

On Elementary Computability-Theoretic Properties of Algorithmic Randomness

Asat Arslanov¹

*School of Computer Science and Software Engineering
Monash University
Victoria 3800
Australia*

Abstract

In this paper we apply some elementary computability-theoretic notions to algorithmic complexity theory with the aim of understanding the role and extent of computability techniques for algorithmic complexity theory. We study some computability-theoretic properties of two different notions of randomness for finite strings: randomness based on the blank-endmarker complexity measure and Chaitin's randomness based on the self-delimiting complexity measure. We introduce the notion of complex infinite sequence of finite strings, which we call *K-bounded* sequences.

1 Introduction

In this paper we apply some elementary computability-theoretic notions to algorithmic randomness theory with the aim of understanding the role and extent of computability techniques for algorithmic randomness theory. Two standard textbooks in the area of algorithmic randomness are Calude [3] and Li-Vitanyi [11]. Our notation is standard, following that used by Chaitin [5] and Soare [15]. In particular, $\omega = \{0, 1, \dots\}$ is the set of all non-negative integer numbers and $\{W_e\}_{e \in \omega}$ is a standard enumeration of all computably enumerable (c. e.) sets, and $\{\varphi_e\}_{e \in \omega}$ is a Gödel numbering of all partial computable functions. Let $\{0, 1\}^*$ be the set of binary strings (also called programs), and let $\{0, 1\}^n$ be the set of binary strings of length n . We will use the letters $\alpha, \beta, \gamma, \delta$ to denote finite strings. We let $|\alpha|$ denote the length of α and λ denote the empty string.

A *tt-condition* is a pair $\langle \{x_1, x_2, \dots, x_n\}, \eta \rangle$, where x_1, x_2, \dots, x_n are natural numbers and η is an n -ary Boolean function, $n \geq 1$. We assume an

¹ Email: asat@cs.monash.edu.au

effective enumeration of all tt -conditions and we will denote the tt -condition with index k by tt_k . Let $B \subseteq \omega$. We say that B *satisfies* the tt -condition tt_k and write $B \models_{tt} tt_k$, if $\eta((B(x_1), \dots, B(x_n))) = 1$. If there exists a computable function f such that $x \in A \Leftrightarrow B \models_{tt} tt_{f(x)}$, for all x , then we say that A is *tt -reducible to B* and write $A \leq_{tt} B$. A set A is *tt -complete* if A is c. e. and every c. e. set is tt -reducible to A . In general, for any Turing reducibility R , a set A is *R -complete* if A is c. e. and every c. e. set is R -reducible to A .

We shall work with Turing machines operating on strings. The *absolute program-size complexity* induced by a Turing machine φ (sometimes called *blank-endmarker computer*) is defined by $K_\varphi(\alpha) = \min\{|\beta| : \beta \in \{0, 1\}^*, \varphi(\beta) = \alpha\}$. A *Chaitin computer* is a Turing machine (operating on strings) which has a *prefix-free domain* (see Calude [3]). For a Chaitin computer C one associates the *absolute self-delimiting program-size complexity*, or *Chaitin complexity*, $H_C(\alpha) = \min\{|\beta| : \beta \in \{0, 1\}^*, C(\beta) = \alpha\}$. The *Invariance Theorem* states the existence of a Turing machine ψ (Chaitin computer U) such that for every Turing machine φ (Chaitin computer C) there exists a constant *const* such that $K_\psi(\alpha) \leq K_\varphi(\alpha) + \text{const}$ ($H_U(\alpha) \leq H_C(\alpha) + \text{const}$) for all strings α . For this paper we fix a universal Chaitin computer U and denote by H the induced program-size complexity. Also, we fix a universal blank-endmarker computer ψ and denote by K its induced program-size complexity. The notion of randomness for finite strings was defined in an attempt to capture the idea that a string is random if it cannot be algorithmically compressed. The original approach (by Chaitin and Kolmogorov) to define the algorithmic randomness for finite strings was by using the notion of blank-endmarker program-size complexity K .

Definition 1.1 [Chaitin [3], Kolmogorov [3]] A string $\alpha \in \{0, 1\}^*$ is **Kolmogorov t -random** if $K(\alpha) \geq |\alpha| - t$; α is **K -random** if it is Kolmogorov 0-random.

Definition 1.2 [Chaitin [3]] A string $\alpha \in \{0, 1\}^*$ is **Chaitin t -random** if $H(\alpha) \geq \max_{\beta \in \{0, 1\}^{|\alpha|}} H(\beta) - t$; α is **C -random** if it is Chaitin 0-random.

We will denote by $RAND^K$ and $RAND^C$ the sets of Kolmogorov and Chaitin random strings, respectively. For more details on algorithmic randomness we refer the reader to Calude [3].

Our aim is to study the computability-theoretic properties of $RAND^K$ and $RAND^C$ in an attempt to estimate the computational difference between these two sets. It is known that both $RAND^K$ and $RAND^C$ are effectively immune sets, and Turing equivalent to the halting problem. Below we note that the positions of $RAND^K$ and $RAND^C$ are at the *same* level in the scale of immunity notions since they are not hyperimmune sets. This concept of a hyperimmune set turned out to have very interesting characterizations which were later shown to have important applications in many areas of computability and complexity theory. The characterization of hyperimmunity due to

Medvedev and Uspensky (e. g. see [12]) states that for a hyperimmune set A there is no computable function f such that, for each n , $a_n \leq f(n)$, where a_n is the number coding the n -th string of A in an increasing order. Thus from the computability-theoretic point of view both $RAND^K$ and $RAND^C$, being non-hyperimmune, are not “meagre”. Below we can see how these and some other recent results on $RAND^K$ and $RAND^C$ can be easily derived from the few classical facts in the literature. We would like to remark here that, nevertheless, we found some interesting differences between these two notions of randomness in terms of other computability-theoretic hierarchies (see [1]). In the following section on hyperimmunity and K -bounded sequences we will look at a special kind of hyperimmune sets and obtain results which justify the introduction of the concept of *complex* infinite sequences of finite strings. We call them K -bounded sequences.

2 Hypersimple sets and program-size complexity

In this preliminary section we can see how some scattered results on $RAND^K$ and $RAND^C$ can be easily derived from classical facts in the literature. Chaitin in his abstract on the information-theoretic aspects of Post’s construction of a simple set (see Chaitin [6], p. 288) defines, for any integer $n \geq 0$, the following sets $P(n)$ and $Q(n)$ of finite binary strings:

$\alpha \in P(n)$ if and only if there is a program β with $|\alpha| > n + |\beta|$ and α is the first string computed by β , i.e. $\psi(\beta) = \alpha$.

$\alpha \in Q(n)$ if and only if $n + K(\alpha) < |\alpha|$.

Theorem 2.1 (Chaitin [6]) *There is a constant c such that for all n , $P(n+c)$ is contained in $Q(n)$, and $Q(n)$ is contained in $P(n)$.*

The set $P(n)$ is a version of Post’s original construction of a simple set and, in particular, $P(n)$ is an effectively simple, non-hypersimple set. Consequently, all $Q(n)$, $n \geq 0$, are effectively simple and non-hypersimple sets. Notice that $Q(0) = \{\alpha : \alpha \in \{0,1\}^*, K(\alpha) < |\alpha|\} = \overline{RAND^K}$. Therefore we have the following results.

Corollary 2.2 *The set of K -random strings is an effectively immune, non-hyperimmune co-c. e. set.*

Naturally, it is easy to see that the set $RAND^C$ is non-hyperimmune. Indeed, let $\{F_n\}_{n \in \omega}$ be the following strong array: $F_n = \{\alpha : |\alpha| = n\}$, for every n . Obviously, the sequence $\{F_n\}_{n \in \omega}$ is a computable sequence of pairwise disjoint finite sets. For every n , $F_n \cap RAND^C \neq \emptyset$, and the set $RAND^C$ is non-hyperimmune.

Theorem 2.3 *The set of non- K -random strings is wtt-complete.*

Proof. It follows from the theorem (see [8]) that every effectively simple, non-hypersimple set is *wtt*-complete. \square

Theorem 2.4 (Kummer [9]) *The set of non- K -random strings can be *tt*-complete or non-*tt*-complete, depending on the acceptable numbering of the partial computable functions.*

Proof. Lachlan has shown (see [10]) that Post's construction of a simple set can produce both *tt*-complete and non-*tt*-complete effectively simple sets depending on which acceptable numbering of partial computable functions we are working with. Therefore we can now transfer this property to the set of non- K -random strings. \square

It is known that all results proven for Kolmogorov's definition of random strings hold for Chaitin's model of random strings. The underlying complexities H and K are "asymptotical equivalent". Moreover, it is known that Chaitin's definition of randomness is more demanding than Kolmogorov's one (see Calude [3]). The following modified Post's construction of a simple *tt*-complete set (see [13]) "effectively approximates" a proper subset of the set of K -random strings which is effectively immune, non-hyperimmune and co-c. e. *Question: it would be interesting to determine whether or not a variant of Post's construction could be used to effectively enumerate the set of non- C -random strings.*

Let S be a coinfinite, non-hypersimple c. e. set. Then there exists a disjoint strong array $\{F_n\}_{n \in \omega}$ such that $F_n \cap \bar{S} \neq \emptyset$ for all n . We will construct the desired c. e. superset S^* of S by meeting the following list of requirements:

$$\mathcal{R}_e : n \in W_e \quad \Longleftrightarrow \quad S^* \models_{tt} tt_{f(e,n)},$$

that is, $n \in W_e$ if and only if the *tt*-condition with the index $f(e, n)$ satisfies S^* and $f(e, n)$ is a computable function to be constructed.

Obviously, if we construct $S^* \supseteq S$ meeting all these requirements, then $W_e \leq_{tt} S^*$ for any e and the theorem will be proved.

We first effectively split the strong array $\{F_n\}_{n \in \omega}$ into the computable sequence of strong arrays $\{F(e, n)\}_{(e,n) \in \omega \times \omega}$ defining for all e, n : $F(e, n) = F_{\langle e, n \rangle}$, so that we will connect the requirement \mathcal{R}_e to the array $\{F(e, n)\}_{n \in \omega}$. (Here $\langle e, n \rangle = 1/2(e^2 + 2en + n^2 + 3e + n)$ denotes the standard pairing function from $\omega \times \omega$ onto ω , e. g. see Soare [15], page 3.) We define (for fixed e and n) the value of the function $f(e, n)$ as follows. Let $F(e, n) = \{a_1, a_2, \dots, a_k\}$. Then $f(e, n)$ gives the index of the *tt*-condition $\langle \{a_1, a_2, \dots, a_k\}, \eta \rangle$, where η is the following Boolean function of k arguments:

$$\eta(x_1, x_2, \dots, x_k) = 1 \text{ if and only if } x_1 = 1 \ \& \ x_2 = 1 \ \& \ \dots \ \& \ x_k = 1.$$

Now let S^* be the following c. e. superset of S :

$$S^* = S \cup \bigcup_{n \in W_e} F(e, n).$$

If $n \notin W_e$, then by the construction, $F(e, n) \cap \overline{S^*} \neq \emptyset$. It follows that $\eta(S^*(a_1), S^*(a_2), \dots, S^*(a_k)) = 0$ and, therefore, the tt -condition $tt_{f(e, n)}$ is not satisfied by S^* . If $n \in W_e$ then $F(e, n) \subseteq S^*$, and

$$\eta(S^*(a_1), S^*(a_2), \dots, S^*(a_k)) = 1.$$

Therefore, we have $n \in W_e$ if and only if the tt -condition $tt_{f(e, n)}$ satisfies S^* .

Above we considered sets $P(n)$ and $Q(n)$ defined by Chaitin as sets which reflect information-theoretic aspects of Post's simple set. Generalizing his ideas to Dekker's construction of hypersimple sets (see [7]) and to known constructions of effectively hypersimple sets, we arrive at the below definition, which we believe reflects the information-theoretic aspects of these sets. Let Φ denote a Turing machine which is total and injective, i. e. it has the following properties.

- 1) Every program computes some string, i. e. $\Phi(u)$ converges for all programs u ;
- 2) Different programs compute on Φ different strings: if $u \neq v$ then $\Phi(u) \neq \Phi(v)$.

Definition 2.5 Let

$$H_\Phi = \{\alpha : \Phi(\alpha) = \alpha_1 \implies (\exists \beta)(|\beta| > |\alpha| \text{ and } \Phi(\beta) = \beta_1 \text{ and } |\alpha_1| > |\beta_1|)\}.$$

Then, H_Φ is the set of all programs α such that if α computes a string α_1 then there exists a program β such that $|\beta| > |\alpha|$ and β computes a program β_1 with $|\alpha_1| > |\beta_1|$.

Obviously, this definition can be considered as a version of Dekker's original hypersimple set (see [7]). The following theorem about H_Φ holds true.

Theorem 2.6 *For any Turing machine Φ , let A be the c. e. set of all strings which are computed by Φ , i.e. $A = \text{range}(\Phi)$. If the set A is non-computable then H_Φ is hypersimple.*

Proof. The proof has been motivated by the original proof of Dekker's theorem. Obviously, the set H_Φ is c. e. and, since Φ computes different strings for different programs, the set $\overline{H_\Phi}$ is infinite. Let $\overline{H_\Phi} = \{\beta_0 < \beta_1 < \dots\}$. Then, by the definition of H_Φ , we have for any β that

$$\beta \in A \iff \beta \in \{\alpha_0, \alpha_1, \dots, \alpha_{b_\beta}\}.$$

Now, if $\overline{H_\Phi}$ is majorized by a computable function g , then it follows that

$$\beta \in A \iff \beta \in \{\alpha_0, \alpha_1, \dots, \alpha_{g(\beta)}\},$$

which means that A is computable. This is a contradiction. \square

Corollary 2.7 *For any Turing c. e. degree $\mathbf{a} > \mathbf{0}$ there exists a Turing machine Φ such that H_Φ is a hypersimple set of degree \mathbf{a} .*

Proof. It is easy to see that in Theorem 2.6 the set H_Φ has the same Turing degree as the set A . \square

3 Hyperimmunity and K -bounded sequences

In this section we study a special kind of hyperimmune set. Basing on this notion we introduce a new notion of complexity for infinite sequences of finite strings.

Definition 3.1 A sequence $\{F_n\}_{n \in \omega}$ of finite sets is a **disjoint strong (and singular) array** if there is a computable function f such that:

- $F_n = D_{f(n)}$ for all n ;
- $n \neq m \Rightarrow D_{f(n)} \neq D_{f(m)}$ for all n, m ;
- (and $|D_{f(n)}| = 1$ for all n).

In the early forties, Post introduced a *hyperimmune* set with computably enumerable complement in order to solve *Post's Problem* (see Soare [15] or Post [13]) for *tt*-reducibility. The intuition which led to the definition of a hyperimmune set was to strengthen the notion of simple set, which solved Post's Problem for *m*-reducibility, but did not solve Post's Problem for *tt*-reducibility. The idea was to consider in the definition of an immune set A infinite c. e. sets as disjoint strong singular arrays (i. e. arrays whose members are all singular sets), and to weaken this condition by replacing singular sets with finite sets, so that each F_n contains some $x \in \overline{A}$ but we cannot explicitly compute which $x \in F_n$ has this property.

Definition 3.2 A set A is **hyperimmune** if it is infinite, and there is no disjoint strong array with members all intersecting it, i. e. $F_n \cap A \neq \emptyset$ for all n .

Definition 3.3 If f and g are total functions, f **majorizes** g if $f(n) \geq g(n)$ for all n , and f **dominates** a partial function φ if $f(n) \geq \varphi(n)$ for all but finitely many n such that $\varphi(n)$ is defined. If $A = \{a_0 < a_1 < a_2 \dots\}$ is an infinite set, the **principal function** of A is p_A , where $p_A(n) = a_n$. A function f majorizes (dominates) an infinite set A if f majorizes (dominates) p_A . A set A dominates a partial function φ if $p_A(n)$ dominates φ .

Later on, in the early fifties, Kolmogorov presented to the participants of the Moscow's seminar "Recursive Arithmetic" the problem (see Uspensky [16]) of *which sets are not majorizable by computable functions*. Medvedev and Uspensky had shown independently that those sets are exactly the hyperimmune ones in the original sense of Post. Nowadays this beautiful characterization of hyperimmune sets by means of nonexistence of any majorizing computable function is often adopted as a definition of those sets (e. g. see Rogers [14]).

Theorem 3.4 (see [12] or [17]) *An infinite set A is hyperimmune if and only if no computable function f majorizes A .*

The notion of effectively hyperimmune set was defined as a natural effectivization of the definition of the hyperimmune set.

Definition 3.5 [see [2]] An infinite set $A = \{a_0, a_1, \dots\}$ is **effectively hyperimmune** if and only if there is a computable function f such that for any e ,

$$\varphi_e \text{ total} \implies (\exists n \leq f(e))(a_n > \varphi_e(n)).$$

Knowing an index e of a total computable function φ_e , we effectively find the interval $\{0, 1, \dots, f(e)\}$ such that the function φ_e does not dominate A via a witness n from this interval.

The notion of effectively hyperimmune set naturally suggests studying the following notion of complexity for infinite sequences of finite strings. Let $U(e, x)$ be a universal Turing machine defined on $\omega \times \omega$, i. e. $U(e, x) = \varphi_e(x)$ for every e and φ_e is the Turing program with Gödel number e . Let $A = \{\alpha_0, \alpha_1, \alpha_2, \dots\}$ be an infinite sequence of binary strings. Here and below we write $\alpha \subseteq \beta$ if $\alpha * \gamma = \beta$ for some $\gamma \neq \emptyset$.

Definition 3.6 We will say that the infinite sequence A of binary strings is φ_e -**bounded** for a fixed φ_e if the following three conditions hold:

- a) $(\forall i, j) (i < j \implies K(\alpha_i) < K(\alpha_j))$,
- b) $(\forall i) (\alpha_i \subseteq \alpha_{i+1})$,
- c) $(\exists n) (\forall m > n) (\varphi_e(m) < \infty \text{ and } \varphi_e(m) < K(\alpha_m))$.

We can easily see that for any total computable function φ_e there exist φ_e -bounded sequences. Indeed, let us fix an enumeration of all binary strings and define by induction the following sequence:

$$\begin{aligned} \alpha_0 &= \mu\beta \{ \varphi_e(0) < K(\beta) \}, \\ \alpha_{n+1} &= \mu\beta \{ \alpha_n \subseteq \beta \ \& \ \varphi_e(n+1) < K(\beta) \ \& \ K(\alpha_n) < K(\beta) \}. \end{aligned}$$

It is obvious that the sequence $\{\alpha_0, \alpha_1, \dots\}$ is φ_e -bounded.

Definition 3.7 We say that a sequence A is **K -bounded** if it is φ -bounded for every total computable function φ .

Again, it is easy to see, that there are K -bounded sequences. Indeed, let $A = \{a_0, a_1, \dots\}$ be an infinite set which majorizes all partial computable functions. Now we define the sequence $\{\alpha_0, \alpha_1, \dots\}$ as follows:

$$\alpha_0 = \mu\beta\{K(\beta) > a_0\},$$

$$\alpha_{n+1} = \mu\beta\{\alpha_n \subseteq \beta \ \& \ K(\beta) > a_{n+1} \ \& \ K(\beta) < K(\alpha_n)\}.$$

It is obvious that the sequence $\{\alpha_0, \alpha_1, \dots\}$ is K -bounded.

In the next theorem, which is our main theorem, we prove that in the definition of K -bounded sequence we can change the condition

$$(\exists n > 0) (\forall m > n) (\varphi_e(m) < \infty \text{ and } \varphi_e(m) < K(\alpha_m))$$

to the apparently much weaker condition

$$(\exists f \leq_T \emptyset) (\forall e) (\exists y) (\varphi_e \text{ total} \implies y \leq f(e) \ \& \ K(\alpha_y) \geq \varphi_e(y)).$$

Theorem 3.8 *Let A be an infinite sequence of binary strings $\alpha_0, \alpha_1, \dots$ such that the following properties hold:*

- $(\forall i, j) (i < j \implies K(\alpha_i) < K(\alpha_j)),$
- $(\forall i) (\alpha_i \subseteq \alpha_{i+1}),$
- *there is a computable function f such that for any e , if φ_e is total then for some $y \leq f(e)$, $K(\alpha_y) > U(e, y)$.*

Then A is a K -bounded sequence.

Remark 3.9 It follows that if a set A satisfies the hypotheses of the theorem, then A is *effectively K -bounded* in the sense that for any φ_e we can effectively compute the place $n(e)$ (which obviously depends on e) from where the sequence is φ_e -bounded, i.e. $(\forall m > n(e)) (\varphi_e(m) < \infty \implies \varphi_e(m) < K(\alpha_m))$.

The previous example gives a K -bounded sequence with $n(e) \leq e$:

$$(\forall m > e) (\varphi_e(m) < \infty \implies \varphi_e(m) < K(\alpha_m)).$$

Indeed, if for infinitely many e ,

$$(\exists m > e) (\varphi_e(m) < \infty \implies \varphi_e(m) \geq K(\alpha_m)),$$

then A does not majorize the following partial computable function f : for all e , $f(e) = \varphi_e(e)$ if $\varphi_e(e) < \infty$, and $f(e)$ be undefined otherwise. This contradicts the choice of A .²

² It is proved in [2] that a set dominating all partial computable functions is effectively immune.

Proof of Theorem 3.8 The proof will immediately follow from Theorem 3.11 below, which is interesting on its own and was already proved in [2]. We present here a new and simpler proof based on Lemma 3.10.

Lemma 3.10 *Let g be an increasing computable function. Then there exists a increasing computable function α such that for any x :*

1. $\varphi_{\alpha(x)}$ is a non-decreasing computable function;
2. $\varphi_{\alpha(x)}(0) \geq g(\alpha(x+1))$ for all $x \geq x_0$, where x_0 is some fixed number. Further, there exists a computable procedure which given an index of g produces this number x_0 .

Proof. Let β be an increasing computable function such that $\beta(0) = 1$, and, for $e > 0$,

$$\varphi_{\beta(e)}(n) = g(\varphi_e \varphi_e(n))$$

Let b be an index > 0 of β . Then,

$$\varphi_{\beta(b)}(n) = g(\varphi_b \varphi_b(n)) = g\beta\beta(n) = g(\beta)^2(n);$$

$$\varphi_{\beta\beta(b)}(n) = g\varphi_{\beta(b)}\varphi_{\beta(b)}(n) = gg(\beta)^2g(\beta)^2(n) \geq g(\beta)^4(n);$$

And for any $x > 0$,

$$\varphi_{\beta^x(b)}(n) \geq g(\beta^{2x}(n)).$$

Let $\alpha(x) = \beta^x(b)$, if $x > 0$, and $\alpha(0) = b$.

Then we have, for $x \geq b+1$,

$$\begin{aligned} \varphi_{\alpha(x)}(0) &\geq g(\beta^{2x})(0) \geq \\ &\geq g\beta^{x+b+1}(0) \geq g\beta^{x+1}\beta^b(0) \geq g\beta^{x+1}(b) = g(\alpha(x+1)). \end{aligned}$$

□

□

Theorem 3.11 *Let $A = \{a_0 < a_1 < a_2 \dots\}$ be an effectively hyperimmune set and h be a computable function such that*

$$(\forall x)(\varphi_x \text{ is total} \implies (\exists y \leq h(x))(\varphi_x(y) < a_y)).$$

Let f be an arbitrary increasing computable function. Then for some x_0 , which can be computed from the index of f , and for any $x > x_0$, we have $a_x > f(x)$.

Proof. Without loss of generality, we assume that the given set A is effectively hyperimmune via an increasing computable function h . Then let α and x_0 be obtained from $g = fh$ as in Lemma 3.10. For any $n \geq h(\alpha(x_0))$ there exists $x \geq x_0$ such that

$$h(\alpha(x)) \leq n < h(\alpha(x+1)).$$

Since A is effectively hyperimmune, we have

$$(\exists t \leq h(\alpha(x)))[a_t > \varphi_{\alpha(x)}(t) \geq \varphi_{\alpha(x)}(0)].$$

So, if n, x are as before, with $x \geq x_0$, we have:

$$a_n \geq a_{h(\alpha(x))} > \varphi_{\alpha(x)}(0) \geq fh(\alpha(x+1)) > f(n).$$

□

Thus, for any effectively hyperimmune set we can get a K -bounded sequence as in the previous examples.

Acknowledgements

I would like to thank Cris Calude, Douglas Bridges, Douglas Cenzer, John Crossley and one anonymous referee for their interest in this research and numerous suggestions on improvements of the paper. I would like to thank Graham Farr and Chris Wallace for their interest and support of this work. This work was supported in part by Australian Research Council Large Grant A49703170.

References

- [1] Arslanov, A., *On the difference between two notions of randomness for finite objects*, in preparation.
- [2] Arslanov, M.M., *Effectively hyperimmune sets and majorants*, Mat. Zametki **38** (1985), 302–309.
- [3] Calude, C.S., “Information and Randomness. An Algorithmic Perspective,” Springer–Verlag, Berlin, 1994.
- [4] Calude, C.S., *Algorithmic Information theory: Open Problems*, Journal of Universal Computer Science **5** (1996), 439–441.
- [5] Chaitin, G.J., “Algorithmic Information Theory,” third printing 1990, Cambridge University Press, Cambridge, 1987.
- [6] Chaitin, G.J., “Information, Randomness and Incompleteness, Papers on Algorithmic Information Theory,” second edition, World Scientific, 1987.

- [7] Dekker, J.C.E., *A theorem on hypersimple sets*, Proc. Amer. Math. Soc. **5** (1954), 791–796.
- [8] Kanovich, M.I., *On the decision complexity of a recursively enumerable set as a criterion for its completeness*, Dokl. Acad. Nauk **194** (1970), 500–503.
- [9] Kummer, M., *On the complexity of random strings*, Lecture Notes in Computer Science, STACS'96 **1046** (1996), 25–36.
- [10] Lachlan, A.H., *wtt-complete sets are not necessarily tt-complete*, Proc. Amer. Math. Soc. **48** (1975), 429–434.
- [11] Li, M., and P.M. Vitányi, “An Introduction to Kolmogorov Complexity and Its Applications,” Springer-Verlag, Berlin, 1993.
- [12] Medvedev, Y., *On non-isomorphic recursively enumerable sets*, Dokl. Acad. Nauk **102** (1955), 211–214.
- [13] Post, E.L., *Recursively enumerable sets of positive integers and their decision problems*, Bull. Amer. Math. Soc. **50** (1944), 284–316.
- [14] Rogers, H., Jr., “Theory of Recursive Functions and Effective Computability,” McGraw-Hill, New York, 1967.
- [15] Soare, R.I., “Recursively Enumerable Sets and Degrees,” Springer-Verlag, Berlin, 1987.
- [16] Uspensky, V.A., *Kolmogorov and mathematical logic*, J. Symbolic Logic **57** (1992), 385–412.
- [17] Uspensky, V.A., *Some remarks on r. e. sets*, Zeit. Math. Log. Grund. Math. **3** (1957), 157–170.