Theory of Computing Systems manuscript No.
(will be inserted by the editor)

Time-bounded Kolmogorov complexity and Solovay functions

Rupert Hölzl · Thorsten Kräling · Wolfgang Merkle

Received: date / Accepted: date

Abstract A Solovay function is an upper bound g for prefix-free Kolmogorov complexity K that is nontrivial in the sense that g agrees with K, up to some additive constant, on infinitely many places n. We obtain natural examples of computable Solovay functions by showing that for some constant c_0 and all computable functions t such that $c_0 n \le t(n)$, the time-bounded version K^t of K is a Solovay function.

By unifying results of Bienvenu and Downey and of Miller, we show that a right-computable upper bound g of K is a Solovay function if and only if $\Omega_g = \sum 2^{-g(n)}$ is Martin-Löf random. We obtain as a corollary that the Martin-Löf randomness of the various variants of Chaitin's Ω extends to the time-bounded case in so far as Ω_{K^t} is Martin-Löf random for any t as above.

As a step in the direction of a characterization of K-triviality in terms of jump-traceability, we demonstrate that a set A is K-trivial if and only if A is $\mathrm{O}(g(n) - \mathrm{K}(n))$-jump traceable for all computable Solovay functions g. Furthermore, this equivalence remains true when the universal quantification over all computable Solovay functions in the second statement is restricted either to all functions of the form K^t for some function t as above or to a single function K^t of this form.

The first author has been supported by the German Science Foundation (DFG) under grant ME 1806/3-1 and by a Feodor Lynen postdoctoral research fellowship by the Alexander von Humboldt Foundation.

Rupert Hölzl
LIAFA, CNRS & Université Paris 7, Case 7014, 75205 Paris Cedex 13, France
E-mail: r@hoelzl.fr

Thorsten Kräling · Wolfgang Merkle
Ruprecht-Karls-Universität Heidelberg, Institut für Informatik, Im Neuenheimer Feld 294, 69120 Heidelberg, Germany
E-mail: {thorsten.kraeling,merkle}@math.uni-heidelberg.de

Finally, we investigate into the plain Kolmogorov complexity C and its time-bounded variant C^t of initial segments of computably enumerable sets. Our main theorem here asserts that every high c.e. Turing degree contains a c.e. set B such that for any computable function t there is a constant $c_t > 0$ such that for all m it holds that $\mathrm{C}^t(B \upharpoonright m) \geq c_t \cdot m$, whereas for any nonhigh c.e. set A there is a computable time bound t and a constant c such that for infinitely many m it holds that $\mathrm{C}^t(A \upharpoonright m) \leq \log m + c$. By similar methods it can be shown that any high degree contains a set B such that $\mathrm{C}^t(B \upharpoonright m) \geq^+ m/4$. The constructed sets B have low unbounded but high time-bounded Kolmogorov complexity, and accordingly we obtain an alternative proof of the result due to Juedes, Lathrop, and Lutz [9] that every high degree contains a strongly deep set.

1 Introduction and overview

The Kolmogorov complexity [12] of a binary string is defined as the length of a shortest description of the string with respect to an appropriate universal Turing machine. The two most relevant variants of Kolmogorov complexity are plain Kolmogorov complexity C and prefix-free complexity K, where in the latter case the set of possible descriptions is required to be prefix-free. Both variants are neither computable, nor do they allow for unbounded computable lower bounds, as can be seen by the following argument resembling the Berry Paradox: Given such a lower bound f, the least natural number n such that $f(n) > k$ would have complexity of at least k but would have effective descriptions of size at most $2 \log k + \mathrm{O}(1)$ for both the plain and the prefix-free variant, which yields a contradiction for large enough k.

On the other hand, for both the plain and the prefix-free variant of Kolmogorov complexity there exist computable upper bounds g that are nontrivial in the sense that $g(n)$ agrees with $\mathrm{C}(n)$ and $\mathrm{K}(n)$, respectively, up to some additive constant, for infinitely many places n. Such an upper bound is easily obtained by letting $g(n) = n$ in the case of plain Kolmogorov complexity, while in the case of prefix-free complexity a corresponding construction that goes back to Solovay [4,7,16] will be reviewed in Section 2. The term Solovay function will refer to any not necessarily computable nontrivial upper bound of K.

For any computable time-bound t, the time-bounded version K^t of K is obviously a computable upper bound for K, and we show that K^t is indeed a computable Solovay function in case $c_0 n \leq t(n)$ for some appropriate constant c_0. As a corollary, we obtain that the Martin-Löf randomness of the various variants of Chaitin's Ω extends to the time-bounded case in so far as for any t as above, the real number

$$\Omega_{\mathrm{K}^t} = \sum_{n \in \mathbb{N}} \frac{1}{2^{\mathrm{K}^t(n)}}$$

is Martin-Löf random. The corresponding proof exploits the result by Bienvenu and Downey [4] that a computable function g such that $\Omega_g = \sum 2^{-g(n)}$ converges is a Solovay function if and only if Ω_g is Martin-Löf random. In fact, this equivalence extends by an even simpler proof to the case of functions g that are just right-computable, i.e., effectively approximable from above, and one then obtains as special cases the result of Bienvenu and Downey and a related result of Miller where the role of g is played by the fixed right-computable but noncomputable function K.

An open problem that received some attention recently [1,7,14] is whether the class of K-trivial sets coincides with the class of sets that are $g(n)$-jump-traceable for all computable functions g such that $\sum 2^{-g(n)}$ converges. This question was raised because the class of strongly jump-traceable sets is a proper subclass of the K-trivial sets which in turn is a proper subclass of the jump-traceable sets [1,5,6].

As a step in the direction of a characterization of K-triviality in terms of jump-traceability, we demonstrate that a set A is K-trivial if and only if A is $\mathrm{O}(g(n) - \mathrm{K}(n))$-jump traceable for all computable Solovay functions g. Furthermore, this equivalence remains true when the universal quantification over all computable Solovay functions in the second statement is restricted either to all functions of the form K^t for some function t as above or to a single function K^t of this form.

Finally, we consider the time-bounded and unbounded Kolmogorov complexity of the initial segments of sets that are computably enumerable (c.e.). The initial segments of a c.e. set A have small Kolmogorov complexity, more precisely, by Barzdins' lemma it holds that $\mathrm{C}(A \restriction m) \leq 2 \log m + c$ for some constant c.

By Kummer's celebrated gap theorem [7,10], any array noncomputable c.e. Turing degree contains a c.e. set B such that there are infinitely many m such that $\mathrm{C}(B \restriction m) \geq 2 \log m$, whereas all c.e. sets in an array computable Turing degree satisfy $\mathrm{C}(A \restriction m) \leq (1 + \varepsilon) \log m$ for all $\varepsilon > 0$ and almost all m. Theorem 21, the main result of Section 4, has a structure similar to Kummer's gap theorem in so far as it asserts a dichotomy with respect to the complexity of initial segments between high and nonhigh c.e. sets. More precisely, every high c.e. Turing degree contains a c.e. set B such that for any computable function t there is a constant $c_t > 0$ such that for all m it holds that $\mathrm{C}^t(B \restriction m) \geq c_t \cdot m$, whereas for any nonhigh c.e. set A there is a computable time bound t and a constant c such that for infinitely many m it holds that $\mathrm{C}^t(A \restriction m) \leq \log m + c$. By similar methods it can be shown that any high degree contains a set B such that $\mathrm{C}^t(B \restriction m) \geq m/4 + c$ for some constant c. The constructed sets B have low unbounded but high time-bounded Kolmogorov complexity, and accordingly we obtain an alternative proof of the result due to Juedes, Lathrop, and Lutz [9] that every high degree contains a strongly deep set.

Notation. We identify strings with natural numbers by the order isomorphism between the length-lexicographical order on strings and the usual order on $\mathbb{N}$, and we write $|\sigma|$ for the length of the string σ, where then $|\sigma|$ is roughly $\log m$ in case the string σ corresponds to the natural number m. A Turing machine is prefix-free if it has prefix-free domain and an oracle Turing machine is prefix-free if it has prefix-free domain for any fixed oracle.

Given an oracle Turing machine M, for any oracle A and string x, let $\mathrm{C}_M^A(x)$ be the plain Kolmogorov complexity of x with respect to M relative to oracle A, i.e., $\mathrm{C}_M^A(x) = |\sigma|$ for the shortest string σ such that $M^A(\sigma) = x$, and usually we write $\mathrm{K}_M^A(x)$ in place of $\mathrm{C}_M^A(x)$ in case M is prefix-free. In order to define plain and prefix-free Kolmogorov complexity, we fix an additively optimal oracle Turing machine $\mathbb{V}$ and a prefix-free oracle Turing machine $\mathbb{U}$ that is additively optimal for all prefix-free oracle Turing machines. We let $\mathrm{C}^A(x)$ denote the Kolmogorov-complexity of x with respect to $\mathbb{V}$ relative to oracle A, let $\mathrm{C}(x) = \mathrm{C}^{\emptyset}(x)$, and similarly define prefix-free Kolmogorov complexity K^A and K with respect to $\mathbb{U}$.

For a computable function $t\colon \mathbb{N} \to \mathbb{N}$ and a machine M, the Kolmogorov complexity relative to M with time bound t is

$$\mathrm{C}_M^t(x) := \min\{|\sigma|\colon M(\sigma)\downarrow= x \text{ in at most } t(|x|) \text{ steps}\},$$

and we write C^t for $\mathrm{C}_{\mathbb{V}}^t$. The prefix-free Kolmogorov complexity with time bound t denoted by $\mathrm{K}_M^t(x)$ and $\mathrm{K}^t(x) = \mathrm{K}_{\mathbb{U}}^t(x)$ is defined likewise by considering only prefix-free machines and the corresponding universal machine $\mathbb{U}$ in place of $\mathbb{V}$. In connection with the definition of time-bounded Kolmogorov complexity, we assume that $\mathbb{V}$ can simulate any other Turing machine M that runs in $t(n)$ steps in $\mathrm{O}(t(n) \cdot \log(t(n)))$ steps and even in $\mathrm{O}(t(n))$ steps in case M has only two work tapes, and that a corresponding statement is true for simulations of prefix-free machines by $\mathbb{U}$. For the existence of $\mathbb{U}$ and $\mathbb{V}$ with these properties, see the references [12].

Given two functions f and g, we write $f(n) \leq^+ g(n)$ to denote that there exists a constant c such that $f(n) \leq g(n) + c$ for all n. The relation $\geq^+$ is defined likewise. If $f(n) \leq^+ g(n)$ and $g(n) \leq^+ f(n)$, we also write $f(n) =^+ g(n)$. By abuse of notation, we will occasionally say that $f(n) \leq^+ g(n)$ (or $f(n) =^+ g(n)$) holds for all n in some subset I of $\mathbb{N}$, with the obvious meaning that for some constant c and all n in I we have $f(n) \leq g(n) + c$.

2 Solovay functions and Martin-Löf randomness

Definition 1 A function $f\colon \mathbb{N} \to \mathbb{N}$ is an A-Solovay function if $\mathrm{K}^A(n) \leq^+ f(n)$ for all n and $\mathrm{K}^A(n) =^+ f(n)$ for infinitely many n. In case $A = \emptyset$, we say that f is a Solovay function.

Solovay [4,16] had already constructed computable Solovay functions. By slightly varying the standard construction, we observe next that time-bounded prefix-free Kolmogorov complexity provides natural examples of computable Solovay functions.

Theorem 2 *There is a constant c_0 such that time-bounded prefix-free Kolmogorov complexity K^t is a computable Solovay function for any computable function $t\colon \mathbb{N} \to \mathbb{N}$ such that $c_0 n \leq t(n)$ holds for almost all n.*

Proof Let $\langle .,. \rangle\colon \mathbb{N}^2 \to \mathbb{N}$ be a standard effective and effectively invertible pairing function and define a tripling function $[.,.,.]\colon \mathbb{N}^3 \to \{0,1\}^{<\infty}$ by letting

$$[s, \sigma, n] = 1^s 0 \langle \sigma, n \rangle,$$

where the right-hand side of this equation is meant as concatenation of the string $1^s 0$ and the string that is identified with the natural number $\langle \sigma, n \rangle$.

Let M be a Turing machine with two tapes that on input σ uses its first tape to simulate the universal machine $\mathbb{U}$ on input σ and, in case $\mathbb{U}(\sigma) = n$, to compute $\langle \sigma, n \rangle$, while maintaining on the second tape a unary counter for the number of steps of M required for these computations. In case eventually $\langle \sigma, n \rangle$ could be computed with final counter value s, the output of M is $z = [s, \sigma, n]$, where by construction in this case the total running time of M is in $\mathrm{O}(s)$.

Call z of the form $[s, \sigma, n]$ a Solovay triple in case $M(\sigma) = z$, σ is an optimal code for n, i.e., $\mathrm{K}(n) = |\sigma|$, and s is the number of steps it takes until the computation of M on input σ stops. For some appropriate constant c_0 and any computable function t that eventually is at least $c_0 n$, for almost all such triples z it then holds that

$$\mathrm{K}(z) =^+ \mathrm{K}^t(z),$$

because, given a code for M and σ, by assumption the universal machine $\mathbb{U}$ can simulate the computation of the two-tape machine M on input σ with linear overhead, hence the simulation by $\mathbb{U}$ uses at most time $\mathrm{O}(s)$, i.e., at most time $\mathrm{O}(|z|)$. □

Next we derive a unified version of results by Bienvenu and Downey [4] and by Miller [13] where the former result characterizes computable Solovay functions in terms of Martin-Löf randomness of the corresponding Ω-numbers and the latter one asserts that the notions of weakly low and low for Ω coincide. First we review some standard notation and facts relating to Ω-numbers.

Definition 3 For a function $f\colon \mathbb{N} \to \mathbb{N}$, the Ω-number of f is

$$\Omega_f := \sum_{n \in \mathbb{N}} 2^{-f(n)}.$$

As usual, given any set A, we write Ω^A_{K} for $\Omega_{\mathrm{K}^A} = \sum_{i \in \mathbb{N}} 2^{-\mathrm{K}^A(n)}$.

In connection with Ω numbers, recall that a real number with binary expansion $0.a_0 a_1 \ldots$ can be identified with the infinite binary sequence $a_0 a_1 \ldots$, which in turn can be identified with the subset of the natural numbers that contains i if and only if a_i is equal to 1. By this identification, concepts and notation such as Martin-Löf randomness extend from sets to real numbers or vice versa.

Definition 4 A function $f\colon \mathbb{N} \to \mathbb{N}$ is an information content measure relative to a set A in case f is right-computable with access to the oracle A and Ω_f converges; furthermore, the function f is an information content measure if it is an information content measure relative to the empty set.

The following remark describes for a given information content measure f an approximation from below to Ω_f that has certain special properties. For the sake of simplicity, in the remark only the oracle-free case is considered and the virtually identical considerations for the general case are omitted.

Remark 5 For a given information content measure f, we fix as follows a nondecreasing computable sequence $a_0, a_1, \ldots$ that converges to Ω_f and call this sequence the canonical approximation of Ω_f.

First, we fix some standard approximation to the given information content measure f from above, i.e., a computable function $(n,s) \mapsto f_s(n)$ such that for all n the sequence $f_0(n), f_1(n), \ldots$ is a nonascending sequence of natural numbers that converges to $f(n)$ where we assume in addition that $f_s(n) - f_{s+1}(n)$ is always equal to either 0 or 1. Then in order to obtain the a_i, let $a_0 = 0$ and given a_i, and with some ordering of pairs understood, define a_{i+1} by searching for the next pair of either the form $(n,0)$ or the form $(n,s+1)$ where in addition it holds that $f_s(n) - f_{s+1}(n) = 1$, let

$$d_i = 2^{-f_0(n)} \qquad \text{or} \qquad d_i = 2^{-f_{s+1}(n)} - 2^{-f_s(n)} = 2^{-f_s(n)},$$

respectively, and let $a_{i+1} = a_i + d_i$. Furthermore, in this situation, say that the increase from a_i to a_{i+1} of size d_i occurs due to n.

It is well-known [7] that among all right-computable functions exactly the information content measures are, up to an additive constant, upper bounds for the prefix-free Kolmogorov complexity K, and the same applies relative to any oracle A.

Theorem 6 unifies two results by Bienvenu and Downey [4] and by Miller [13], which are stated below as Corollaries 7 and 11. The proof of the backward direction of the equivalence stated in Theorem 6 is somewhat more direct and uses different methods when compared to the proof of Bienvenu and Downey, and is quite a bit shorter than Miller's proof, though the main trick of delaying the enumeration via the notion of a matched increase is already implicit in the latter [7,13]. Note in this connection that Bienvenu has independently shown that Miller's result can be obtained as a corollary to the result of Bienvenu and Downey [7].

Theorem 6 *Let f be an information content measure relative to a set A. Then f is an A-Solovay function if and only if Ω_f is Martin-Löf random relative to A.*

Proof We first show the backwards direction of the equivalence asserted in the theorem, where the construction and its verification bear some similarities to

Kučera and Slaman's [11] proof that left-computable sets that are not Solovay complete cannot be Martin-Löf random. We assume that f is not an A-Solovay function and construct a sequence $U_0, U_1, \dots$ of sets that is a Martin-Löf test relative to A and covers Ω_f. In order to obtain the component U_c, let $a_0, a_1, \dots$ be the canonical approximation to Ω_f where in particular $a_{i+1} = a_i + d_i$ for increases d_i that occur due to some n. Let $b_{i,n}$ be the sum of the first i increases d_j that are due to n. Say that $b_{i,n}$ is c-matched if it holds that

$$b_{i,n} \leq \frac{2^{-\mathrm{K}^A(n)}}{2^{c+1}}. \tag{1}$$

For every $b_{i,n}$ for which it could be verified that it is c-matched, let j be the largest index such that d_j contributes to $b_{i,n}$. Now add an interval of size $2d_j$ to U_c, where this interval either starts at the maximum place that is already covered by U_c or at a_ℓ, whichever is larger. Here ℓ is the number of steps in the approximation of Ω_f that we have made, where we need to make sure that ℓ is at least j (if it is not we can just approximate Ω_f further until it is).

By construction, for all $b_{i,n}$ that are c-matched, (1) together with the trivial bound $\sum_{n\in\mathbb{N}} 2^{-\mathrm{K}^A(n)} \leq 1$ implies the measure bound 2^{-c} for U_c. Also, the sets U_c are uniformly c.e. relative to A, hence $U_0, U_1, \dots$ is a Martin-Löf test relative to A. Furthermore, this test covers Ω_f because by the assumption that f is not an A-Solovay function, and by the fact stated above that information content measures relative to A are upper bounds for K^A (up to a constant), it holds that

$$\lim_{n\to\infty} (f(n) - \mathrm{K}^A(n)) = \infty.$$

Hence for any fixed c, for any j large enough the $b_{i,n}$'s to which d_j contributes will eventually become c-matched, resulting in the addition of an interval to U_c. This makes sure that almost every a_ℓ is contained in one of the intervals from which U_c is built.

For ease of reference, we review the proof of the forward direction of the equivalence asserted in the theorem, which follows by the same line of standard argument that has already been used by Bienvenu and Downey and by Miller. For a proof by contraposition, assume that Ω_f is not Martin-Löf random relative to A, that is, for every constant c there is a prefix σ_c of Ω_f such that $\mathrm{K}^A(\sigma_c) \leq |\sigma_c| - 2c$. Again consider the canonical approximation $a_0, a_1, \dots$ to Ω_f where $f(n,0), f(n,1), \dots$ is the corresponding effective approximation from above to $f(n)$ as in Remark 5. Moreover, for σ_c as above we let s_c be the least index s such that a_s exceeds σ_c. Since $\sigma_c \sqsubseteq \Omega_f$ this implies $\sigma_c \leq a_s \leq \Omega_f$; so the sum over all values $2^{-f(n)}$, where the n are such that none of the increases d_0 through d_{s_c} was due to n, is at most $2^{-|\sigma_c|}$. Hence all pairs of the form $(f(n,s) - |\sigma_c| + 1, n)$ for such n and s where either $s = 0$ or $f(n,s)$ differs from $f(n, s-1)$ form a sequence of Kraft-Chaitin axioms, which is uniformly effective in c and σ_c relative to oracle A. Observe that the approximation $f(n,s)$ will eventually reach the correct value $f(n)$; so for each n, there is an axiom of the form $(f(n) - |\sigma_c| + 1, n)$. Also, since we only add axioms to the

KC set when the approximation of $f(n)$ has actually changed, the sum of all terms 2^{-k} over all axioms of the form (k, n) is less than $2^{-f(n)-|\sigma_c|}$.

Now consider a prefix-free Turing machine M with oracle A that given codes for c and σ_c and some other word p as input, first computes c and σ_c, then searches for s_c, and finally outputs the word that is coded by p according to the Kraft-Chaitin axioms for c, if such a word exists. If we let d be the coding constant for M, we have for all sufficiently large c and n that

$$\mathrm{K}^A(n) \leq 2\log c + \mathrm{K}^A(\sigma_c) + f(n) - |\sigma_c| + 1 + d \leq f(n) - c,$$

as needed. □

As special cases of Theorem 6 we obtain the following results by Bienvenu and Downey [4] and by Miller [13], where the former is immediate and for the latter it suffices to apply the known fact that the definition of the notion low for Ω in terms of Chaitin's Ω number

$$\Omega := \sum_{\{x\colon\, \mathbb{U}(x)\downarrow\}} 2^{-|x|}.$$

is equivalent to a definition in terms of Ω_{K}, see Remark 10 below.

Corollary 7 (Bienvenu and Downey) *A computable information content measure f is a Solovay function if and only if Ω_f is Martin-Löf random.*

Definition 8 A set A is low for K if $\mathrm{K}(n) \leq^+ \mathrm{K}^A(n)$. A set A is weakly low for K if $\mathrm{K}(n) \leq^+ \mathrm{K}^A(n)$ for infinitely many n.

Definition 9 A set A is low for Ω if Ω is Martin-Löf random relative to A. A set A is low for Ω_f if Ω_f is Martin-Löf random relative to A.

Remark 10 Because all left-computable Martin-Löf random sets are mutually Solovay equivalent, it follows that a set A is low for Ω if and only if some left-computable Martin-Löf random set is Martin-Löf random relative to A if and only if all left-computable Martin-Löf random sets are Martin-Löf random relative to A [14, Proposition 8.8.1].

Corollary 11 (Miller) *A set A is weakly low for* K *if and only if A is low for Ω.*

Proof In order to see the latter result, it suffices to let $f = \mathrm{K}$ and to recall that for this choice of f the properties of A that occur in the two equivalent assertions in the conclusion of Theorem 6 coincide with the concepts weakly low and low for Ω_K. But the latter property is equivalent to being low for Ω by Remark 10. □

By Corollary 7 and Theorem 2 it is immediate that the known Martin-Löf randomness of Ω_{K} extends to the time-bounded case.

Corollary 12 *There is a constant c_0 such that $\Omega_{\mathrm{K}^t} = \sum_{x\in\mathbb{N}} 2^{-\mathrm{K}^t(x)}$ is Martin-Löf random for any computable function t where $c_0 n \leq t(n)$ for almost all n.*

3 Solovay functions and jump-traceability

In an attempt to define K-triviality without resorting to effective randomness or measure, Barmpalias, Downey and Greenberg [1] investigated into characterizations of K-triviality via jump-traceability. They demonstrated that K-triviality is not implied by being h-jump-traceable for all computable functions h such that $\sum_n 1/h(n)$ converges. Subsequently, the following question received some attention: Can K-triviality be characterized by being g-jump traceable for all computable functions g such that $\sum 2^{-g(n)}$ converges, that is, for all computable functions g that, up to an additive constant term, are upper bounds for K?

We will now argue that Solovay functions can be used for a characterization of K-triviality in terms of jump traceability. However, in doing so we will not be able to completely avoid the notion of Kolmogorov complexity.

Definition 13 A set A is K-trivial if $\mathrm{K}(A \upharpoonright n) \leq^+ \mathrm{K}(n)$ for all n.

Definition 14 Let $h\colon \mathbb{N} \to \mathbb{N}$ be a computable function. A set A is $\mathrm{O}(g(n))$-jump-traceable if for every function Φ partially computable in A there is a function $h \in \mathrm{O}(g(n))$ and a sequence $(T_n)_{n\in\mathbb{N}}$ of uniformly c.e. finite sets, which is called a trace for Φ, such that

$$|T_n| \leq h(n) \text{ for all } n, \quad \text{and} \quad \Phi(n) \in T_n \text{ for all } n \text{ where } \Phi(n) \text{ is defined.}$$

Theorem 15 *There is a constant c_0 such that the following assertions are equivalent for any set A.*

(i) A is K*-trivial.*
(ii) A is $\mathrm{O}(g(n) - \mathrm{K}(n))$-jump-traceable for every computable Solovay function g.
(iii) A is $\mathrm{O}(\mathrm{K}^t(n) - \mathrm{K}(n))$-jump-traceable for all computable functions t where $c_0 n \leq t(n)$ holds for almost all n.
(iv) A is $\mathrm{O}(\mathrm{K}^t(n) - \mathrm{K}(n))$-jump-traceable for some computable function t where $c_0 n \leq t(n)$ holds for almost all n.

Proof Fix a constant c_0 as in Theorem 2. Then the implication from (ii) to (iii) is immediate by choice of c_0, while the implication from (iii) to (iv) is trivial.

(i) implies (ii): Let A be K-trivial and let Φ^A be any partially A-computable function. Let $\langle .,. \rangle$ be some standard effective pairing function. Since A is K-trivial and hence low for K (Nies [14], see Downey and Hirschfeldt [7, Chapter 11.4]), we have

$$\mathrm{K}(\langle n, \Phi^A(n)\rangle) =^+ \mathrm{K}^A(\langle n, \Phi^A(n)\rangle) =^+ \mathrm{K}^A(n) =^+ \mathrm{K}(n), \tag{2}$$

whenever $\Phi^A(n)$ is defined. Observe that the constant that is implicit in the relation $=^+$ depends only on A in the case of the first and the last equality in (2), but depends also on Φ in the case of the middle one. Let d be a constant such that $\mathrm{K}(\langle n, \Phi^A(n)\rangle) \leq \mathrm{K}(n) + d$ for all n where $\Phi^A(n)$ is defined.

By the coding theorem [7] there can be at most constantly many pairs of the form (n, y) such that $\mathrm{K}(\langle n, y\rangle)$ exceeds $\mathrm{K}(n)$ at most by a given constant, and given n, $\mathrm{K}(n)$ and the constant, we can enumerate all such pairs. So let c be a constant such that for all n,

$$\#\{\sigma\colon |\sigma| \leq \mathrm{K}(n) + d \text{ and } \mathbb{U}(\sigma) = \langle n, y\rangle \text{ for some } y\} \leq c. \tag{3}$$

If we knew $\mathrm{K}(n)$, we could build a trace T_n for $\Phi^A(n)$ by simply trying to compute $\mathbb{U}(\sigma)$ for all strings σ of length at most $\mathrm{K}(n) + d$ and, whenever one such computation converges and outputs some string of the form $\langle n, y\rangle$, putting y in T_n. Then all T_n would have size at most c by (3) and $\Phi^A(n) \in T_n$ would follow by (2).

Since $\mathrm{K}(n)$ is not computable, this strategy will not work. Instead, we computably approximate $\mathrm{K}(n)$ from above by a decreasing sequence $\mathrm{K}_s(n)$. Here, as we know that $\mathrm{K}(n) \leq g(n) + \mathrm{O}(1)$, and since this upper bound is computable, we may as well assume that $\mathrm{K}_0(n) \leq g(n) + \mathrm{O}(1)$. As soon as a value $\mathrm{K}_s(n)$ is reached in the approximation, we apply the strategy for enumerating T_n as described, but with $\mathrm{K}_s(n)$ in place of $\mathrm{K}(n)$. We can stop the enumeration for the current value $\mathrm{K}_s(n)$ as soon as c elements have been enumerated into T_n. As soon as a new value $\mathrm{K}_{s+1}(n) < \mathrm{K}_s(n)$ is reached, we start the strategy anew, again enumerating up to c elements etc. Eventually, $\mathrm{K}_s(n)$ will drop to the true value $\mathrm{K}(n)$ and by (2) we can be sure that $\Phi^A(n)$ will be enumerated if it is defined. Since the value of K_s drops at most $g(n) - \mathrm{K}(n) + \mathrm{O}(1)$ times and for each change we enumerate at most c elements, the size of the trace thus enumerated can be at most $c \cdot (g(n) - \mathrm{K}(n) + \mathrm{O}(1))$, hence is in $\mathrm{O}(g(n) - \mathrm{K}(n))$.

(iv) implies (i): Let c_0 be the constant from Theorem 2 and let t be a computable time bound such that (iv) is true for this value of c_0. Then K^t is a computable Solovay function by choice of c_0.

Recall the tripling function $[.,.,.]$ and the concept of a Solovay triple $[s, \sigma, n]$ from the proof of Theorem 2, and define a partial A-computable function Φ that maps any Solovay triple $[s, \sigma, n]$ to $A \restriction n$. Then given an optimal code σ for n, one can compute the corresponding Solovay triple $z = [s, \sigma, n]$, where then $\mathrm{K}^t(z)$ and $\mathrm{K}(z)$ differ only by a constant, hence, by $\mathrm{O}(\mathrm{K}^t(n) - \mathrm{K}(n))$-traceability, the trace of Φ^A at z has constant size and contains the value $A \restriction n$, i.e., we have $\mathrm{K}(A \restriction n) \leq^+ |\sigma| = \mathrm{K}(n)$, hence A is K-trivial. □

The following remark follows easily by arguments similar to those in the proof of Theorem 15. The details are left to the reader.

Remark 16 A set A is K-trivial if and only if every partial function Φ that is partially computable in A has constant size c.e. traces that can be computed from optimal codes for n in the sense that for every such Φ there is a constant c and a sequence $T_0, T_1, \ldots$ of uniformly c.e. sets of size at most c such that T_{n^*} contains $\Phi(n)$ whenever n^* is a code for n of minimum length.

4 Time bounded Kolmogorov complexity and strong depth

The main result of this section stated in Theorem 21 is a dichotomy with respect to the time-bounded Kolmogorov complexity of the prefixes of high and nonhigh c.e. sets. The result is in spirit and structure very similar to Kummer's celebrated gap theorem [7,10], a dichotomy with respect to the unbounded plain Kolmogorov complexity of the prefixes of array noncomputable and array computable c.e. sets. For a start, we discuss the Kolmogorov complexity of initial segments of c.e. sets in general.

The initial segments of any given c.e. set A have small Kolmogorov complexity; by Barzdins' lemma [2,7] it holds for all m that

$$\mathrm{C}(A \upharpoonright m \mid m) \leq^{+} \log m \quad \text{and} \quad \mathrm{C}(A \upharpoonright m) \leq^{+} 2 \log m.$$

Furthermore, there are infinitely many initial segments that have considerably smaller complexity. The corresponding observation in the following remark is extremely easy, but was only noted recently [8] and, in particular, improves on corresponding statements in the literature [7, Theorem 16.1.2 attributed to Solovay].

Remark 17 Let A be a c.e. set. Then there is a constant c such that for infinitely many m it holds that

$$\mathrm{C}(A \upharpoonright m \mid m) \leq c, \;\; \mathrm{C}(A \upharpoonright m) \leq^{+} \mathrm{C}(m) + c, \;\; \text{and } \mathrm{C}(A \upharpoonright m) \leq \log m + c.$$

For a proof, it suffices to fix an effective enumeration of A and to observe that there are infinitely many $m \in A$ such that m is enumerated after all numbers $n \leq m$ that are in A, i.e., when knowing m one can simulate the enumeration until m appears, at which point one then knows $A \upharpoonright m$.

Barzdins [2] states that there are c.e. sets with high time-bounded Kolmogorov complexity, and the following lemma generalizes this in so far as such sets can be found in every high Turing degree.

Lemma 18 *For any high set A there is a set B where $A \equiv_{\mathrm{T}} B$ such that for every computable time bound t there is a constant $c_t > 0$ where*

$$\mathrm{C}^t(B \upharpoonright m) \geq^{+} c_t \cdot m \quad \text{and} \quad \mathrm{C}(B \upharpoonright m) \leq^{+} 2 \log m.$$

Moreover, if A is c.e., B can be chosen to be c.e. as well.

Proof Let A be any high set. We will construct a Turing-equivalent set B as required. Since A is high there is a function g computable in A that dominates any computable function f, i.e., $f(n) \leq g(n)$ for almost all n. Fix such a function g, and observe that in case A is c.e., we can assume that g can be effectively approximated from below. This is because otherwise we may replace g with the function g' defined as follows. Let M_g be an oracle Turing machine that computes g if supplied with oracle A. For all n, let

$$g'(n) := \max(\{0\} \cup \{M_g^{A_i}(n) \colon i \in \mathbb{N} \text{ and } M_g^{A_i}(n) \downarrow \text{in at most } i \text{ steps}\}),$$

where A_i is the approximation to A after i steps of enumeration. By construction, we have $g(n) \le g'(n)$ for all n, and g' can be effectively approximated from below and is computable in A.

Partition $\mathbb{N}$ into consecutive intervals $I_0, I_1, \dots$ where interval I_j has length 2^j and let $m_j = \max I_j$. Let $t_0, t_1, \dots$ be an effective enumeration of all partial computable functions. Observe that it is sufficient to ensure that the assertion in the theorem is true for all $t = t_i$ such that t_i is computable, nondecreasing and unbounded. Assign the potential time bounds $t_0, t_1, \dots$ to the intervals $I_0, I_1, \dots$ such that t_0 will be assigned to every second interval including the first one, t_1 to every second interval including the first one of the *remaining* intervals, and so on for $t_2, t_3, \dots$, and note that this way t_i will be assigned to every 2^{i+1}th interval.

We construct a set B as required. In order to code A into B, for all j let $B(m_j)$ be equal to $A(j)$, while the remaining bits of B are specified as follows. Fix any interval I_j and assume that this interval is assigned to $t = t_i$. Let B have empty intersection with $I_j \setminus \{m_j\}$ in case the computation of $t(m_j)$ requires more than $g(m_j)$ steps. Otherwise, run all codes of length at most $|I_j| - 2$ on the universal machine $\mathbb{V}$ for $2t(m_j)$ steps each, and let w_j be the least word of length $|I_j| - 1$ that is not output by any of these computations, hence $\mathrm{C}^{2t}(w_j) \ge |w_j| - 1$. Let the restriction of B to the first $|w_j|$ places in I_j be equal to w_j.

Now let v_j be the initial segment of B of length $m_j + 1$, i.e., up to and including I_j. In case $t = t_i$ is computable, nondecreasing and unbounded, for almost all intervals I_j assigned to t, we have $\mathrm{C}^t(v_j) > |v_j|/3$, because otherwise, for some appropriate constant c and infinitely many such intervals I_j the corresponding codes would yield $\mathrm{C}^{2t}(w_j) \le |v_j|/3 + c \le |I_j| - 2$. Furthermore, by construction for every such t there is a constant $c_t > 0$ such that for almost all m, there is some interval I_j assigned to t such that $m_j \le m$ and $c_t m \le m_j/4$. To see this, assume m could be arbitrarily large compared to m_j. If it became *too* large, due to the regular appearance of intervals assigned to t, this would imply that m is larger than the next m_k assigned to t — so replace m_j by m_k. Hence for almost all m the initial segment of B up to m cannot have Kolmogorov complexity of less than $c_t m$.

To see that $B \le_{\mathrm{T}} A$, let's compute any fixed interval I_j using A. We compute $g(m_j)$ using A and try to compute the assigned time bound $t_i(m_j)$. If this computation does not halt in at most $g(m_j)$ steps, then we know that during the construction of B no diagonalization has occurred on this interval, so we output all 0's on the positions in $(m_{j-1}, m_j - 1]$. If on the other hand the computation of $t_i(m_j)$ halts within the given number of steps, we can retrace the diagonalization done in the construction of B in order to compute the restriction w_j of B to the interval $I_j \setminus \{m_j\}$. In any case we output $A[j]$ at position m_j. In case A is c.e., the function g can be chosen to be in addition approximable from below. By simulating a corresponding approximation of $g(m_j)$, either the approximated values remain so small that the restriction of B to $I_j \setminus \{m_j\}$ is empty or the approximation reveals that $t_i(m_j)$ can be computed in at most $g(m_j)$ steps, where then w_j can be computed as before. In summary, B be can be chosen to be c.e. in case A is c.e.

Finally, to see that $\mathrm{C}(B \restriction m) \leq^+ 2\log m$ holds, notice that in order to determine $B \restriction m$ without time bounds it is enough to know on which of the intervals I_j the assigned time bounds t_i terminate before their computation is canceled by g, which requires one bit per interval, plus another one describing the bit of A coded into B at the end of each interval. □

Using a similar, but slightly simpler construction, we obtain the following variant of Lemma 18. Recall that an order is a nondecreasing and unbounded function.

Lemma 19 *Every high degree contains for every computable order h a set B such that for every computable time bound t and almost all m,*

$$\mathrm{C}^t(B \restriction m) \geq^+ \frac{1}{4}m \quad \text{and} \quad \mathrm{C}(B \restriction m) \leq h(m) \cdot \log m.$$

Proof The argument is similar to the proof of Lemma 18, but now, when considering interval I_j, we diagonalize against the largest running time among $t_0(m_j), \ldots, t_{h(j)-2}(m_j)$ such that the computation of this value requires not more than $g(j)$ steps. This way we ensure — for any computable time bound t — that *at the end* of almost all intervals I_j compression by a factor of at most 1/2 is possible, and that *within* interval I_j, we have compressibility to a factor of at most 1/4, up to a constant additive term, because $B \restriction m_{j-1}$ was compressible by a factor of at most 1/2 and $m_{j-1} = |I_j|$. □

Remark 20 The order of quantifiers in Lemma 19 can be reversed in the sense that there is a single set B that works for all computable orders h.

For a proof, it suffices to apply the construction in the proof of Lemma 19 with an order h that grows slower than any computable order h' in the sense that $h(n) \leq h'(n)$ for almost all n. Such an order h can be computed from any high set A, simply by requiring that $h(g(n)) = n$ for all n where g is an A-computable function that dominates all computable functions.

Kummer's gap theorem [7,10] asserts that any array noncomputable c.e. Turing degree contains a c.e. set A such that there are infinitely many m such that $\mathrm{C}(A \restriction m) \geq^+ 2\log m$, whereas all c.e. sets in an array computable Turing degree satisfy $\mathrm{C}(A \restriction m) \leq (1+\varepsilon)\log m$ for all $\varepsilon > 0$ and almost all m. Similarly, Theorem 21, the main result of this section, asserts a dichotomy for the time-bounded complexity of initial segments between high and nonhigh sets.

Theorem 21 *Let A be any c.e. set.*

(i) If A is high, then there is a c.e. set B with $B =_{\mathrm{T}} A$ such that for every computable time bound t there is a constant $c_t > 0$ such that for all m, it holds that $\mathrm{C}^t(B \restriction m) \geq c_t \cdot m$.

(ii) If A is not high, then there is a computable time bound t such that

$$\mathrm{C}^t(A \upharpoonright m) \leq^+ \log m$$

for infinitely many m.

Proof The first assertion is immediate from Lemma 18. In order to demonstrate the second assertion, let m_A be a modulus of convergence of the set A, i.e.,

$$\mathrm{m}_A(m) = \min\{s \geq m \colon A_s \upharpoonright m = A \upharpoonright m\},$$

where A_s is the finite set of numbers that have been enumerated by a fixed enumeration of A after s steps. The modulus m_A is obviously computable in A, and since A is not high there is a computable function f such that for infinitely many m it holds that $\mathrm{m}_A(m) \leq f(m)$. That means that there are infinitely many lengths m where $A \upharpoonright m$ can be computed by enumerating A for $f(m)$ steps. For these lengths, $A \upharpoonright m$ can be coded by providing the number m (code length $\log m$) and a constant-size code for f. Because f is computable, the time needed for the computation of $f(m)$ and for simulating the enumeration of A is computable itself, hence there is a computable time bound t as required. □

Remark 22 The second assertion in Theorem 21 does not extend in general to sets that are not c.e. This can be seen by using Schnorr's well-known theorem [15] that a set is Martin-Löf random if and only if there exists a constant c such that $\mathrm{K}(A \upharpoonright n) \geq n - c$ for all n as follows: Fix a value for c to get a Π^0_1 class of all sets that satisfy the latter condition for this choice of c. Then apply the Low Basis Theorem [7] to obtain a low, hence nonhigh ML-random set, and for such a set the second assertion in Theorem 21 cannot hold.

As another easy consequence of Lemma 18, we get an alternative proof of the result due to Juedes, Lathrop and Lutz [9] that every high degree contains a strongly deep set.

Definition 23 (Bennett [3]; Juedes, Lathrop, Lutz [9]) For functions $t, g \colon \mathbb{N} \to \mathbb{N}$, $n \in \mathbb{N}$ and $x \in \{0,1\}^{<\infty}$ let

$$\begin{aligned} \mathrm{PROG}^t(x) &:= \{p \colon \mathbb{U}(p) = x \text{ in time at most } t(|x|)\}, \\ \mathsf{D}^t_g(n) &:= \{A \colon \mathrm{K}(p) \leq |p| - g(n) \text{ for all } p \in \mathrm{PROG}^t(A \upharpoonright n)\}, \\ \mathsf{D}^t_g &:= \{A \colon A \in \mathsf{D}^t_g(n) \text{ for almost all } n\}. \end{aligned}$$

A set A is called strongly deep if A is in D^t_c for every computable time bound t and every constant $c \in \mathbb{N}$.

At this stage we cannot connect this definition with Lemma 18, since instead of comparing $\mathrm{K}(A \upharpoonright n)$ with $\mathrm{K}^t(A \upharpoonright n)$ in the definitions of D^t_g and strong depth we are comparing $|p|$ and $\mathrm{K}(p)$. This is resolved by the following lemma.

Definition 24 (Bennett [3]; Juedes, Lathrop, Lutz [9]) Define the following sets

$$\widehat{\mathsf{D}}^t_g(n) := \{A\colon \mathrm{K}(A \restriction n) \leq \mathrm{K}^t(A \restriction n) - g(n)\},$$
$$\widehat{\mathsf{D}}^t_g := \{A\colon A \in \widehat{\mathsf{D}}^t_g(n) \text{ for almost all } n\}.$$

Lemma 25 (Bennett [3], Juedes, Lathrop, Lutz [9]) *For a computable time bound t there are constants c, d and a computable time bound t' such that for all g and n,*

$$\mathsf{D}^t_{g+c}(n) \subseteq \widehat{\mathsf{D}}^t_g(n), \qquad \mathsf{D}^t_{g+c} \subseteq \widehat{\mathsf{D}}^t_g, \qquad \widehat{\mathsf{D}}^{t'}_{g+d}(n) \subseteq \mathsf{D}^t_g(n), \qquad \widehat{\mathsf{D}}^{t'}_{g+d} \subseteq \mathsf{D}^t_g.$$

Using this lemma we can now apply our previous result to prove the desired statement as a corollary of Lemma 18.

Corollary 26 *Every high degree contains a strongly deep set.*

Proof We use the set B constructed in Lemma 18. We know that for every computable time bound t there is a constant c_t such that for all n:

$$\mathrm{C}^t(B \restriction n) \geq c_t \cdot n \quad \text{and} \quad \mathrm{C}(B \restriction n) \leq 2 \log n.$$

Since $\mathrm{K}^t(x) \geq \mathrm{C}^t(x)$ and $2\mathrm{C}(x) \geq \mathrm{K}(x)$ for all words x, it follows that B is in all $\widehat{D}^t_c$. Using Lemma 25 it follows that B is strongly deep. □

References

1. George Barmpalias, Rod Downey, and Noam Greenberg. K-trivial degrees and the jump-traceability hierarchy. *Proc. Amer. Math. Soc.*, 137(6):2099–2109, 2009.
2. Janis Barzdin. Complexity of programs to determine whether natural numbers not greater than n belong to a recursively enumerable set. *Soviet Math. Dokl.*, 9:1251–1254, 1968.
3. Charles H. Bennett. Logical depth and physical complexity. In *The universal Turing machine (2nd ed.): a half-century survey*, pages 207–235. Springer, 1995.
4. Laurent Bienvenu and Rod Downey. Kolmogorov complexity and solovay functions. In *Proceedings of the 26th International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 147–158, 2009.
5. Peter Cholak, Rod Downey, and Noam Greenberg. Strong jump-traceability I: The computably enumerable case. *Advances in Mathematics*, 217(5):2045–2074, 2008.
6. Rod Downey and Noam Greenberg. Strong jump-traceability II: K-triviality. *Israel Journal of Mathematics*. To appear.
7. Rodney G. Downey and Denis R. Hirschfeldt. *Algorithmic Randomness and Complexity*. Theory and Applications of Computability. Springer, New York, 2010.
8. Rupert Hölzl, Thorsten Kräling, and Wolfgang Merkle. Time-bounded Kolmogorov complexity and Solovay functions. In *Proceedings of the 34th International Symposium on Mathematical Foundations of Computer Science*, pages 392–402, August 2009.
9. David W. Juedes, James I. Lathrop, and Jack H. Lutz. Computational depth and reducibility. *Theor. Comput. Sci.*, 132(1-2):37–70, 1994.
10. Martin Kummer. Kolmogorov complexity and instance complexity of recursively enumerable sets. *SIAM J. Comput.*, 25(6):1123–1143, 1996.

11. Antonín Kučera and Theodore A. Slaman. Randomness and recursive enumerability. *SIAM J. Comput.*, 31(1):199–211, 2001.
12. Ming Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications.* Springer, 2008.
13. Joseph S. Miller. The K-degrees, low for K-degrees and weakly low for K-degrees. *Notre Dame Journal of Formal Logic*, 50(4)(4):381–391, 2010.
14. André Nies. *Computability and Randomness.* Oxford University Press, 2009.
15. Claus P. Schnorr. Process complexity and effective random tests. *J. Comput. System Sci.*, 7:376–388, 1973.
16. Robert M. Solovay. Draft of paper (or series of papers) on Chaitin's work. 1975. Unpublished notes.