Enumerations including laconic enumerators

Sanjay Jain[*]
National University of Singapore
sanjay@comp.nus.edu.sg

Jason Teutsch[†]
TrueBit Foundation
jt@truebit.io

November 3, 2017

Abstract

We show that it is possible, for every machine universal for Kolmogorov complexity, to enumerate the lexicographically least description of a length n string in $O(n)$ attempts. In contrast to this positive result for strings, we find that, in any Kolmogorov numbering, no enumerator of nontrivial size can generate a list containing the minimal index of a given partial-computable function. One cannot even achieve a laconic enumerator for *nearly*-minimal indices of partial-computable functions.

1   Short list approximations for minimal programs

No effective algorithm exists which computes shortest descriptions for strings, let alone lexicographically least descriptions. Such an algorithm would contradict the well-known fact that Kolmogorov complexity is not computable [11]. This paper investigates the extent to which one can effectively enumerate a "short" list of candidate indices which includes the lexicographically minimal program for a given string or a function.

Definition 1. An *enumerator* is an algorithm which takes an integer input and, over time, enumerates a list of integers. For an enumerator f, we let $f(e)$ denote the set of all elements which f eventually enumerates on input e.

Enumerators with non-trivial list sizes (i.e., of size much smaller than the length of the string x) fail to list-approximate Kolmogorov complexity.

Indeed any enumerator f such that $f(x)$ always contains the Kolmogorov complexity of x must, for all but finitely many n, for some string x of length n, include in the list $f(x)$ at least a fixed fraction of the lengths below $n + O(1)$ [4]. One might expect a similar result for enumerators whose enumerations always include the minimal index for a desired string — that is, one might expect the enumerators to enumerate all but a constant fraction of indices with length at most n. However in Theorem 3 below we show that for every universal machine for Kolmogorov complexity, there exists an enumerator f such that for all x, $|f(x)| = O(|x|)$ and f contains the minimal program for x. In contrast, we show that enumerators with short lists (of sublinear size) fail to list minimal indices for functions and that even enumerators containing nearly-minimal indices have large list sizes.

Prior investigations on short list-approximations of minimal indices for strings and functions have focused on computable functions. Bauwens, Makhlin, Vereshchagin, and Zimand [2] proved the optimal result that for any universal machine one can compute a quadratic-length list containing a description for a given string which is no more than $O(1)$ bits longer than that string's minimal description length. Teutsch [14] showed that one can do the same thing in polynomial-time if one relaxes the size of the list-approximation from quadratic to polynomial-length; see [18] for an alternative construction and a slightly shorter list. Bauwens and Zimand [3] showed that a randomized procedure can even achieve a linear-length list which, with high probability, contains a minimal description of the given string which is within $O(\log n)$ bits of optimal. Most recently, Vereshchagin [17] solved a problem posed in a preliminary version of [15] by showing that short computable list-approximations of minimal indices for functions do not exist. See [16] for a survey of related results.

We now introduce the notation and key definitions for this manuscript. A *numbering* φ is a partial-computable function $\langle e, x \rangle \mapsto \varphi_e(x)$. We say φ is a *Gödel* numbering if for any further numbering ψ, there exists a computable *translator* function t such that $\varphi_{t(e)} = \psi_e$. If in addition t satisfies $t(e) \leq c \cdot e + c$ for some constant c (depending on ψ), then φ is called a *Kolmogorov* numbering, and we call such a computable, linearly-bounded t a *Kolmogorov* translator from ψ to φ. Similar to universal machines for Kolmogorov complexity, which we define below, Kolmogorov numberings admit incoming translations which produce at most $O(1)$-bits increase in program size.

Kolmogorov himself introduced the notion of Kolmogorov numberings under the name "asymptotically optimal" [9]. Schnorr [12] later shortened this to "optimal numberings" and proved the following fundamental result.

Schnorr's Linear Isomorphism Theorem ([12])**.** *For every pair of Kolmogorov numberings φ and ψ, there exist a computable, bijective function t such that*

(I) *t and t^{-1} are both bounded by some linear function, and*

(II) *$\psi_{t(e)} = \varphi_e$ for all e.*

(It follows that also $\psi_e = \varphi_{t^{-1}(e)}$ for all e.)

We thank the anonymous referee who pointed us to the above valuable result which simplified and improved theorems from an earlier version of this manuscript.

For a Turing machine M, we let $C_M(x) = \min\{|p| \colon M(p) = x\}$ denote the *Kolmogorov complexity of x with respect to M*. A machine U is called *universal* if for any further machine M, $C_U(x) \leq C_M(x) + O(1)$. Universal machines exist [11].

Definition 2. For two partial-computable functions f and g, we say $f =^* g$ if f and g agree everywhere except on a finite set. For any numbering φ,

(I) let $\min_\varphi(e)$ denote the least index j such that $\varphi_j = \varphi_e$, and

(II) let $\min^*_\varphi(e)$ denote the least index j such that $\varphi_j =^* \varphi_e$.

Similarly, for any universal machine U,

(III) let $\min_U(x)$ denote the length lexicographically least program p such that $U(p) = x$, and

(IV) let $\min_U(x \mid y)$ denote the length lexicographically least program p such that $U(\langle p, y \rangle) = x$.

Let "p.c." stand for partial-computable, and let K denote the halting set for some fixed Gödel numbering. Let $\langle \cdot, \cdot \rangle$ denote a canonical, computable pairing function, and extend $\langle \cdot, \cdot \rangle$ to pairing of n-tuples by taking $\langle x_1, x_2, \ldots, x_n \rangle = \langle x_1, \langle x_2, \ldots, x_n \rangle \rangle$. Finally, let $|x| = \lceil \log(x + 1) \rceil$ be the size of the string x in binary. $\operatorname{dom} \eta$ denotes the set of values on which the partial function η is defined.

2 Strings

For any string x and any universal machine U, one can generate a list of length $|x|+O(1)$ containing a minimal-length program for x by enumerating the first program found for x at each length. We can even enumerate the length lexicographically least program.

Theorem 3. *For every universal machine U, there exists an enumerator f such that for all strings x, $|f(x)| = O(|x|)$ and $\min_U(x) \in f(x)$.*

Proof. Let U be a universal machine, and let a be a constant such that for each string x there exists a program p of size at most $|x| + a$ such that $U(p) = x$. We define a further machine M as follows. Let $T_{b,n}$ be the set of all x such that $U(q) = x$ for at least 2^b many different values q of length n.

Let $r_{b,n} = \lfloor 2^n/2^b \rfloor$, and note that $|T_{b,n}| \leq r_{b,n}$. On input $\langle b, k \rangle$, machine M first identifies the unique integer n such that $r_{b,n} \leq k < r_{b,n+1}$ and then sets $M(\langle b, k \rangle)$ to be the $(k+1-r_{b,n})$-th element enumerated into $T_{b,n}$. For each $x \in T_{b,n}$, there exists $k < r_{b,n+1}$ such that $M(\langle b, k \rangle) = x$, and so by universality of U, for such k,

$$C_U(x) \leq C_U(\langle b,k \rangle)+c \leq 2 \log b+\log(r_{b,n+1})+c+d \leq 2 \log b+(n+1)-b+c+d$$

for some constants c and d.

Fix b such that $b > 2 \log b+c+d+1$. Then for all $x \in T_{b,n}$, there exists a program of length less than n which computes x. We define $f(x)$ as follows: for each length $0 \leq n \leq |x| + a$, output the first 2^b U-programs (found in some algorithmic search) of length n which compute x. It follows from the definition of $T_{b,n}$ that either $f(x)$ enumerates all the U-programs of length n which compute x, or there exists a program of length less than n which computes x. By induction, $\min_U(x) \in f(x)$ and $|f(x)| \leq 2^b \cdot (|x| + 1 + a) = O(|x|)$. $\qquad\square$

The size of the list in Theorem 3 is optimal. This follows from Improved Gács's Theorem [1], which states that there exist infinitely many strings x such that $C_U(C_U(x) \mid x) \geq \log|x| - O(1)$. Indeed, as noted in [2], for any enumerator f which yields the complexity of its input, we have for infinitely many x

$$\log|x| - O(1) \leq C_U(C_U(x) \mid x) \leq \log|f(x)| + O(1),$$

hence $|f(x)| = \Omega(|x|)$. [4, Theorem 3.1] gives the same conclusion. As one can effectively deduce the value $C_U(x)$ from $\min_U(x)$, the optimality of Theorem 3 follows.

Remark 4. The enumeration algorithm in Theorem 3 simply enumerates (up to) constant many descriptions for x at each length below $|x| + O(1)$. Alternatively, one can verify the correctness of this algorithm by appealing to a theorem of Chaitin [5] [6, Lemma 3.4.2] which says that, for any universal machine, the number of minimal-length descriptions of any given string is bounded by some constant.

3   Functions

The following theorem and corollary extend a result of Vereshchagin [17], who showed that minimal indices for functions do not admit computable (short) list approximations. We show that not only are short, computable list approximations of minimal indices for functions impossible, but p.c. enumerators fail to exist as well (Theorem 5). Our approach is similar to Vereshchagin's, but since we now have enumerator list approximation instead of computable ones, we must "clear the board" and begin the game anew each time the enumerator enumerates a new element.

The function $h(|e|)$ in Theorem 5 below approximates the length of the minimal index for program e. One can take h to be the function $h(\ell) = \ell/c$ for any constant $c > 1$, or even the near identity function $h(\ell) = \ell - 4$; Theorem 5(III) is vacuous for $h(\ell) \geq \ell - 3$.

Theorem 5. *Suppose h is a non-decreasing, unbounded computable function such that $h(\ell) < \ell - 3$ for all $\ell > 4$. Then there exists a Kolmogorov numbering φ such that for any enumerator function f satisfying $|f(e)| \leq 2^{h(|e|)-1}$ for all e, there exist infinitely many e such that*

(I) $\min_\varphi(e) \notin f(e)$,

(II) $\min_\varphi^*(e) \notin f(e)$, *and*

(III) $2^{h(|e|)} \leq \min_\varphi^*(e) = \min_\varphi(e) < 8 \cdot 2^{h(|e|)}$.

Proof. Let h be as in the hypothesis and, using the fact that h is unbounded and computable, let w be an increasing computable function such that for all ℓ,

$$h(3w(\ell) + 2) + 3 < h(3w(\ell + 1) + 2).$$

Let $f_0, f_1, \ldots$ be an effective enumeration of all enumerator functions. We assume without loss of generality that $|f_n(e)| \leq 2^{h(|e|)-1}$ for all n and e since we can just ignore extra elements enumerated into $f_n(e)$. The overall goal when building φ is to ensure that, for each n, there exist infinitely

5

many pairs of strings (σ, τ) such that $\tau = \min_\varphi(\sigma) = \min_\varphi^*(\sigma)$ has length approximately $h(|\sigma|)$ and $\tau \notin f(\sigma)$.

We construct the Kolmogorov numbering φ as follows. We will use indices of length 0 mod 3 to ensure that φ is a Kolmogorov numbering and indices of lengths 1 mod 3 and 2 mod 3 for diagonalization against various f_n's. The given function w will prevent distinct parts of the diagonalization from interfering with each other.

Fix a Kolmogorov numbering ψ in which at least half of the indices of each length compute the everywhere undefined function. Let

$$\varphi_{001\xi} = \psi_\xi \quad \text{if } |\xi| \equiv 0 \mod 3,$$
$$\varphi_{01\xi} = \psi_\xi \quad \text{if } |\xi| \equiv 1 \mod 3,$$
$$\varphi_{1\xi} = \psi_\xi \quad \text{if } |\xi| \equiv 2 \mod 3.$$

For any z of the form $3w(\ell) + 2$, let $g(z) = h(z) + d$, where $d \in \{0, 1, 2\}$ is such that $g(z) \equiv 1 \mod 3$. The programs computed by φ-indices which are not explicitly defined either above or below will code the everywhere divergent function.

Fix an enumerator function f_n. Let q be the $(n+1)$-th prime number, and u be a positive integer. Set $p = 3w(q^u) + 2$. Primality ensures that $q_1^{u_1}$ and $q_2^{u_2}$ are distinct whenever either $q_1 \neq q_2$ or $u_1 \neq u_2$. Thus diagonalizations of distinct f_n's don't disrupt each other, and for each n, we can achieve the diagonalization for infinitely many e by using infinitely many values u.

By construction, $g(p)$ does not collide with any numbers of the form $3w(\ell) + 2$, and it follows from our assumption on h that $g(p) < p$. The φ-indices of lengths p and $g(p)$ (for different values of u) as defined above will be used to diagonalize against f_n. Let $\tau_1, \tau_2, \ldots, \tau_{2^{g(p)}}$ be the strings of length $g(p)$, and let $\sigma_1, \sigma_2, \ldots, \sigma_{2^p}$ be the strings of length p. For the rest of the proof, the indices i will range from 1 to $2^{g(p)}$ and the indices j will range from 1 to 2^p, representing the τ_i's and σ_j's respectively. Our construction will have the property that each $\varphi_{\tau_i}(x)$ and each $\varphi_{\sigma_j}(x)$ either converges to 1 or is undefined.

For every $s \geq 0$, stage s proceeds in two phases. First, we initialize the s-th stage of each function φ_{τ_i} and φ_{σ_j} by clearing all the information from previous stages as follows. If some φ_{τ_i} or some φ_{σ_j} converged on some input x prior to stage s, then at the beginning of stage s we set all of these functions to converge on input x. We add these convergences in a minimal way so that if none of these functions converged on the input x prior to stage s, then at the beginning of stage s still none of the functions converge on input x. Finally, we make the τ_i's pairwise infinitely different by setting

$\varphi_{\tau_i}(\langle p, i, s, t \rangle)$ equal to 1 for all t.

Next, the stage s of the construction splits into two threads. Let $f_{n,s}(\sigma_j)$ denote the set of elements enumerated into $f_n(\sigma_j)$ after s steps. The first thread searches for a j such that $f_n(\sigma_j)$ properly extends $f_{n,s}(\sigma_j)$. If such a j is found, then we terminate *stage s* and begin *stage $s+1$*. While this thread is still searching, the following is done.

    for $j = 1$ to $2^{g(p)-1}$ **do**

        Search for the least natural number k such that

            • $\tau_k \notin f_{n,s}(\sigma_j)$ and

            • k was not already chosen in a previous iteration of the for loop

    in stage s.

        Set φ_{σ_j} to follow φ_{τ_k}.

    end for

Let us verify that the search in the for loop above succeeds for all j. The set of unfollowed indices among $\tau_1, \dots \tau_{g(p)}$ at the beginning of the j-th iteration of the for loop above includes all indices which haven't been used in this stage's prior loop iterations, and there are at least

$$2^{g(p)} - (j-1) \geq 2^{g(p)-1} + 1 > |f_n(\sigma_j)|$$

distinct such indices. Therefore at least one of the τ_i's is not spoiled by any of the indices in $f_{n,s}(\sigma_j)$, and φ_{σ_j} can follow it.

As $|f_n(\sigma_j)|$ is finite for $1 \leq j \leq 2^{g(p)}$, there are only finitely many stages. Thus some stage s starts but does not end. By the following counting argument, there exists a j such that $1 \leq j \leq 2^{g(p)-1}$ and $\varphi_{\sigma_j} \neq^* \varphi_\xi$ for all ξ with $|\xi| < g(p)$. Any ξ with $|\xi| < g(p)$ and $|\xi| \not\equiv 0 \bmod 3$ is not an index for φ_{σ_j} by construction, and at least half of ξ with $|\xi| < g(p)$ and $|\xi| \equiv 0 \bmod 3$ are φ-indices for the everywhere divergent function by definition of ψ. Hence for this value of j, $|\min_\varphi(\sigma_j)| = |\min_\varphi^*(\sigma_j)| = g(p)$. It follows that

- $f_n(\sigma_j)$ enumerates neither $\min_\varphi^*(\sigma_j)$ nor $\min_\varphi(\sigma_j)$, and

- $\min_\varphi(\sigma_j) < 2^{g(p)+1} \leq 8^{h(|\sigma_j|)}$.

Since the above construction simultaneously diagonalizes against f_n for all n, for all $u > 0$, the theorem follows. $\qquad\square$

Theorem 5 carries over to all Kolmogorov numberings with a couple minor adjustments: in the hypothesis, change the exponent "$h(|e|) - 1$" to "$h(|e| - O(1)) - 1$," and in conclusion (III), change the exponent "$h(|e|)$" to "$h(|e| + O(1)) + O(1)$." The hidden constants here depend on the underlying Kolmogorov numbering; the next proof illustrates the role of these constants.

Corollary 6. *For any Kolmogorov numbering ψ, there exists a constant d such that for any enumerator g, $|g(e)| < e/d$ for all $e > 0$ implies $\min_\psi(e) \notin g(e)$ for infinitely many e. The conclusion also holds with "$\min_\psi(e)$" replaced with "$\min_\psi^*(e)$."*

Proof. Let φ be the special Kolmogorov numbering constructed in Theorem 5, and let ψ be an arbitrary Kolmogorov numbering. By Schnorr's Linear Isomorphism Theorem, there exists a bijective translator t from φ to ψ such that for some constant $c > 0$ and for all indices $e > 0$, $t(e)$ and $t^{-1}(e)$ are both bounded above by $c \cdot e$. Fix a function $h(\ell) = \ell - 4$ as in the hypothesis of Theorem 5. Then taking the enumerator f in Theorem 5 to be $t^{-1} \circ g \circ t$, we derive the existence of infinitely many e satisfying $\min_\varphi(e) \notin t^{-1}(g(t(e)))$. It follows that for all such e, $\min_\psi(t(e)) \notin g(t(e))$, and since t is injective, all of the infinitely many $t(e)$ give distinct values. Finally, note that this conclusion holds for any enumerator g satisfying $|g(e)| < e/32c$. Indeed such enumerators satisfy the necessary hypothesis of Theorem 5:

$$|f(e)| = \left|t^{-1}(g(t(e)))\right| = |g(t(e))| \leq t(e)/32c \leq e/32 \leq 2^{h(|e|)-1}. \qquad \square$$

Our proof of Corollary 6 crucially relies on the fact that the translators for the underlying Kolmogorov numbering are computable. In contrast, Theorem 3 holds for any universal machine. At present it is open whether some numbering, which admits linearly bounded but noncomputable translators from every other numbering, can have an enumerator with nontrivial output size which always contains a minimal index for its given input.

For numberings which do not satisfy the Kolmogorov property, sublogarithmic list approximations of minimal indices are possible for trivial reasons. Indeed one can code all the p.c. functions inside a Gödel numbering so sparsely that all smaller indices for non-empty functions can be enumerated while the enumerator list stays small, see [15]. Nevertheless, we can show that constant-size enumerators for minimal indices of functions do not exist. Theorem 7 below extends to enumerations a result in [15] which rules out constant-size computable list approximations for function minimal indices. Our proof of Theorem 7 makes use of the combinatorial result below.

Kummer Cardinality Theorem ([7],[10]). *Let B be a set of non-negative integers, and let k be a positive integer. Suppose that there exists an algorithm which, on any input $x_1, \ldots, x_k$ enumerates at most k integers among $\{0, 1, \ldots, k\}$ such that one of these integers equals $|B \cap \{x_1, \ldots, x_k\}|$. Then B is computable.*

The Kummer Cardinality Theorem has appeared previously in the literature in conjunction with applications for minimal indices [8, 13].

Theorem 7. *Let φ be a Gödel numbering, and let $k \geq 1$ be a constant. There is no enumerator f satisfying $|f(e)| \leq k$ and $|\min_\varphi(e)| \in f(e)$ for all e.*

Proof. Fix φ and k, and suppose such an f exists. Let $m_0, m_1, \ldots, m_k$ be such that for all $i < k$:

$$\varphi_{m_i} \subsetneq \varphi_{m_{i+1}} \quad \text{and} \quad |m_i| < |m_{i+1}| \quad \text{and} \quad \min_\varphi(m_i) = m_i.$$

Let $A_0, A_1, \ldots$ be an effective listing of all finite sets of cardinality k, and define a numbering ψ by $\psi_e = \bigcup_{\{i : |K \cap A_e| \geq i\}} \varphi_{m_i}$. Let g be a computable function such that $\varphi_{g(e)} = \psi_e$ for all e. Now it must be the case that $\min_\varphi(g(e)) \in \{m_0, \ldots, m_k\}$, as $|K \cap A_e| \leq k$ for all e. On the other hand, $f(g(e))$ enumerates at most k of these indices. Let h be a computable function satisfying $h(m_n) = n$ for all $n \leq k$. Then applying h to each of the outputs of $f(g(e))$ yields a subset of $\{0, 1, \ldots, k\}$ of size at most k containing $|K \cap A_e|$, whence by the Kummer Cardinality Theorem, K is computable. However, as K is not computable, this contradiction proves the theorem. $\square$

4 Nearly-minimal indices

If one relaxes the enumerator requirements so as only to include a small index rather than the absolute minimal one, then the size of the enumerator output can be reduced. Theorem 8 suggests a quantitative trade-off between the enumerator's output size $|f(e)|$ and the difference between $\min_\varphi(e)$ and φ_e's smallest index in $f(e)$.

Theorem 8. *Suppose h is a non-decreasing, unbounded computable function such that $h(\ell) < \ell - 3$ for all $\ell > 4$. Then there exists a Kolmogorov numbering φ such that for any enumerator function f satisfying $|f(e)| \leq 2^{h(|e|)-1}$, there exist infinitely many indices e such that*

(I) *every $j \in f(e)$ with $|j| < |e|$ satisfies $\varphi_j \neq \varphi_e$, and*

(II) *$\min_\varphi(e) < 8 \cdot 2^{h(|e|)}$.*

Proof. The initialization part of this proof is similar to the proof of Theorem 5, but the construction changes when we reach the **for** loop. Let h, w, ψ, g, q, u, p, σ_j, τ_i, and $f_{n,s}(\sigma_j)$ be as in Theorem 5. We remark that

an arbitrary Kolmogorov numbering ψ would suffice here, and that the τ_i's in this argument need only pairwise differ on single inputs rather than infinitely often (that is, we only define $\varphi_{\tau_i}(\langle p, i, s, t \rangle)$ for $t = 0$, rather than for all values of t as in Theorem 5).

As before, the construction splits into two threads. The first thread searches for a j such that $f_n(\sigma_j)$ properly extends $f_{n,s}(\sigma_j)$. If such a j is found, then we terminate *stage s* and begin *stage s + 1*. While this thread is still searching, the following is done with initial condition $t = s$. Let $\varphi_{\xi,t}$ denote φ_ξ computed within t steps, that is, $\varphi_{\xi,t}(x) = \varphi_\xi(x)$, if $\varphi_\xi(x)$ halts within t steps; otherwise $\varphi_{\xi,t}(x)$ is undefined.

for $j = 1$ to 2^p **do**:

1. Let k be the least natural number less than $2^{g(p)}$ such that every $\xi \in f_{n,s}(\sigma_j)$ of length less than p satisfies $\varphi_{\xi,t} \neq \varphi_{\tau_k}$ defined up to now. Such a k must exist because there are $2^{g(p)}$ distinct φ_{τ_i}'s while, by assumption, $|f_n(\sigma_j)| < 2^{g(p)-1}$. Define φ_{σ_j} to be φ_{τ_k} defined up to now.

2. Search for a $t' > t$ and $\xi \in f_{n,s}(\sigma_j)$ of length less than p such that $\varphi_{\xi,t'} = \varphi_{\tau_k}$ defined up to now. If found, proceed to the next iteration of the **for** loop with $t = t'$.

end for

Observe that if the **for** loop gets stuck in some iteration j, then the corresponding σ_j witnesses both of the conclusions of the theorem. By the first step of the loop, we have $\varphi_{\sigma_j} = \varphi_{\tau_k}$, whence by definition of τ_k and g,

$$\min_\varphi(\sigma_j) = \min_\varphi(\tau_k) < 2^{g(|\sigma_j|)+1} \leq 2^{h(|\sigma_j|)+3}.$$

This gives conclusion (II). Since the search in the second step of the loop failed to terminate, we obtain that for all $\xi \in f(\sigma_j)$ with $|\xi| < |\sigma_j|$, $\varphi_\xi \neq \varphi_{\sigma_j}$ which gives conclusion (I).

Finally we verify that the **for** loop does not terminate for some $j \leq 2^p$. Each iteration of the loop commits, at the end of its second step, at least one index of length less than p to forever follow at most one of the pairwise incomparable φ_{τ_k}. Since there are 2^p loop iterations and only $2^p - 1$ strings of length less than p, at least one of these iterations must not terminate. $\quad\square$

The following corollary presents an example of parameters achievable in an arbitrary Kolmogorov numbering.

Corollary 9. *Let ψ be a Kolmogorov numbering. Let g be an enumerator with $|g(e)| < \sqrt{e}$ for all $e > 0$. Then for infinitely many indices e, any*

10

index $y \in g(e)$ such that $\psi_y = \psi_e$ satisfies $|y| - |\min_\psi(e)| = \Omega(|e|)$. The hidden constant depends on the numbering ψ.

Proof. The present proof roughly follows the argument from Corollary 6. Let φ be the special Kolmogorov numbering from Theorem 8, let ψ be an arbitrary Kolmogorov numbering, and let g be an enumerator with $|g(e)| < \sqrt{e}$ for all e. By Schnorr's Linear Isomorphism Theorem, there exists a bijective translator t from φ to ψ such that for some constant $c > 0$ and for all indices $e > 0$, $t(e)$ and $t^{-1}(e)$ are both bounded above by $c \cdot e$.

Fix a function $h(\ell) = \log \sqrt{c} + \ell/2 + 1$ and an enumerator $f = t^{-1} \circ g \circ t$ so that f satisfies the necessary hypothesis of Theorem 8:

$$|f(e)| = \left|t^{-1}(g(t(e)))\right| = |g(t(e))| < \sqrt{c \cdot e} \le 2^{\left(\log\sqrt{c} + \frac{|e|}{2} + 1\right) - 1} = 2^{h(|e|) - 1}.$$

(Note that we may need minor adjustment of the above h, f to make sure that $h(\ell) < \ell - 3$ for small values of ℓ).

Now there exist infinitely many e such that the least index for φ_e in $f(e) = t^{-1}(g(t(e))$, if it exists, is at least $e/2$, and therefore the least index for $\psi_{t(e)}$ in $g(t(e))$, if it exists, is at least $t(e)/(2c^2)$. Moreover

$$\min_\psi(t(e)) < 8c \cdot 2^{h(|e|)} = 16c \cdot \sqrt{c \cdot e} \le 16c^2 \cdot \sqrt{t(e)}.$$

This gives a difference of $t(e)/(2c^2) - 16c^2 \cdot \sqrt{t(e)} = \Omega(e)$. Taking the difference between the logarithm of this minuend and subtrahend also yields the gap promised in the statement of the corollary in terms of program lengths. $\square$

We wonder whether or not the results in this section carry over to *-minimal indices.

Question 10. *Can we replace "=" with "=*" and "min" with "min*" in the statement of Corollary 9?*

References

[1] B. Bauwens and A. Shen. Complexity of complexity and strings with maximal plain and prefix Kolmogorov complexity. *The Journal of Symbolic Logic*, 79(2):620–632, June 2014.

[2] Bruno Bauwens, Anton Makhlin, Nikolay Vereshchagin, and Marius Zimand. Short lists with short programs in short time. In *IEEE 28th Conference on Computational Complexity (CCC)*, pages 98–108, 2013.

[3] Bruno Bauwens and Marius Zimand. Linear list-approximation for short programs (or the power of a few random bits). In *IEEE 29th Conference on Computational Complexity (CCC)*, pages 241–247, 2014.

[4] Richard Beigel, Harry Buhrman, Peter Fejer, Lance Fortnow, Piotr Grabowski, Luc Longpré, Andrej Muchnik, Frank Stephan, and Leen Torenvliet. Enumerations of the Kolmogorov function. *The Journal of Symbolic Logic*, 71(2):501–528, June 2006.

[5] Gregory J. Chaitin. Information-theoretic characterizations of recursive infinite strings. *Theoretical Computer Science*, 2(1):45 – 48, 1976.

[6] Rodney G. Downey and Denis R. Hirschfeldt. *Algorithmic randomness and complexity*. Theory and Applications of Computability. Springer, New York, 2010.

[7] Valentina Harizanov, Martin Kummer, and Jim Owings. Frequency computations and the cardinality theorem. *The Journal of Symbolic Logic*, 57(2):682–687, June 1992.

[8] Sanjay Jain, Frank Stephan, and Jason Teutsch. Index sets and universal numberings. *Journal of Computer and System Sciences*, 77(4):760–773, 2011.

[9] A. N. Kolmogorov. Three approaches to the quantitative definition of information. *Internat. J. Comput. Math.*, 2:157–168, 1968.

[10] Martin Kummer. A proof of Beigel's cardinality conjecture. *The Journal of Symbolic Logic*, 57(2):677–681, June 1992.

[11] Ming Li and Paul Vitányi. *An introduction to Kolmogorov complexity and its applications*. Texts in Computer Science. Springer, New York, third edition, 2008.

[12] C. P. Schnorr. Optimal enumerations and optimal Gödel numberings. *Mathematical Systems Theory*, 8(2):182–191, june 1974.

[13] Frank Stephan. The complexity of the set of nonrandom numbers. In Cristian S. Calude, editor, *Randomness and Complexity, from Leibnitz to Chaitin*, pages 217–230. World Scientific, Singapore, 2007.

[14] Jason Teutsch. Short lists for shortest descriptions in short time. *Computational Complexity*, 23(4):565–583, 2014.

[15] Jason Teutsch and Marius Zimand. On approximate decidability of minimal programs. *ACM Transactions on Computational Theory*, 7(4):17:1–17:16, August 2015.

[16] Jason Teutsch and Marius Zimand. A brief on short descriptions. *SIGACT News*, 47 (1), 42–67. March 2016.

[17] Nikolay Vereshchagin. Short lists with short programs for functions. *CoRR-arxiv*, abs/1409.5906, 2014.

[18] Marius Zimand. Short lists with short programs in short time—a short proof. In Arnold Beckmann, Erzsébet Csuhaj-Varjú, and Klaus Meer, editors, *Proceedings 10th CiE, Budapest, Hungary, Language, Life, Limits*, volume 8493 of *Lecture Notes in Computer Science*, pages 403–408. Springer International Publishing, 2014.