Theoretical Computer Science 700 (2017) 89–95

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Enumerations including laconic enumerators

Sanjay Jain [a,1], Jason Teutsch [b,*,2]

[a] National University of Singapore, Singapore
[b] TrueBit Foundation, Liechtenstein

ARTICLE INFO

Article history:
Received 30 March 2016
Received in revised form 28 June 2017
Accepted 1 August 2017
Available online 24 August 2017
Communicated by M. Golin

Keywords:
List approximations
Minimal programs
Kolmogorov complexity
Kolmogorov numberings
Recursion theory

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

* Corresponding author.
E-mail addresses: sanjay@comp.nus.edu.sg (S. Jain), jt@truebit.io (J. Teutsch).

1 Sanjay Jain is supported in part by NUS grant numbers R146-000-181-112, R252-000-534-112 and C252-000-087-001.
2 Jason Teutsch is supported in part by MOE grant number R-252-000-560-112.

http://dx.doi.org/10.1016/j.tcs.2017.08.001
0304-3975/

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 $t(e) \le c \cdot e + c$ 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 paper.

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