal{G}} \in HA_{\mathrm{AtCoat}}$. □

We note that the same encoding allows to show that the class HA_{AtCoat} is complete in the sense of [Hirschfeldt, Khoussainov, Shore, and Slinko 2002].

It is still open whether the class of Heyting algebras HA satisfies an analogue of Theorem 1.

As a consequence of the proof of Theorem 1, we obtain:

Corollary 1

The index set of computably categorical members of HA_{AtCoat} is m-complete Π^1_1.

This contrasts with the known result:

Theorem (Remmel 1981)

A computable Boolean algebra with distinguished atoms $(\mathcal{B}, \mathrm{At})$ is computably categorical if and only if $\mathcal{B}$ is isomorphic to a finite product of the following algebras:

- the countable atomless algebra,
- the algebra of finite and cofinite subsets of $\mathbb{N}$, and
- finite algebras.