Computability 0 (0) 1
IOS Press

Effective aspects of algorithmically random structures

Matthew Harrison-Trainor
Department of Pure Mathematics, University of Waterloo, Canada
maharris@uwaterloo.ca
Bakh Khoussainov
Department of Computer Science, University of Auckland, New Zealand
bmk@cs.auckland.ac.nz
Daniel Turetsky
Department of Mathematics, University of Notre Dame, IN, USA
dturetsk@nd.edu

Abstract. In this paper we study various properties of algorithmically random infinite structures. Our results address the following questions. How would one define algorithmic randomness for infinite structures? Could algorithmically random structures be computable? What are the similarities and differences between algorithmically random structures and algorithmically random infinite strings? What are the possible Turing degrees of algorithmically random structures? Are there algorithmically random infinite groups? For instance, we prove the following in this paper: (1) there are classes which contain algorithmically random yet computable structures, (2) there exist algorithmically random universal algebras with co-computably enumerable as well as computably enumerable word problems, (3) there are natural classes of structures in which the Turing degrees of algorithmically random structures can only be either computable or equivalent to the halting set, and (4) there are examples of algorithmically random groups. The first result shows a dramatic difference between algorithmically random strings and algorithmically random structures. The second result significantly improves the known theorem that algorithmically random structures computable in the halting set exist; these examples of algebras are sharp in terms of arithmetical hierarchy of the word problem for random algebras. The third result is a dichotomy theorem that characterises all possible Turing degrees of algorithmically random structures. Finally, the fourth result answers a nontrivial open question about the existence of algorithmically random groups.

1. Introduction

1.1. Background and motivation

Algorithmic randomness of infinite strings has a captivating history going back to Kolmogorov [9], Martin-Löf [17], Chaitin [2], Schnorr [21, 22], Levin [25]. In the last two decades the topic has attracted the attention of experts in complexity, computability, logic, philosophy, and algorithms. For instance, see the monographs by Downey/Hirschfeldt [3], Nies [18] and textbooks by Lee/Vitani [14], Calude [1]. Martin-Löf gave one of the foundational definitions of algorithmic randomness for infinite strings. The key ingredient in the definition is the natural measure present on the Cantor space $\{0,1\}^{\omega}$. Martin-Löf randomness states that random strings are those that avoid all effective measure zero sets.

Martin-Löf random (ML-random) infinite strings possess many properties that are intuitively clear (and that can formally be proved; see [3] [18]): they are incompressible; they do not contain infinite computable substrings; they satisfy the law of large numbers; no winning strategies exist that given an initial segment of the string bet on the next bit of it.

How would one define algorithmic randomness for infinite structures such as graphs, trees, groups, or universal algebras? What properties would these algorithmically random structures possess? Can algorithmically random structures be computable? Can algorithmically random universal algebras have computably enumerable word problem? What are the similarities and differences between algorithmically random structures and random infinite strings? Are there algorithmically random infinite groups? What are the possible Turing degrees of algorithmically random structures? In this paper we address and (partially) answer all these questions.

A natural yet naive way to introduce algorithmic randomness for structures is this. Let $\mathcal{G} = (\omega; R)$ be a relational structure, say in the language of undirected graphs. So, G is a graph. List all unordered pairs from ω. Now we code $\mathcal{G}$ into the following binary string $\alpha_{\mathcal{G}}$: $\alpha_{\mathcal{G}}(i) = 1$ if and only if R is true on the i^{th} unordered pair. Call the graph

$\mathcal{G}$ *algorithmically random* if the string $\alpha_{\mathcal{G}}$ is Martin-Löf random. Is this a good definition? One proves that if $\alpha_{\mathcal{G}}$ is ML-random then $\mathcal{G}$ is the Fraïssé limit of all finite graphs [10]. So, from model theory [8] we know that (1) the first order theory of graph $\mathcal{G}$ is $\aleph_0$-categorical, (2) the graph $\mathcal{G}$ is isomorphic to a computable graph, (3) the graph $\mathcal{G}$ satisfies the extension axioms, and (4) the theory of $\mathcal{G}$ admits effective quantifier elimination. Property (1) implies that all algorithmically random structures (as defined above) are isomorphic. Erdös and Spencer [4] remark that this phenomenon "demolishes the theory of infinite random graphs". As far as algorithmic randomness is concerned, property (2) is counter-intuitive in two ways. The first is that one would like algorithmic randomness to be a property of the isomorphism type rather than a property of the presentation of the structure. The second is that it is not clear at all why computability of a structure is compatible with algorithmic randomness; certainly this fails for infinite binary strings. Properties (3) and (4) provide an effective description of the structure in the first order logic and decidability of its the first order theory. All these defy the intuitive notion of algorithmic randomness and suggest alternative approaches should be taken towards defining algorithmic randomness for infinite structures.

As the natural measure in the Cantor space is central in defining algorithmic randomness for infinite strings, ultimately the task (of defining algorithmically random structures) consists of introducing meaningful measures into the classes of infinite algebraic structures. The second author accomplishes this task in papers [10] and [11] thus initiating a systematic study of algorithmic randomness for infinite algebraic structures. This paper continuous the line of research for studying algorithmic randomness for infinite structures proposed in [10] and [11]. In particular, the paper answers some of the important questions, including the ones we posed earlier, necessary for deep understanding of algorithmic random algebraic structures.

1.2. Outline and contributions

1. Algorithmic randomness for infinite structures is introduced through the concept of *branching classes* (*B-classes* for short) defined in [11]. Informally, a branching class provides a context in which one can reason about algorithmic randomness of algebraic structures. Section 2 recalls the definition and provides examples of branching classes. Every branching class $\mathcal{K}$ consists of finite structures and determines the class $\mathcal{K}_\omega$ of infinite structures that are direct limits of structures from $\mathcal{K}$. The class $\mathcal{K}$ also determines a finitely branching tree $T(\mathcal{K})$ such that there is a bijective operator $\eta \rightarrow \mathcal{A}_\eta$ mapping infinite paths η of $T(\mathcal{K})$ to structure A_η from $\mathcal{K}_\omega$. Using the tree $T(\mathcal{K})$, we naturally equip the class $\mathcal{K}_\omega$ with measure. Having the measure, one defines Martin-Löf random structures in $\mathcal{K}_\omega$. This implies, just like for infinite bit strings, that ML-random structures in the class $\mathcal{K}_\omega$ forms the continuum (Theorem 3.6), and among them there are ML-random structures computable in the halting set $\mathbf{0}'$ (Theorem 4.11).

2. It is an expected phenomenon that there exist ML-random structures computable in the halting set $\mathbf{0}'$. Such structures correspond to the leftmost paths of computable infinite finitely branching trees, and such paths are computable in $\mathbf{0}'$. The mapping $\eta \rightarrow \mathcal{A}_\eta$ mentioned above is a computable operator; hence, the structure $\mathcal{A}_\eta$ is computable in any oracle computing η. In contrast, building η from $\mathcal{A}_\eta$ requires the jump of the open diagram of $\mathcal{A}_\eta$. We exploit this in constructing ML-random and yet computable structures (Theorem 5.1). Such example is also built in [11], but our construction here is considerably simpler and shorter. This example is also essential for understanding results appearing later in the paper. We remarked earlier that it is counter-intuitive that a computable object might be algorithmically random; it is because ML-randomness does not preclude computability just like ML-randomness of strings does not preclude computability in the halting set $\mathbf{0}'$. But, we prove that no 2-ML-random structure is computable (Theorem 5.4), where 2-ML-randomness is ML-randomness relative to the halting set $\mathbf{0}'$.

3. We construct ML-random universal algebra $\mathcal{A}_1$ with computably enumerable word problem (Theorem 6.8). We also construct ML-random universal algebra $\mathcal{A}_2$ with co-computably enumerable word problem (Theorem 6.2). These examples are important because of the following reasons. The first is that no ML-random algebra has a computable word problem [10]. So, there are no computable ML-random algebras. Compare this with existence of ML-random computable structures mentioned above. However, ML-random algebras computable in the halting set $\mathbf{0}'$ exist by Theorem 4.11. The word problems in such algebras are Δ^0_2-sets. The examples $\mathcal{A}_1$ and $\mathcal{A}_2$ show that the word problem in ML-random algebras can be Σ^0_1 and Π^0_1 thus significantly improving the Δ^0_2 bound. The second reason is that these are the *only* possible imrpovements in terms of arithmetical hierarchy. Finally, drawing a parallel to infinite strings, it is known that there exist ML-random computably enumerable from the left (and also there exist

ML-random computably enumerable from the right) strings, e.g. the halting probability of the prefix free universal Turing machine [2]. The existence of structures $\mathcal{A}_1$ and $\mathcal{A}_2$ can be viewed as a reflection of this phenomenon in the case of universal algebras.

4. We study the degrees of ML-random structures. The degree of a structure is a fundamental concept used in modern model theory and computability [13] [20] [24]. Let $\mathcal{B}$ be an infinite structure with domain ω. The open diagram of $\mathcal{B}$ is the set of atomic relations or their negations true in $\mathcal{B}$. The *degree* of a structure $\mathcal{A}$, denoted by $deg(\mathcal{A})$, is the least Turing degree among all Turing degrees of open diagrams of structures isomorphic to $\mathcal{A}$ in case such degree exists. So, x is the degree of $\mathcal{A}$ if the following two properties are satisfied: (a) x computes a copy of the structure, and (b) all copies of $\mathcal{A}$ compute x. The degree represents the least amount of computability needed to represent the structure. We prove a dichotomy theorem showing that for natural branching classes $\mathcal{K}$, the only possible degrees of ML-random structures are the computable degree and the degree of the halting set, and both degrees are realisable (Theorem 7.5). This is an unexpected phenomenon and stands in a stark contrast with algorithmic randomness for infinite strings.

5. The study of random finitely generated groups has a long history [6, 23]. Traditionally, this is done by fixing generators and choosing for each n a relator of length n at random. By Gromov [6], random groups are hyperbolic (with probability 1), and hence have decidable word problem. The approach to randomness taken in this paper gives a different notion of randomness for groups, which corresponds to randomly determining larger and larger balls in Cayley graphs. Our approach contrasts Gromov's one in the sense that all ML-random groups have undecidable word problem. But, to exhibit ML-random groups we need to build branching classes of groups, the problem asked in [10]. The problem of finding branching classes of groups is hard because it intuitively relates to the word problem in groups. Using small cancellation theory developed in [15], we solve the problem and provide an example of a branching class of groups. So, there are ML-random groups (Theorem 8.6). The reason Gromov's random groups contrast with our random groups is that Gromov's definition is syntactic (select relators at random) while our definition is semantic and algebraic (select extensions of Cayley graphs at random).

2. Branching classes

In this section we review the branching classes of [10, 11]. A relational signature σ is $(R_1^{n_0}, \ldots, R_m^{n_m}, c_1, \ldots, c_k)$, where $R_i^{n_i}$ is a relational symbol of arity n_i and c_j is a constant symbol. We identify structures of the signature up to isomorphisms and call them σ-structures or structures if σ is clear from the content. All our structures are countable. Structures that contain functional operations can be identified with relational structures by replacing operations with their graphs.

Definition 2.1. An *embedded system* is a sequence $\{(\mathcal{A}_i, f_i)\}_{i\in\omega}$ such that each $\mathcal{A}_i$ is a finite structure, the cardinality of $\mathcal{A}_i$ is smaller than the cardinality of $\mathcal{A}_{i+1}$, and f_i is an embedding from $\mathcal{A}_i$ into $\mathcal{A}_{i+1}$. Call the sequence $\mathcal{A}_0, \mathcal{A}_1, \ldots$ *the base* of the system.

Each embedded system $\{(\mathcal{A}_i, f_i)\}_{i\in\omega}$ determines the *limit structure* denoted by $\lim_i(\mathcal{A}_i, f_i)$. The limit structure depends on embeddings f_i. For instance, each countable linear order is a direct limit of an embedded system with base $\{0 < \cdots < i\}_{i\in\omega}$. We want the limit to be independent of the embeddings, and formalise this through the following definition.

Definition 2.2. An embedded system $\{(\mathcal{A}_i, f_i)\}_{i\in\omega}$ is *strict* if $\lim_i(\mathcal{A}_i, f_i)$ is isomorphic to the direct limit of any embedded system with the base $\mathcal{A}_0, \mathcal{A}_1, \ldots$.

Let $\mathcal{K}$ be a *decidable* class of finite structures. We assume that the structures are given by their explicit representations. Our goal is to define classes of structures in which all embedded systems are strict.

Definition 2.3 (Height Function)**.** A computable function $h : \mathcal{K} \to \omega$ is called a *height function* for the class $\mathcal{K}$ if it satisfies the following properties:

(1) The set $h^{-1}(i)$ is finite for all $i \in \omega$, and we can compute the cardinality of $h^{-1}(i)$ for every i. A structure $\mathcal{A} \in \mathcal{K}$ *has height i* if $h(\mathcal{A}) = i$.

(2) For every $\mathcal{A} \in \mathcal{K}$ of height i there is a substructure $\mathcal{A}[i-1]$ of height $i-1$ such that all substructures of $\mathcal{A}$ of height $\leqslant i-1$ are contained in $\mathcal{A}[i-1]$.
(3) For all $\mathcal{A} \in \mathcal{K}$ of height i and all $\mathcal{B} \in \mathcal{K}$ with $\mathcal{A}[i-1] \subset \mathcal{B} \subseteq \mathcal{A}$ and $\mathcal{A}[i-1] = \mathcal{B}[i-1]$, we have $h(\mathcal{B}) = i$.

Note that h is an isomorphism invariant. Also, condition (2) implies that the substructure $\mathcal{A}[i-1]$ is unique, as it is the largest substructure of A of height $i-1$. In addition, for all $\mathcal{A} \in \mathcal{K}$ of height i and $j \leqslant i$ there exists a substructure $\mathcal{A}[j]$ of height j such that all substructures of $\mathcal{A}$ of height $\leqslant j$ are contained in $\mathcal{A}[j]$. Also, $\mathcal{A}[0] \subset \mathcal{A}[1] \subset \ldots \subset \mathcal{A}[i]$, where $\mathcal{A}[i] = \mathcal{A}$, and for $\ell \leqslant j$, $\mathcal{A}[j][\ell] = \mathcal{A}[\ell]$. Thus:

Corollary 2.4. *For all $\mathcal{A}, \mathcal{B} \in \mathcal{K}$, the structures $\mathcal{A}$ and $\mathcal{B}$ are isomorphic if and only if $h(\mathcal{A}) = h(\mathcal{B})$ and $\mathcal{A}[j] = \mathcal{B}[j]$ for all $j \leqslant h(\mathcal{A})$.*

The next lemma states the main property of classes $\mathcal{K}$ with height functions.

Lemma 2.5 ([11]). *Let $\mathcal{K}$ and h be as above. Every embedded system of structures from $\mathcal{K}$ is strict.*

Proof. Briefly, this is because of the uniqueness of the substructure $\mathcal{A}[i]$. Let $\{(\mathcal{A}_i, f_i)\}_{i\in\omega}$ and $\{(\mathcal{B}_i, g_i)\}_{i\in\omega}$ be embedded systems with the same base. For $j < i$, let $f_{j,i} : \mathcal{A}_j \to \mathcal{A}_i$ and $g_{j,i} : \mathcal{B}_j \to \mathcal{B}_i$ by the obvious compositions of maps. Let $\mathcal{A}$ and $\mathcal{B}$ be the direct limits of these two systems, respectively. We prove that $\mathcal{A}$ and $\mathcal{B}$ are isomorphic. If need be, by selecting subsequences, we can assume that the heights of $\mathcal{A}_i$ and $\mathcal{B}_i$ are at least i, and $\mathcal{A}_i[j] = \mathcal{A}_j[j]$ and $\mathcal{B}_i[j] = \mathcal{B}_j[j]$ for all $i \geqslant j$. This can be done since the number of structures of height j is finite and $\mathcal{A}_i$, $\mathcal{B}_i$ are increasing in size.

For all i, consider all isomorphisms $\alpha : \mathcal{A}_i[i] \to \mathcal{B}_i[i]$. Such an isomorphism necessarily exists, since $i \leqslant \ell = \min\{h(\mathcal{A}_i), h(\mathcal{B}_i)\}$, and $\mathcal{A}_i[i] = \mathcal{A}_i[\ell][i]$ and is unique. By the uniqueness of $\mathcal{A}_i[j]$, we have $\alpha(\mathcal{A}_i[j]) = \mathcal{B}_i[j]$ for all $j \leqslant i$. By uniqueness of $\mathcal{A}_i[j]$ and our assumption, $f_{j,i}(\mathcal{A}_j[j]) = \mathcal{A}_i[j]$. Similarly, $g_{j,i}(\mathcal{B}_j[j]) = \mathcal{B}_i[j]$. Thus the composition $g_{j,i}^{-1} \circ \alpha \circ f_{j,i}$ is an isomorphism from $\mathcal{A}_j[j]$ to $\mathcal{B}_j[j]$.

For $j \leqslant i$, $\alpha_i : \mathcal{A}_i[i] \to \mathcal{B}_i[i]$ and $\alpha_j : \mathcal{A}_j[j] \to \mathcal{B}_j[j]$, write $\alpha_j \trianglelefteq \alpha_i$ if $\alpha_i \circ f_{j,i} = g_{j,i} \circ \alpha_j$. As we have just argued, for every α_i there is an α_j with $\alpha_j \trianglelefteq \alpha_i$. The set of all such isomorphisms α thus forms a tree T under $\trianglelefteq$. Namely, nodes of this tree at level i are isomorphisms from $\mathcal{A}_i[i]$ onto $\mathcal{B}_i[i]$. This tree T is finitely branching, since each $\mathcal{A}_i[i]$ is finite. By König's lemma, there is an infinite path $(\alpha_0, \alpha_1, \ldots)$ on the tree. By definition of $\trianglelefteq$, this path induces a well-defined map from $\mathcal{A}$ to $\mathcal{B}$.

It remains to show that α is total and surjective. For $\mathcal{A}_j$ of height i, there is an embedding $f_{j,i}$ of $\mathcal{A}_j$ into $\mathcal{A}_i[i]$, and so $f_{j,i}(\mathcal{A}_j) \subseteq \mathcal{A}_i[i]$. Thus $\mathcal{A}_j \subseteq \text{dom } \alpha_i \circ f_{j,i}$. So α is total. A symmetric argument shows that it is surjective. □

Thus, for every embedded system $(\mathcal{A}_i, f_i)_{i\in\omega}$ from $\mathcal{K}$ there is an embedded system $\{\mathcal{B}_i, g_i\}_{i\in\omega}$ of structures from the same class such that: (1) the direct limits $\lim_i(\mathcal{A}_i, f_i)$ and $\lim_i(\mathcal{B}_i, g_i)$ are isomorphic; (2) the height of each $\mathcal{B}_i$ is i; (3) the embeddings g_i are identity embeddings; and (4) for all $i \leqslant j$ we have $\mathcal{B}_j[i] = \mathcal{B}_i[i] = \mathcal{B}_i$. First, we can assume that the $\mathcal{A}_0 \subseteq \mathcal{A}_1 \subseteq \mathcal{A}_2 \subseteq \cdots$, and then we can simply take $\mathcal{B}_i = \mathcal{A}_j[i]$ for sufficiently large j.

Definition 2.6. Let $\mathcal{K}$ and h be as above. Set:

$$\mathcal{K}_\omega = \{\mathcal{A} \mid \mathcal{A} \text{ is the direct limit of an embedded system from the class } \mathcal{K}\}.$$

Here is now our definition (from [11]) that provides the context for defining algorithmically random structures:

Definition 2.7. The class $\mathcal{K}$ together with the height function h is called a *branching class*, or *B-class for short*, if for all $\mathcal{A} \in \mathcal{K}$ of height i there exist distinct structures $\mathcal{B}, \mathcal{C} \in \mathcal{K}$ such that $h(\mathcal{B}) = h(\mathcal{C}) > h(\mathcal{A})$ and $\mathcal{B}[i] = \mathcal{C}[i] = \mathcal{A}$. If a class $\mathcal{K}$ is a *B*-class, then we refer to the class $\mathcal{K}_\omega$ also as a *B-class*.

We use the following examples of B-classes. The first two examples are from [10]. Many more examples are in [11].

Example 1 (Rooted trees). Consider the class $Tree_d$ of all finite rooted trees T such that every node of T has not more than d immediate successors. The height $h(T)$ of T is the length of the longest path from the root to the leaves of T. The gives a B-class.

Example 2 (c-generated algebras). An *algebra* $\mathcal{A}$ is a tuple $(A; f_1, \dots, f_n, c_1, \dots, c_m)$, where $A \neq \emptyset$ is the *domain* of $\mathcal{A}$, each $f_i : A^{k_i} \to A$ is a *total* function called an *atomic operation* of arity k_i, and each c_j is a a *constant*) of $\mathcal{A}$. The algebra $\mathcal{A}$ is *c-generated* if every element a of $\mathcal{A}$ is the interpretation of some ground term t. Call the term t a *representation* of a.

The *height* $h(a)$ of the element $a \in A$ is the minimal height among the heights of all the ground terms representing a. The *height* $h(\mathcal{A})$ of the algebra $\mathcal{A}$ is the supremum of all the heights of its elements.

For a c-generated algebra $\mathcal{A}$ and $n \in \omega$, we define: $A(n) = \{a \in A \mid h(a) \leqslant n\}$. Each k_i-ary operation f_i of $\mathcal{A}$ defines the *partial operation* $f_{i,n}$ on $A(n)$: $f_{i,n}(a_1, \dots, a_{k_i}) = f_i(a_1, \dots, a_{k_i})$ if $h(a_i) < n$ for $i = 1, \dots, k_i$, and else undefined.

Call the partial algebras of the form $\mathcal{A}(n)$ *proper partial algebras*. The term "proper" refers to the fact in $\mathcal{A}(n)$ we do not decide the operations on elements of height n, even if the result is also of height n (this is vital to make the class branching). Define the height of $\mathcal{A}(n)$ to be n. Every infinite c-generated algebra $\mathcal{A}$ is the direct limit of the sequence $\{\mathcal{A}(n)\}_{n\in\omega}$. Define:

$$PAlg = \{\mathcal{B} \mid \ \mathcal{B} \text{ is a proper partial algebra } \}.$$

The class $PAlg$ is a B-class. The class $PAlg_\omega$ consists of all c-generated infinite algebras.

Example 3 (Binary rooted ordered trees). Let $OT(2)$ be the class of all binary rooted trees where the children of internal node are linearly ordered. Since the trees are binary, every internal node has either left-child or right-child. These trees have signature $\sigma = (L, R, c)$, where c is the root, $L(x, y)$ indicates that y is the left child of x, and $R(x, y)$ that y is the right child of x. The class $OT(2)$ is a B-class.

3. Topology and measure

In this section we will review the definition of a random structure from [10, 11]. Let $\mathcal{K}$ be a branching class, $r_{\mathcal{K}}(n)$ be the number of structures in $\mathcal{K}$ of height n. We define the tree $\mathcal{T}(\mathcal{K})$.

The root of $\mathcal{T}(\mathcal{K})$ is the empty set. This is level -1. The nodes of $\mathcal{T}(\mathcal{K})$ at level $n \geqslant 0$ are all structures from $\mathcal{K}$ of height n. There are exactly $r_{\mathcal{K}}(n)$ of them. Let $\mathcal{B}$ be a structure from $\mathcal{K}$ of height n. Its successors in $\mathcal{T}(\mathcal{K})$ are the structures $\mathcal{C}$ of height $n+1$ such that $\mathcal{B} = \mathcal{C}[n]$.

To generate a random structure, we randomly choose a height zero structure, and then a height one structure extending that, a height two structure extending that, and so on, to obtain the structure which is the union of all of them. This corresponds essentially to picking a random path through the tree $\mathcal{T}(\mathcal{K})$. We need to show that this tree is effective, following which we will put a probability measure on paths through the tree, similar to the way in which one puts a measure on 2^ω when considering binary strings.

The proof of the next lemma is easy:

Lemma 3.1 (Computable Tree Lemma). *For $\mathcal{T}(\mathcal{K})$ we have the following:*

(1) *Given a node x of $\mathcal{T}(\mathcal{K})$, we can effectively compute the structure $\mathcal{B}_x$ associated with x. We identify the nodes x and the structures $\mathcal{B}_x$.*
(2) *For each node x in $\mathcal{T}(\mathcal{K})$, the structure $\mathcal{B}_x$ has an immediate successor. Moreover, we can compute the number of immediate successors of x.*
(3) *For each path $\eta = \mathcal{B}_0, \mathcal{B}_1, \dots$ in $\mathcal{T}(\mathcal{K})$ we have: $\mathcal{B}_0 \subset \mathcal{B}_1 \subset \dots$. Thus, the union of this chain determines the structure $\mathcal{B}_\eta = \cup_i \mathcal{B}_i \in \mathcal{K}_\omega$.*
(4) *The mapping $\eta \to \mathcal{B}_\eta$ is a bijection between all infinite paths of $\mathcal{T}(\mathcal{K})$ and the class $\mathcal{K}_\omega$.*

For $\mathcal{A} \in \mathcal{K}_\omega$, let $\mathcal{A}[i]$ be the largest substructure of $\mathcal{A}$ of height i. This is well-defined. The structure $\mathcal{A}[i]$ contains all substructures of $\mathcal{A}$ of height $\leqslant i$.

Using the tree $\mathcal{T}(\mathcal{K})$ we introduce topology in $\mathcal{K}_\omega$:

Definition 3.2 (**Topology**). For $\mathcal{B} \in \mathcal{K}$, set:

$$Cone(\mathcal{B}) = \{\mathcal{A} \mid \mathcal{A} \in K_\omega, \text{ and } \mathcal{A}[n] = \mathcal{B}\ \}.$$

These are *base open sets*. Call $\mathcal{B}$ the *base of the cone*.

We now define the following measures on K_ω:

Definition 3.3 (**Measure**). For the root r, define $\mu(Cone(\mathcal{B}_r)) = 1$. Let $\mathcal{B}_x \in \mathcal{K}$ be of height n. Let e_x be the number of immediate successors of $\mathcal{B}_x$ in the tree $\mathcal{T}(\mathcal{K})$. For any immediate successor y of x, set $\mu(Cone(\mathcal{B}_y)) = \mu(Cone(\mathcal{B}_x))/e_x$.

Definition 3.4 (**Metric**). For structures $\mathcal{A}, \mathcal{C} \in \mathcal{K}_\omega$, let n be the maximal level at which $\mathcal{A}[n]$ and $\mathcal{C}[n]$ coincide. Let $\mathcal{B}$ be the node of the tree such that $\mathcal{A}[n] = \mathcal{B}$. The distance $d_\mu(\mathcal{A}, \mathcal{C})$ between $\mathcal{A}$ and $\mathcal{C}$ is then: $d_\mu(\mathcal{A}, \mathcal{C}) = \mu(Cone(\mathcal{B}))$.

The space $\mathcal{K}_\omega$ has the following properties: (1) The function d is a metric on $\mathcal{K}_\omega$; (2) $\mathcal{K}_\omega$ is compact; (3) Finite unions of cones form clopen sets in the topology; (4) All μ-measurable sets is a σ-algebra.

Now we define ML-random structures in $\mathcal{K}_\omega$ using definitions from algorithmic randomness. A class $C \subseteq \mathcal{K}_\omega$ is a Σ^0_1-*class* if there is a computably enumerable (c.e.) sequence $\mathcal{B}_0, \mathcal{B}_1, \ldots$ of structures from $\mathcal{K}$ such that $C = \cup_{i \geqslant 1} Cone(\mathcal{B}_i)$.

Definition 3.5. Let $\mathcal{K}$ be a B-class. Consider the class $\mathcal{K}_\omega$ of infinite structures.

(1) A *Martin-Löf (ML) test* is a uniformly c.e. sequence $\{G_n\}_{n\in\omega}$ of Σ^0_1-classes with $G_{n+1} \subset G_n$ and $\mu(G_n) < 2^{-n}$ for all n.
(2) A structure $\mathcal{A}$ from $\mathcal{K}_\omega$ *fails* a ML-test $\{G_n\}_{n\in\omega}$ if $\mathcal{A}$ belongs to $\cap_n G_n$. Otherwise, we say that the structure $\mathcal{A}$ *passes* the test.
(3) A structure $\mathcal{A}$ from $\mathcal{K}$ is *ML-random* if it passes every Martin-Löf test.

If $C \subset \mathcal{K}_\omega$ is contained in a ML-test, then we say that C has *effective measure* 0.

It is standard to show that there is a *universal* ML-test in the sense that passing that test is equivalent to passing all ML-tests [18]. Formally, a ML-test $\{U_n\}_{n\in\omega}$ is *universal* if for any ML-test $\{G_m\}_{m\in\omega}$ it is the case that $\cap_m G_m \subseteq \cap_n U_n$. A construction of a universal ML-test is easy: list all ML-tests $\{G^e_k\}_{k\in\omega}$ uniformly on e and k, and set $U_n = \cup_e G^e_{n+e+1}$. The resulting sequence $\{U_n\}_{n\in\omega}$ is a universal ML-test. Hence, to prove that a structure $\mathcal{A} \in \mathcal{K}_\omega$ is ML-random it suffices to show that $\mathcal{A}$ passes the universal ML-test $\{U_n\}_{n\in\omega}$. The class of non-ML-random structures has effective measure 0. Thus, we have:

Theorem 3.6 (Theorem 6.7 of [11]). *Let $\mathcal{K}$ be a B-class. The number of ML-random structures in $\mathcal{K}_\omega$ is continuum.*

4. Randomness and the halting set

We study elementary effective aspects of ML-random structures from $\mathcal{K}_\omega$. We start with the following definition that goes back to Malcev [16] and Rabin [19].

Definition 4.1. An infinite structure $\mathcal{A}$ is *computable* if it is isomorphic to a structure with domain ω such that all atomic operations and relations of the structure are computable.

Computability is an isomorphism property. A structure is computable iff it is isomorphic to a structure that has a computable open diagram. A stronger definition that involves the height function and fits the setting of this paper is this:

Definition 4.2. A computable structure $\mathcal{A}$ from $\mathcal{K}_\omega$ is *strictly computable* if the size of the substructure $\mathcal{A}[i]$ can be effectively computed for all $i \in \omega$.

Strict computability implies non ML-randomness:

Theorem 4.3 (Lemma 6.11 of [11])**.** *If $\mathcal{A}$ is strictly computable then $\mathcal{A}$ is not ML-random.*

Proof. There exists an effective procedure that given n computes the structure $\mathcal{A}[n]$. Hence, for an appropriately chosen sequence $(n_i)_{i\in\omega}$, the sequence of cones $Cone(\mathcal{A}[n_0])$, $Cone(\mathcal{A}[n_1])$, $Cone(\mathcal{A}[n_2])$, ... forms a Martin-Löf test that the structure $\mathcal{A}$ fails. □

We derive several corollaries. The first corollary is the following logical property of ML-random structures. For $\mathcal{A} \in \mathcal{K}_\omega$ consider the binary predicate L:

$$L(x,y) \text{ iff } x \in \mathcal{A}[i] \iff y \in \mathcal{A}[i] \text{ for all } i \geqslant 0.$$

Definition 4.4. Call L the *level predicate*. Extend $\mathcal{A}$ with L thus defining the extended structure $(\mathcal{A}, L)$.

The extension $(\mathcal{A}, L)$ of $\mathcal{A}$ is a natural extension in the sense that the automorphism group of $\mathcal{A}$ coincides with the automorphism group of $(\mathcal{A}, L)$.

Corollary 4.5 (Corollary 6.12 of [11])**.** *If $(\mathcal{A}, L)$ is a computable structure and the $\exists$-theory of $(\mathcal{A}, L)$, that is the set*

$$\{\phi \mid (\mathcal{A}, L) \models \phi \text{ and } \phi \text{ is an existential first-order sentence}\},$$

is decidable, then $\mathcal{A}$ is not ML-random.

Proof. We can determine the elements of $\mathcal{K}$ of height i, and for each such finite structure $\mathcal{B}$ we can construct an existential sentence stating that $\mathcal{B}$ is a substructure of $\mathcal{A}$. Thus we can determine the substructures of $\mathcal{A}$ of height i. The largest is $\mathcal{A}[i]$, allowing us to effectively compute $\mathcal{A}[i]$ for all i. □

The second property concerns c-generated algebras:

Corollary 4.6. *No computable c-generated ML-random algebra exists.*

Proof. If $\mathcal{A}$ is c-generated, then one computes $\mathcal{A}[i]$ for any given i. Hence, $\mathcal{A}$ is strictly computable. □

In particular, we have the following property for classical algebraic structures such as groups and rings:

Corollary 4.7. *No finitely generated computable group or ring exists that is ML-random.*

Proof. Say G is a finitely generated group with generators a and b. We can view G as a universal algebra with generators a and b (from the signature). Hence, since G is computable, for each i we can compute the domain $G[i]$. Hence, G is strictly computable. □

The next property concerns connected graphs:

Definition 4.8. A connected computable locally finite graph $G = (\omega, E)$ is *highly computable* if for each $v \in G$ we can effectively compute all neighbours (that is, the set $\{u \mid \{v,u\} \in E\}$) of v.

Corollary 4.9. *No connected highly computable graph G is ML-random.*

Proof. For each fixed vertex g in the graph G, the naturally defined height function

$$f(v) = \text{the shortest distance from } v \text{ to } g$$

is computable since G is highly computable. In any branching class $\mathcal{K}$ of graphs (as formally defined in [11]), the height function for the class $\mathcal{K}$ is compatible with the height function f defined above. Now note that for each i we can compute $G[i]$ (that is all vertices at distance $\leqslant i$ from g). Therefore, G is strictly computable, and hence G is not ML-random. □

Note that the hypotheses of the theorem and all the corollaries above depend only on the structures chosen, and not on the branching classes $\mathcal{K}_\omega$. Namely, if $\mathcal{A}$ is strictly computable and $\mathcal{A} \in \mathcal{K}_\omega$ (no matter what the branching class $\mathcal{K}$ is) then $\mathcal{A}$ is not ML-random in the class K_ω. Thus, non-randomness of strictly computable structures is context independent.

Definition 4.10. A structure $\mathcal{A}$ is $\mathbf{0}'$*-computable* if it is isomorphic to a structure with domain ω such that all atomic relations, including equality, and operations of $\mathcal{A}$ are computable in $\mathbf{0}'$.

Each computable structure is $\mathbf{0}'$-computable. The reverse is false: there are finitely presented groups with undecidable word problem. The next theorem shows that Theorem 4.3 can't be extended to $\mathbf{0}'$-computable structures.

Theorem 4.11 (Theorem 6.14 of [11])**.** *Every branching class $\mathcal{K}_\omega$ contains a $\mathbf{0}'$-computable ML-random structure.*

Proof. Consider the mapping $\eta \to \mathcal{A}_\eta$ that maps all paths of the tree onto the structures K_ω. The mapping $\eta \to \mathcal{A}_\eta$ is such that the structure $\mathcal{A}_\eta$ is a computable operator. Hence $\mathcal{A}_\eta$ is computable in any oracle that computes η. It is standard to show that there exists an ML-random path η in the tree such that η is computable in the halting set $\mathbf{0}'$ [3] [18]. Hence the structure $\mathcal{A}_\eta$ is computable in $\mathbf{0}'$. □

5. Randomness and computability

For any structure $\mathcal{A}$ from a branching class $\mathcal{K}_\omega$ there is a path $\eta \in T(\mathcal{K})$ such that the structure $\mathcal{A}_\eta$ (determined by η) is isomorphic to $\mathcal{A}$. As we noted the path η can be constructed in the jump of the open diagram of $\mathcal{A}$. In particular, if $\mathcal{A}$ is computable then η is computable in the halting set $\mathbf{0}'$. We exploit this observation to prove the following theorem.

Theorem 5.1 (Theorem 7.1 of [11])**.** *There exists a branching class $\mathcal{S}_\omega$ that has a computable ML-random structure.*

The proof we give here is better than that from [11]; in particular, it highlights certain combinatorial properties of the branching class which we will motivate results later in the paper.

Proof. Consider the full binary tree $T = \{0,1\}^\star$ (whose elements are finite binary strings). The signature of this tree consists of the root symbol c and two binary predicates L and R such that $L(x,y)$ if and only if $y = x0$ and $R(x,y)$ if and only if $y = x1$.

Let $\preceq$ be the lexicographic order. For each $x \in \{0,1\}^\star$ consider the structure $\mathcal{B}_x$ whose domain consists of all strings y such that $|y| \leqslant |x|$ and $y \preceq x$. Clearly $\mathcal{B}_x$ is a tree in the signature (c, L, R).

We set $\mathcal{S} = \{\mathcal{B}_x \mid x \in \{0,1\}^\star\}$. The height function of the class $\mathcal{S}$ associates with every structure $\mathcal{B}_x$ the length $|x|$ of x. Obviously, $\mathcal{S}$ is a branching class; also, $T(\mathcal{S})$ can naturally be identified with the binary tree $T = \{0,1\}^\star$. We have the following easy lemma:

Lemma 5.2 (Algebraic left-embedding lemma)**.** *Suppose that $x \preceq y$. Then:*

(1) *If $|x| \leqslant |y|$ then $\mathcal{B}_x$ is embedded into $\mathcal{B}_y$.*

(2) *If* $|x| > |y|$ *then* $\mathcal{B}_x$ *is embedded into* $\mathcal{B}_{yz}$ *for all* z *such that* $|x| \leqslant |yz|$ *and* $x \preceq yz$.

Let $\{U_n\}_{n\in\omega}$ be the universal ML-test. Consider the set $\{0,1\}^\omega \setminus U_1$. All infinite paths in this set are paths through a computable binary tree. Let η be the leftmost infinite path of the tree. The path is a ML-random path. So, the structure $\mathcal{B}_\eta$ is ML-random. The path is effectively approximable from the left.

Let $x_0 \preceq x_1 \preceq x_2 \preceq \ldots$ be computable approximation to η such that $|x_i| < |x_{i+1}|$. By the lemma above we have a computable sequence of embedded structures: $\mathcal{B}_{x_0} \subset \mathcal{B}_{x_1} \subset \ldots$. The direct limit of this structure is isomorphic to $\mathcal{B}_\eta$. Thus we have built a ML-random structure which is computable. □

In the introduction, we remarked that a ML-random string cannot be computable, and yet by the theorem ML-random structures can be computable. The issue is that an ML-random structure is not sufficiently random to preclude being computable just as an ML-random string is not sufficiently random to preclude being computable in the halting set. Nevertheless, below we show that no computable structure is 2-ML-random.

Definition 5.3. Consider a B-class $\mathcal{K}_\omega$. One defines what it means for a structure $\mathcal{A}$ from the class $\mathcal{K}_\omega$ to be *2-ML-random* by replacing the Σ^0_1-classes in the definition of ML-randomness by Σ^0_1-classes relative to the halting set $\mathbf{0}'$.

Clearly, 2-ML-randomness implies ML-randomness. The theorem below shows that 2-ML-randomness precludes computable structures to be 2-ML-random (just like ML-randomness for strings precludes ML-random strings to be computable).

Theorem 5.4. *If structure* $\mathcal{A}$ *is computable then* $\mathcal{A}$ *is not 2-ML-random.*

Proof. The proof is just recasting of the proof of Theorem 4.3 with an oracle for $\mathbf{0}'$. There is a $\mathbf{0}'$-computable function that computes, for each n, the finite structure $\mathcal{A}[n]$. Then we can use the same argument as in Theorem 4.3: for an appropriately chosen sequence $(n_i)_{i\in\omega}$, the sequence of cones $Cone(\mathcal{A}[n_0]), Cone(\mathcal{A}[n_1]), \ldots$ forms a Martin-Löf test relative to $\mathbf{0}'$ that the structure $\mathcal{A}$ fails. □

6. Co-c.e. and c.e. ML-random algebras

Our interest is in ML-random c-generated algebras as explained in Example 3. From Corollary 4.6 no finitely generated computable algebra is ML-random. But we also know, from Theorem 4.11, that the class contains ML-random structures computable in the halting set. It is not too hard to see that the word problem in such an ML-random algebra is always a Δ^0_2-set. Thus, a natural question is whether it is possible to sharpen these results. We sharpen these results by showing that there exist computably enumerable as well as co-computably enumerable ML-random algebras. Clearly, the word problems in such algebras are Σ^0_1 and Π^0_1-sets.

Formally, the word problem is defined as follows. Let $\mathcal{A}$ be a c-generated algebra. Elements of this algebra are represented by ground terms. *The word problem* for algebra $\mathcal{A}$ refers to the set

$$\{(t,p) \mid t \text{ and } p \text{ are ground terms and } \mathcal{A} \models t = p\}.$$

Definition 6.1. Let $\mathcal{A}$ be a c-generated algebra.

(1) Call $\mathcal{A}$ *computably enumerable* if the word problem for $\mathcal{A}$ is computably enumerable.
(2) Call $\mathcal{A}$ *co-computably enumerable* if the word problem for $\mathcal{A}$ is co-computably enumerable.

There are a plethora of examples of c.e. and co-c.e. algebras. For instance, all finitely presented groups (in fact, all finitely presented algebras) are c.e. algebras. An example of a co-c.e. algebra is any group generated by finitely many computable permutations of ω. Every c.e. or co-c.e. algebra is computable in the halting set $\mathbf{0}'$. So, the next theorem strengthens Theorem 4.11 for the class of algebras.

Theorem 6.2. *There is a branching class of finitely generated algebras that contains a co-computably enumerable ML-random structure.*

Proof. Consider the B-class $\mathcal{K}$ of all finite ordered trees T as where each non-leaf node has at most two children. The children of the node are ordered from left to right. Order the nodes of the tree T on the same level (from left to right) in the natural way. We call to this order the *level order* of the nodes. The tree T is a structure in the signature (c, L, R).

We now transform any tree T from $\mathcal{K}$ into a *partial-tree algebra* $\mathcal{A}(T)$ as follows. The signature of $\mathcal{A}(T)$ has unary operation L and R, and the generator, or constant, c (root). The domain of $\mathcal{A}(T)$ is T. Operations L and R are defined as follows. For a leaf x, $L(x)$ and $R(x)$ are undefined. For a non-leaf node x, if x has a left child y and right child z then $L(x) = y$ and $R(x) = z$. If x has one child only, say y, then $L(x) = R(x) = y$. This transformed the tree T to the partial algebra $\mathcal{A}(T)$. We identify T with $\mathcal{A}(T)$.

To define the desired class S, consider the concatenation operations on partial tree–algebras T_1 and T_2. The *concatenation* of T_1 and T_2, written $T_1 \cdot T_2$, is obtained by attaching to every leaf of T_1 the tree T_2 and by identifying the leaf with the root of T_2. Formally, the domain of $T_1 \cdot T_2$ is the set $T_1 \setminus \{x \mid x \text{ is a leaf of } T_1\} \cup \{(v, x) \mid x \text{ is a leaf of } T_1 \text{ and } v \in T_2\}$. The L and R operations on T_1 are inherited from the original L and R. For elements (v, x), we have $L(v, x) = (v', x)$ if $L(v) = v'$ and $R(v, x) = v''$ if $R(v) = v''$.

We need some notation: denote the full binary partial-tree algebra of height k by T_k; the tree L_k denotes the partial tree-algebra isomorphic to $(\{a_0, \ldots, a_k\}; L, R)$ where $L(a_i) = R(a_i) = a_{i+1}$ for $i < k$, and the root is a_0; finally, $A_{k,i}$ denotes the partial tree-algebra $L_{k-i} \cdot T_i$, where $i \leqslant k$.

Lemma 6.3. *The partial tree-algebras $A_{k,0}$, $A_{k,1}$, ..., $A_{k,k}$ have the same height. In addition, each partial algebra $A_{k,i}$ is a homomorphic image of the partial algebra $A_{k,i+1}$, and the homomorphism is unique.*

We define, by induction, our branching class $\mathcal{S}$ by directly constructing the tree $T(\mathcal{S})$ that represents structures from $\mathcal{S}$. At stage n we define partial-tree algebras at level n of the tree $T(\mathcal{S})$. We also define an ordering on the elements of level n.

Stage 0. The root of $T(\mathcal{S})$ is the partial-tree algebra L_0 (and so, the root is also T_0).

Stage $n + 1$. Suppose we constructed partial tree-algebras $\mathcal{B}_1$, ..., $\mathcal{B}_{2^n}$ at level n of the tree $T(\mathcal{S})$, listed from left to right. At level n the tree $T(\mathcal{S})$ has exactly 2^{n+1} nodes. Namely, each node (partial tree-algebra) $\mathcal{B}_i$ has exactly two immediate successors: $\mathcal{B}_i \cdot A_{2^n,i-1}$ and $\mathcal{B}_i \cdot A_{2^n,i}$. We order $\mathcal{B}_i \cdot A_{2^n,i-1}$ to the left of $\mathcal{B}_i \cdot A_{2^n,i}$. For $j < i$, we order the successors of $\mathcal{B}_j$ to the left of the successors of $\mathcal{B}_i$.

Thus, the class $\mathcal{S}$ consists of all partial tree-algebras T that appear at some stage n of the construction above. All the partial tree-algebras at level n of the tree $T(\mathcal{S})$ are defined to have height n. We provide several lemmas about the partial tree-algebras of the class $\mathcal{S}$. The next lemma is obvious:

Lemma 6.4. *The class $\mathcal{S}$ is a B-class. Moreover, the tree $T(\mathcal{S})$ is isomorphic to the infinite binary tree.*

From this lemma above we immediately get:

Lemma 6.5. *For all $\mathcal{B} \in \mathcal{S}$, $\mu(Cone(\mathcal{B})) = 2^{-h(\mathcal{B})}$.*

The next lemma proves an algebraic property of tree-algebras of height n.

Lemma 6.6. *If $\mathcal{B}_1, \ldots, \mathcal{B}_{2^n}$ are the partial tree-algebras at level n of $T(\mathcal{S})$ listed from left to right, then each $\mathcal{B}_i$ is a homomorphic image of $\mathcal{B}_{i+1}$. Further, each homomorphism is unique.*

Proof. The proof is by induction on n. When $n = 0$ the statement is clear. Assume that we have proved the lemma for all trees $\mathcal{B}_1, \ldots, \mathcal{B}_{2^n}$ that correspond to nodes at level n of the tree $T(\mathcal{S})$. Consider $\mathcal{B}_i$. This has exactly two immediate successors: $\mathcal{B}_i \cdot A_{2^n,i-1}$ and $\mathcal{B}_i \cdot A_{2^n,i}$. By Lemma 6.3 there is a unique homomorphism from $A_{2^n,i}$ onto $A_{2^n,i-1}$. Hence, there exists a homomorphism from $\mathcal{B}_i \cdot A_{2^n,i}$ onto $\mathcal{B}_i \cdot A_{2^n,i-1}$. It is clearly unique.

Consider $\mathcal{B}_{i-1} \cdot A_{2^n,i-1}$, which is the element at level $n+1$ immediately to the left of $\mathcal{B}_i \cdot A_{2^n,i-1}$. There is a unique homomorphism h_i, by inductive hypothesis, from $\mathcal{B}_i$ onto $\mathcal{B}_{i-1}$. This implies that there exists a unique homomorphism from $\mathcal{B}_i \cdot A_{2^n,i-1}$ onto $\mathcal{B}_{i-1} \cdot A_{2^n,i-1}$. This proves that all the partial tree-algebras at level n of $T(\mathcal{S})$ form a chain of homomorphically embedded structures. □

Thus there is a natural bijection $x \to \mathcal{A}_x$ from the set of all binary strings onto the class $\mathcal{S}$. We identify the tree $T(\mathcal{S})$ with the full binary tree.

Corollary 6.7. *If $x \preceq y$ & $|x| = |y|$, then $\mathcal{A}_y$ is homomorphically mapped onto $\mathcal{A}_x$.*

Let $\{U_n\}_{n\in\omega}$ be the universal ML-test in the class $\mathcal{S}_\omega$ (which we identify with the Cantor space $\{0,1\}^\omega$). Consider the set $\{0,1\}^\omega \setminus U_1$. Since $\mu(U_1) < 1/2$, all infinite paths in $\{0,1\}^\omega \setminus U_1$ can be considered as paths through a computable binary tree. Let η be the leftmost infinite path of the tree.

Let $x_0 \preceq x_1 \preceq x_2 \preceq \ldots$ be computable approximation to η such that $|x_i| < |x_{i+1}|$. We have the sequence of partial tree-algebras: $\mathcal{A}_{x_0}, \mathcal{A}_{x_1}, \mathcal{A}_{x_2}, \ldots$. The direct limit of this structure is isomorphic to $\mathcal{B}_\eta$. This structure is ML-random. Our goal is to show that $\mathcal{B}_\eta$ is a co-c.e. algebra.

Consider all terms of the signature $< L, R, c >$ built from the constant c and operation symbols L and R: c, $L(c)$, $R(c)$, $LL(c)$, $LR(c)$, $RL(c)$, ... The set of all terms forms an algebra; in this algebra for terms t the values of L and of R on t are terms $L(t)$ and $R(t)$. This is known as the term algebra $Term$. The algebra $\mathcal{B}_\eta$ is a homomorphic image of the term algebra $Term$. Let $g : Term \to B_\eta$ be the onto homomorphism, and let $E = \{(t,q) \mid g(t) = g(q)\}$ be the kernel of the homomorphism. The quotient algebra $Term/E$ is isomorphic to $\mathcal{B}_\eta$. It suffices to explain why E is a co-c.e. relation.

To show that E is co-c.e. consider the partial algebra $\mathcal{A}_{x_{n+1}}$ in the sequence above. We can assume that we have an equivalence relation E_n on the partial algebra $T_{k(n)}$ such that the factor structure $T_{k(n)}/E_n$ is isomorphic to $\mathcal{A}_{x_n}$. At stage $n+1$ either $\mathcal{A}_{x_{n+1}}$ extends A_{x_n} or A_{x_n} is a subtructure of a homomorphic image of $\mathcal{A}_{x_{n+1}}$. In the second case some E_n-equivalent elements become non-E_{n+1}-equivalent. These elements can effectively be computed. If two elements are not E_n-equivalent then they will never be E_m-equivalent for all $m \geqslant n$. So, non-equality in the algebra $Term/E$ is co-c.e. □

Theorem 6.8. *There exists a branching class of finitely generated algebras that contains a computably enumerable ML-random structure.*

Proof. The proof of Theorem 6.2 can be changed slightly to construct the desired algebra. Simply replace η with the rightmost infinite path, say $1 - \eta$, which instead of being computable from the left below, is computable from the right. Consider an effective sequence of strings $\ldots \preceq y_3 \preceq y_2 \preceq y_1 \preceq y_0$ that converges to $1 - \eta$ from the right. The resulting structure $\mathcal{B}_{1-\eta}$ is a desired algebra. This is because, by Corollary 6.7, whenever we move from $\mathcal{A}_{y_n}$ to $\mathcal{A}_{y_{n+1}}$, we pass to a homomorphic image and then extend. □

7. Degrees of Random Structures

In this section we study Turing degrees of algorithmically random structures in branching classes $\mathcal{K}_\omega$.

Definition 7.1. For a structure $\mathcal{A}$, the *degree of* $\mathcal{A}$ is the least Turing degree computing an isomorphic copy of $\mathcal{A}$, if such degree exists.

Degrees of structures play important role in understanding effective aspects of structures and their interactions with model theory [13] [20] [24]. Here we show that the degrees of ML-random structures depend on the underlying branching classes $\mathcal{K}_\omega$. Here is one natural example of branching classes:

Definition 7.2. A branching class $\mathcal{K}$ is *jumpless* if for every path η in $T(\mathcal{K})$, every isomorphic copy of $\mathcal{A}_\eta$ computes η.

For instance, the class of c-generated algebras is jumpless. There are many other examples of jumpless classes among graphs and trees. The following proposition is easy to prove:

Proposition 7.3. *If the class $\mathcal{K}_\omega$ is jumpless, then every structure in $\mathcal{K}_\omega$ has a degree, and the degrees of ML-random structures in the class are precisely the Turing degrees which contain random binary strings.*

Motivated by the algebraic left-embedding lemma (Lemma 5.2), we give the next definition:

Definition 7.4. A B-class $\mathcal{K}$ is *left-algebraic* if there is a computable ordering on the elements of each level of $T(\mathcal{K})$ such that for the induced lexicographic ordering $\preceq$ the following holds:

(1) For every path η through $T(\mathcal{K})$, for every sequence $x_0 \preceq x_1 \preceq x_2 \preceq \dots$ with limit η, the sequence computes an isomorphic copy of $\mathcal{A}_\eta$.
(2) For every path η through $T(\mathcal{K})$, for every isomorphic copy of $\mathcal{A}_\eta$, the copy computes a sequence $x_0 \preceq x_1 \preceq x_2 \preceq \dots$ with limit η.

An example of a left-algebraic branching class is the class $\mathcal{S}$ constructed in the proof of Theorem 5.1.

Theorem 7.5. *If a B-class $\mathcal{K}$ is left-algebraic, then the class $\mathcal{K}_\omega$ contains ML-random structures of degree $\mathbf{0}$ and $\mathbf{0}'$, but of no other degree.*

Proof. Let $(U_n)_{n\in\omega}$ be the universal ML-test for $\mathcal{K}$. Then $\mu(U_1) < 1/2$, and the elements of $\mathcal{K}_\omega - U_1$ are the paths through a computable tree. Let η be the leftmost path. Then there is a computable sequence $x_0 \preceq x_1 \preceq \dots$ with limit η. Since $\mathcal{K}$ is left-algebraic, there is a computable copy of $\mathcal{A}_\eta$. Hence, the class $\mathcal{K}_\omega$ has an ML-random structure of degree 0.

The following fact is analogous to one for random infinite binary strings, and the proof is similar:

Lemma 7.6. *If $A_\eta \in \mathcal{K}_\omega$ is ML-random and there is a computable sequence $x_0 \preceq x_1 \preceq x_2 \preceq \dots$ with limit η, or a computable sequence $x_0 \succeq x_1 \succeq x_2 \succeq \dots$ with limit η, then η has degree $\mathbf{0}'$.*

The path η in this lemma should not be confused with the sequence $x_0 \preceq x_1 \preceq x_2 \preceq \dots$; we do not, in general, expect them to have the same Turing degree.

Now consider the computable tree for which the paths are elements of $\mathcal{K}_\omega - U_1$, and now let η be the rightmost path. There exists a computable sequence $x_0 \succeq x_1 \succeq x_2 \succeq \dots$ converging to η. Since $\mathcal{K}$ is left-algebraic, every isomorphic copy of $\mathcal{A}_\eta$ computes a sequence $\hat{\eta}_0 \preceq \hat{\eta}_1 \preceq \hat{\eta}_2 \preceq \dots$ with limit η. So every isomorphic copy of $\mathcal{A}_\eta$ computes η as follows: for an input i, search for an ℓ such that $\eta_\ell[j] = \hat{\eta}_\ell[j]$ for all $j \leqslant i$. Such an ℓ must exist. Then $\eta_\ell[i] = \eta[i]$. Since η has degree $\mathbf{0}'$, this shows that $\mathcal{A}_\eta$ has degree $\mathbf{0}'$. So, $\mathcal{K}_\omega$ contains a random structure of degree $\mathbf{0}'$.

Thus, the class $\mathcal{K}_\omega$ has ML-random structures of degrees 0 and $\mathbf{0}'$. To show that no other random degrees are possible, we need a deeper analysis provided by the next lemmas.

Lemma 7.7. *If $\mathcal{A}_\eta \in \mathcal{K}_\omega$ and $\mathcal{A}_\eta$ has a degree, then there is a sequence $\eta_0 \preceq \eta_1 \preceq \eta_2 \preceq \dots$ with limit η and a computably enumerable set $F \subset \omega \times \mathcal{K} \times \mathcal{K}$ such that $\eta_n = x$ iff there is a $y \preceq \eta$ with $(n, x, y) \in F$.*

Proof. Consider an isomorphic copy $\mathcal{B}$ of $\mathcal{A}_\eta$ realising the degree of the structure $\mathcal{A}_\eta$. Since $\mathcal{K}$ is left-algebraic, this copy computes a sequence $(\eta_0 \preceq \eta_1 \preceq \eta_2 \preceq \dots)$ with limit η. From now on we fix this sequence. As $\mathcal{B}$ realisez the least degree, every other copy of $\mathcal{A}_\eta$ computes this fixed sequence $(\eta_0 \preceq \eta_1 \preceq \eta_2 \preceq \dots)$. Also, every other sequence $(\rho_0 \preceq \rho_1 \preceq \rho_2 \preceq \dots)$ with limit η computes a copy of $\mathcal{A}_\eta$. Hence, the sequence $(\rho_0 \preceq \rho_1 \preceq \rho_2 \preceq \dots)$ also computes the specified sequence $(\eta_0 \preceq \eta_1 \preceq \eta_2 \preceq \dots)$.

The idea is now the following. We will attempt to build a sequence $(\rho_0 \preceq \rho_1 \preceq \rho_2 \preceq \dots)$ with limit η which does not compute $(\eta_0 \preceq \eta_1 \preceq \eta_2 \preceq \dots)$. Any such attempt must fail, and we use this failure to construct the desired c.e. set F.

Let $(\Phi_i)_{i\in\omega}$ be an effective listing of all Turing functionals, and let $(\sigma_i)_{i\in\omega}$ be a listing of the elements of $T(\mathcal{K})$ such that $\sigma_i \preceq \eta$ for all i. We construct $(\rho_0 \preceq \rho_1 \preceq \rho_2 \preceq \dots)$ in stages, although we do not claim that our construction is effective.

Stage 0. We begin by letting $\rho_0 = \sigma_0$.

Stage $2i+1$. Suppose we defined ρ_0 through $\rho_{k_{2i}}$. At this stage, we must ensure that Φ_i does not compute $(\eta_0 \preceq \eta_1 \preceq \eta_2 \preceq \dots)$. To ensure this, we search for an n and a finite sequence $(\tau_1 \preceq \tau_2 \preceq \dots \preceq \tau_m)$ such that $\rho_{k_{2i}} \preceq \tau_1$, $\tau_m \preceq \eta$ and

$$\Phi_i^{(\rho_0 \preceq \dots \preceq \rho_{k_{2i}} \preceq \tau_1 \preceq \dots \preceq \tau_m)}(n) \text{ is defined}$$
$$\text{and this value is not equal to } \eta_n.$$

Define $\rho_{k_{2i}+j} = \tau_j$ for $1 \leqslant j \leqslant m$. Otherwise, we make no new definitions at this stage.

Stage $2i+2$. Suppose we have defined ρ_0 through $\rho_{k_{2i+1}}$. At this stage, we must ensure that our sequence surpasses σ_i without exceeding η. We define $\rho_{k_{2i+1}+1} = \sigma_i$ if $\rho_{k_{2i+1}} \preceq \sigma_i$. Otherwise, we make no new definitions at this stage.

Note that even numbered stages guarantee that the limit of $(\rho_0 \preceq \rho_1 \preceq \rho_2 \preceq \dots)$ is η. As we argued earlier, this sequence computes the specified sequence $(\eta_0 \preceq \eta_1 \preceq \eta_2 \preceq \dots)$. Therefore there must exists the least i such that

$$\Phi_i^{(\rho_0 \preceq \rho_1 \preceq \dots)}(n) = \eta_n \text{ for all } n.$$

Consider now the way the construction acts at stage $2i+1$. There must be no sequence $(\tau_1 \preceq \dots \preceq \tau_m)$ as desired. Now we define F. A triple (n, x, y) belongs to F iff there is a sequence $(\tau_1 \preceq \dots \preceq \tau_m)$ with $\Phi_i^{(\rho_0 \preceq \dots \preceq \rho_{k_{2i}} \preceq \tau_1 \preceq \dots \preceq \tau_m)}(n) = x$ and $\tau_m = y$. Observe that F is computably enumerable. We now argue that F has the desired properties.

Suppose $(n, x, y) \in F$ and $y \preceq \eta$. Then there is some sequence $(\tau_1 \preceq \dots \preceq \tau_m)$ which witnessed that (n, x, y) belongs in F. If $\eta_n \neq x$, then this sequence is as desired for Stage $2i+1$, contrary to our choice of i. So $\eta_n = x$.

Conversely, suppose $\eta_n = x$. There is an m with $\Phi_i^{(\rho_0 \preceq \dots \preceq \rho_{k_{2i}+m})}(n) = x$. Define the sequence $(\tau_1 \preceq \dots \preceq \tau_m)$ by $\tau_j = \rho_{k_{2i}+j}$, where $j = 1, \dots, m$. This sequence witnesses that $(n, x, \rho_{k_{2i}+m}) \in F$, and $\rho_{k_{2i}+m} \preceq \eta$. □

Lemma 7.8. *If $\mathcal{A}_\eta \in \mathcal{K}_\omega$ has a degree, then either there is a computable sequence $(\rho_0 \succeq \rho_1 \succeq \dots)$ with limit η, or there is $(\eta_0 \preceq \eta_1 \preceq \dots)$ a computable sequence with limit η.*

Proof. Consider the set F and the sequence $(\eta_0 \preceq \eta_1 \preceq \dots)$ as in the previous lemma and its proof. We attempt to effectively construct $(\rho_0 \succeq \rho_1 \succeq \dots)$ as stated in the lemma. If this fails, we will be able to argue that $(\eta_0 \preceq \eta_1 \preceq \dots)$ is a computable sequence. We proceed in stages, building ρ_n at stage n.

Stage n. We enumerate F until we see two triples (m, x_0, y_0) and (m, x_1, y_1) with $x_0 \neq x_1$ and $y_0 \preceq y_1$. If $n > 0$, we also require $\rho_{n-1} \succeq y_1$. If we find such a pair of triples, we know that $\eta \prec y_1$. Otherwise, we would have $y_0 \preceq y_1 \preceq \eta$, and so by the nature of F, we would have $x_0 = \eta_m = x_1$. We set $\rho_n = y_1$ for the first pair of triples we find.

Now, suppose that for every $z \succ \eta$, there is a pair of triples (m, x_0, y_0) and (m, x_1, y_1) as above with $z \succeq y_1$. Then our construction will always find such a pair, and we will eventually define a $\rho_n \preceq z$ for every such z. So we will have $(\rho_0 \succeq \rho_1 \succeq \dots)$ with limit η.

Suppose instead there is a $z \succ \eta$ such that there is no pair of desired triples with $z \succeq y_1$. We know that for every m there is a triple (m, x_0, y_0) with $y_0 \preceq \eta \prec z$ and $x_0 = \eta_m$. So it must be that there is no triple (m, x_1, y_1) with $y_1 \preceq z$ and $x_1 \neq \eta_m$. Then we can compute η_n by enumerating F until we see a triple (n, x, y) with $y \preceq z$. When we see such a triple, we know that $\eta_n = x$. □

To finish the proof of the theorem, suppose $\mathcal{A}_\eta \in \mathcal{K}_\omega$ has a degree and is random. Then there is either a computable sequence $(\eta_0 \preceq \eta_1 \preceq \dots)$ or a computable sequence $(\eta_0 \succeq \eta_1 \succeq \dots)$ with limit η. As we earlier argued, in the first case $\mathcal{A}_\eta$ has a computable copy, and in the second case $\mathcal{A}_\eta$ has degree $\mathbf{0}'$. □

We dont know degrees realised by random structures from neither jumpless nor left-algebraic B-classes.

8. A Branching Class of Groups

In this section, we build a branching class of groups. We want to define a class $\mathcal{K}$ of finite partial groups on a fixed set of generators, in the style of Example 3, so that $\mathcal{K}_\omega$ consists of groups. There are two difficulties. First, $\mathcal{K}$ cannot include any finite group, as that would violate the branching condition; nor can it include any finite partial group which only has one extension to an infinite group. Second, there is the problem of effectiveness. The class $\mathcal{K}$ must be decidable but because the word problem for groups is undecidable, we cannot decide whether a finite partial atomic diagram can be extended to a group.

Our construction uses ideas from [7], namely the application of small cancellation to code structure into a finitely generated groups. We begin by recalling the basic definitions of small cancellation. For a reference on small cancellation, see Chapter 5 of [15].

A presentation $\langle S \mid R\rangle$ is *symmetrized* if every relator in R is cyclically reduced and the relator set R is closed under inverses and cyclic permutation. For the symmetrized presentation $\langle S \mid R\rangle$, a word $u \in F(S)$ is a *piece* if there are two distinct $r_1, r_2 \in R$ such that u is an initial subword of both r_1 and r_2.

Definition 8.1. The presentation $\langle S \mid R\rangle$ has the $C'(\lambda)$ *small cancellation hypothesis* if for each $r \in R$ and every piece u with $r = uv$, we have $|u| < \lambda|r|$. We also say that a presentation *satisfies the small cancellation hypothesis* if it does after we replace the relators set with its symmetrized closure.

The key lemma on small cancellation groups is Greendlinger's Lemma. It says that in a small cancellation group, every presentation of the trivial word must share a long subword in common with a relator.

Lemma 8.2 (Greendlinger's Lemma)**.** *Assume that $G = \langle S \mid R\rangle$ is a $C'(\lambda)$ small cancellation group with $0 \leqslant \lambda \leqslant 1/6$. Let w be a non-trivial freely reduced word representing e of G. Then there is a subword v of w and a defining relator r such that v is also a subword of r and such that $|v| > (1-3\lambda)|r|$.*

We are now ready to define the branching class of groups. Given $x \in \{0,1\}^\omega$, define the group G_x generated by $a = a_0$, b, and c. The group G_x is such that it is generated with $\{a_i\}_{i\in\omega} \cup \{b, c\}$ and relations

- $u(a_i, b) = a_{i+1}$, and
- $u(a_i, c) = e$ for each i with $x(i) = 1$.

where $u(s,t) = sts^2ts^3t\cdots s^{100}t$. The group G_x is a $C'(1/10)$ small cancellation group.

Given a group G generated by $\{a_0, b, c\}$, let $G[i]$ be the finite partial group consisting of the elements represented by words of length at most i (in a_0, b, c). It is important to note that we think of G_x as being generated by $\{a_i\}_{i\in\omega} \cup \{b, c\}$ when we think of it as a small cancellation group, but as being generated by $\{a_0, b, c\}$ when we consider the lengths of elements. Our branching class is the class of finite partial groups which arise from the groups G_x, where $x \in 2^\omega$.

Definition 8.3. Let $\mathcal{K}$ be the class of finite partial groups $G_x[i]$ where $x \in 2^\omega$ and $i \in \omega$. The height function on $\mathcal{K}$ is $G_x[i] \mapsto i$.

The next two lemmas show that there is a connection between $G_x[i]$ and an initial segment of x, with the length of this initial segment depending on i. These lemmas will show that $\mathcal{K}$ is a B-class.

Lemma 8.4. *If $x\restriction_\ell = y\restriction_\ell$, then $G_x[1958^\ell/4] = G_y[1958^\ell/4]$. Moreover, from $x\restriction_\ell$, we can effectively determine the atomic diagram of $G_x[1958^\ell/4]$.*

Using this Lemma, it makes sense to define $G_x[i]$ whenever $x \in 2^{<\omega}$ and $i \leqslant 1958^{|x|}/4$.

Proof. Given a freely reduced word w in $\{a_i\}_{i\in\omega} \cup \{b, c\}$, for each $i \in \omega$, let n_i be the number of occurrences of a_i or a_i^{-1} in w. Define the weight $\mu(w)$ to be

$$\mu(w) = n_0 + 1958n_1 + 1958^2 n_2 + \cdots .$$

We will often say "the number of a_i's" when we mean "the number of occurrences of a_i's and a_i^{-1}'s".

Suppose w_1 and w_2 are words in a_0, b, c, of length at most $1958^\ell/4$, such that $w_1 = w_2$ in G_x. We claim that $w_1 = w_2$ in G_y. From this and a similar argument reversing the roles of x and y, it will follow that $G_x[1958^\ell/4] = G_y[1958^\ell/4]$.

Let $w = w_1 w_2^{-1}$. Note that $\mu(w) < 1958^\ell$. We will argue that for each non-trivial word v with $\mu(v) < 1958^\ell$, with v equal to the identity in G_x, there is a word v' of shorter length than v, with v equivalent to v' in both G_x and G_y. From this it will follow that $w = e$ in G_y, and so $w_1 = w_2$ in G_y.

Given v as above, since $v = e$ in G_x, by Greendlinger's Lemma there is a generating relator $r = u_1 u_2$ of G_x, with u_1 a subword of v and $|u_1| > (1 - \frac{3}{10})|r| = \frac{7}{10}|r|$. We consider a number of cases:

- r is an inverse or cyclic permutation of $u(a_i, b) = a_{i+1}$. Then r is a relator in both G_x and G_y. Note that $|r| = 5151$, so $|u_2| \leqslant \frac{3}{10}|r| < 1546$. Thus u_2 can have at most 1546 a_i's and one a_{i+1}. Also, u_1 has at least 3504 a_i's (as $u(a_i, b) = a_{i+1}$ has 5050 a_i's in it). Thus replacing u_1 by u_2^{-1} in v to obtain v', we get that the number of a_i's in v is at least 1958 more than the number of a_i's in v', and the number of a_{i+1}'s is at most one more in v' than in v. Thus $\mu(v') \leqslant \mu(v)$. Since r was a relator in both G_x and G_y, v and v' are equivalent in both G_x and G_y.
- r is an inverse or cyclic permutation of $u(a_i, c) = e$. Then u_1, and hence v, has at least 3504 a_i's. Since

  $$1958^i(\text{number of } a_i\text{'s in } v) \leqslant \mu(v) < 1958^\ell$$

  we have that $i < \ell$. Since $x\restriction_\ell = y\restriction_\ell$, r is also a relator in G_y. Replacing u_1 in v by u_2^{-1} to get v', we have that $\mu(v') \leqslant \mu(v)$ as v_2 has at most 1546 a_i's.

Notice that this procedure also gives a method for determining whether $w_1 = w_2$ in G_x using only $x\restriction_\ell$. □

Lemma 8.5. *From $G_x[5150^\ell]$, we can effectively determine $x\restriction_\ell$.*

Proof. We claim that $x(i) = 1$ iff $u(a_i, c) = e$ in G_x. If $x(i) = 1$, then $u(a_i, c) = e$ is a defining relator. Assume that $u(a_i, c) = e$ in G_x. We will show that $x(i) = 1$. Since $u(a_i, c)$ is a reduced word equivalent to e in G_x, there is an inverse or cyclic permutation $r = u_1 u_2$ of a defining relator of G_x such that u_1 is a subword of $u(a_i, c)$, and $|u_1| \geqslant \frac{7}{10}|r|$. Then u_1 must contain at least two instances of a_i and at least one instance of c, and so the only option is that $u(a_i, c) = e$ is a defining relator of G_x and r is an inverse or cyclic permutation of it. This implies that $x(i) = 1$.

Finally, it suffices to see that for $i < \ell$, $u(a_i, c)$ is a word of length at most 5150^ℓ in a_0, b, c, in G_x. We argue by induction that a_i is equivalent to a word of length at most 5150^i in G_x. For a_0, this is true trivially. Suppose that we know that a_i is equivalent to a word of length at most 5150^i. Then $a_{i+1} = u(a_i, b)$ is equivalent to a word of length at most 5150^{i+1}. This completes our inductive argument. Then if $i < \ell$, a_i is equivalent to a word of length at most 5150^i, and so $u(a_i, c)$ is a word of length at most $5150^{i+1} \leqslant 5150^\ell$. □

The two lemmas above show that h is a computable height function. Moreover, we can compute $h^{-1}(i)$ for every i. The other properties of the height function are clear. We show t $\mathcal{K}$ is branching. Given a finite partial group $G_x[i]$, choose y such that $y \upharpoonright_i = x \upharpoonright_i$, but $y(i) \neq x(i)$. Then by Lemma 8.4, $G_x[i] = G_y[i]$, but by Lemma 8.5, $G_x[5150^{i+1}] \neq G_y[5150^{i+1}]$. Thus we have:

Theorem 8.6. *The class $\mathcal{K}$ is a branching class. Moreover, each $\mathcal{A} \in \mathcal{K}_\omega$ is a group.*

Thus we have the following corollary.

Corollary 8.7. *The branching class $\mathcal{K}_\omega$ contains countinuumly many ML-random groups. Some of these ML-random groups are computable in $\mathbf{0}'$.*

References

[1] C. S. Calude. Information and Randomness - An Algorithmic Perspective. 2nd Edition, Revised and Extended, Springer-Verlag, Berlin, 2002.

[2] G. J. Chaitin. On the length of programs for computing finite binary sequences: statistical considerations, Journal of the ACM, 16, 145-159, 1969.

[3] R. Downey and D. Hirschfeldt. Algorithmic Randomness and Complexity, Theory and Applications of Computability, Springer, New York.

[4] P. Erdös and J. Spencer. Probabilistic methods in combinatorics. Probability and Mathematical Statistics, No. 17. Academic Press, New York-London, 1974

[5] G. Grätzer. Universal Algebra. Revised reprint of the second edition, Springer, New York, 2008.

[6] M. Gromov. Random walk in random groups, Geom. Funct. Anal. 13, no.1, p.73-146, 2003.

[7] M. Harrison-Trainor and M.-C. Ho. On optimal Scott sentences of finitely generated structures, to appear in Proceedings of the American Mathematical Society.

[8] W. Hodges. Model Theory. Encyclopaedia of Mathematics and Its Applications, No 42. Cambridge University Press, 1993.

[9] A. Kolmogorov. Three approaches to the quantitative definition of information, Problems of Information Transmission 1, p.1-7, 1965.

[10] B. Khoussainov. A quest for algorithmically random infinite structures. Proceedings of LICS-CSL 2014 conference. Vienna, Austria.

[11] B. Khoussainov. A quest for algorithmically random infinite structures, II. Proceedings of LFCS 2015.

[12] B. Khoussainov. Randomness, Computability, and Algebraic Specifications, Annals of Pure and Applied Logic 91, no.1, p.1-15, 1998.

[13] J. F. Knight, Degrees coded in jumps of orderings, Journal of Symbolic Logic 51, 1034-1042, 1986.

[14] M. Li and P. Vitanyi. An introduction to Kolmogorov complexity and its applications, Springer-Verlag, 3rd Edition 2008 (xx+792 pp).

[15] R. C. Lyndon and P. E. Schupp. Combinatorial group theory. Springer-Verlag, Berlin, 1977.

[16] A. I. Mal'cev. Constructive algebras. I. Uspehi Mat. Nauk, 16(3 (99)):3-60, 1961.

[17] P. Martin-Löf. The definition of random sequences, Information and Control 9(6), 602-619, 1966.

[18] A. Nies. Computability and Randomness, Oxford University Press, 2009.

[19] M. O. Rabin. Computable algebra, general theory and theory of computable fields. Trans. Amer. Math. Soc., 95:341-360, 1960.

[20] L. J. Richter. Degrees of structures. J. Symbolic Logic, 46(4):723-731, 1981.

[21] C. P. Schnorr. A unified approach to the definition of random sequences, Mathematical Systems Theory 5(3):246-258 (1971).

[22] C. P. Schnorr. The process complexity and effective random tests, Proceedings of the fourth ACM symposium of Theory of Computing, Denver, Colorado, May 1-3, 1972.

[23] L. Silberman. Addendum to: "Random walk in random groups", Geom. Funct. Anal. 13, no.1, p.147-177, 2003.

[24] T. A. Slaman, Relative to any nonrecursive set, Proceedings of the American Mathematical Society 126, 2117-2122, 1998.

[25] A. K. Zvonkin and L. A. Levin. The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms. Russian Math. Surveys, 25(6):83-124, 1970.