Theoretical Computer Science 412 (2011) 178–182

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Simplicity via provability for universal prefix-free Turing machines

Cristian S. Calude *

Department of Computer Science, University of Auckland, Private Bag 92019, Auckland, New Zealand

ARTICLE INFO

Keywords:
Universal Turing machine
Provability
Simplicity

ABSTRACT

Universality, provability and simplicity are key notions in computability theory. There are various criteria of simplicity for universal Turing machines. Probably the most popular one is to count the number of states/symbols. This criterion is more complex than it may appear at a first glance. In this note we propose three new criteria of simplicity for universal prefix-free Turing machines. These criteria refer to the possibility of proving various natural properties of such a machine (its universality, for example) in a formal theory, Peano arithmetic or Zermelo–Fraenkel set theory. In all cases some, but not all, machines are simple.

1. Introduction

Universality, provability and simplicity are key notions in computability theory which are relevant for both theoretical and applied computer science. The goal of this note is two-fold: (a) to propose three new criteria of simplicity for universal prefix-free Turing machines, (b) to show that every new criterion is non-trivial, i.e. in each case simple machines exist, but not all machines are simple. To achieve our goal we revisit a few old and recent results in Algorithmic Information Theory, and we prove a few new facts too. The accent is less on technical details, but more on the main concepts and results.

The paper is organised as follows. In the next section we discuss the idea of the smallest universal Turing machine. In the following three sections we introduce the universal prefix-free Turing machines, the concept of provability (in Peano arithmetic and Zermelo–Fraenkel set theory) and we review a few elementary facts in Algorithmic Information Theory. Each of the remaining three sections is devoted to a new criterion of simplicity.

2. The smallest universal Turing machine

Roughly speaking, a universal Turing machine is a Turing machine capable of simulating any other Turing machine. In Turing's words:

> *It can be shown that a single special machine of that type can be made to do the work of all. It could in fact be made to work as a model of any other machine. The special machine may be called the universal machine.*

The first universal Turing machine was constructed by Turing [32,33]. Shannon [29] studied the problem of finding the smallest possible universal Turing machine and showed that two symbols were sufficient, if enough states can be used. He also proved that "it is possible to exchange symbols for states and vice versa (within certain limits) without much change in the product". Notable universal Turing machines include the machines constructed by Minsky (7-state 4-symbol) [19], Rogozhin (4-state 6-symbol) [28], Neary–Woods (5-state 5-symbol) [22]. Herken's book [15] celebrates the first 50 years of universality.

* Tel.: +64 21 2411 454.
E-mail address: cristian@cs.auckland.ac.nz.
URL: http://www.cs.auckland.ac.nz/~cristian.

0304-3975/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2010.08.002

Weak forms of universality were proved by Watanabe (4-state 5-symbol) [34], Cook [12] for Wolfram's 2-state 5-symbol machine [35,36], Neary–Woods [20,21], and Smith [30] for Wolfram's 2-state 3-symbol machine[1].

3. Universal prefix-free Turing machines

A prefix-free Turing machine, shortly, machine, is a Turing machine whose domain is a prefix-free set. In what follows we will be concerned only with machines working on the binary alphabet $\{0, 1\}$. A *universal machine* U is a machine such that for every other machine C there exists a constant c (which depends upon U and C) such that for every program x there exists a program x' with $|x'| \leq |x| + c$ such that $U(x') = C(x)$. Universal machines can be effectively constructed. For example, given a computable enumeration of all machines $(C_i)_i$, the machine U defined by $U(0^i 1x) = C_i(x)$ is universal.[2] The domains of universal machines have interesting computational and coding properties; cf. [10,9].

4. Peano arithmetic and Zermelo–Fraenkel set theory

With the advent of formal proofs in mathematics, see for example the papers on formal proofs included in the December issue of the *Notices of AMS*, especially [14], problems related to provability in specific formal theories are becoming important. In what follows we will be dealing with provability in Peano arithmetic and Zermelo–Fraenkel set theory.

By $\mathcal{L}_A$ we denote the first-order language of arithmetic whose non-logical symbols consist of the constant symbols 0 and 1, the binary relation symbol $<$ and two binary function symbols $+$ (addition) and $\cdot$ (multiplication). Peano arithmetic, PA, is the first-order theory [16] given by a set of 15 axioms defining discretely ordered rings, together with induction axioms for each formula $\varphi(x, y_1, \ldots, y_n)$ in $\mathcal{L}_A$:

$$\forall \overline{y}(\varphi(0, \overline{y}) \wedge \forall x(\varphi(x, \overline{y}) \to \varphi(x + 1, \overline{y})) \to \forall x(\varphi(x, \overline{y})).$$

By $\mathrm{PA} \vdash \theta$ we mean "there is a proof in PA for θ".

PA is a first-order theory of arithmetic powerful enough to prove many important results in computability and complexity theories. For example, there are total computable functions for which PA cannot prove their totality, but PA can prove the totality of every primitive recursive function (and also of Ackermann's total computable, non-primitive recursive function); see [16].

Zermelo–Fraenkel set theory with the axiom of choice, ZFC, is the standard one-sorted first-order theory of sets; it is considered as the most common foundation of mathematics. In ZFC set membership is a primitive relation. By $\mathrm{ZFC} \vdash \theta$ we mean "there is a proof in ZFC for θ".

Our metatheory is ZFC. We fix a (relative) interpretation of PA in ZFC according to which each formula of $\mathcal{L}_A$ has a translation into a formula of ZFC. By abuse of language we shall use the phrase "sentence of arithmetic" to mean a sentence (a formula with no free variables) of ZFC that is the translation of some formula of PA.

5. Rudiments of Algorithmic Information Theory

The set of (bit) strings is denoted by Σ^*. If s is a string then $|s|$ denotes the length of s. All reals and rationals will be in the unit interval. A computably enumerable (shortly, c.e.) real number α is given by an increasing computable sequence of rationals converging to α. Equivalently, a c.e. real α is the limit of an increasing primitive recursive sequence of rationals. We will blur the distinction between the real α and the infinite base-two expansion of α, i.e. the infinite bit sequence $\alpha_1\alpha_2 \cdots \alpha_n \cdots (\alpha_n \in \{0, 1\})$ such that $\alpha = 0.\alpha_1\alpha_2 \cdots \alpha_n \cdots$. By $\alpha(n)$ we denote the string of length n, $\alpha_1\alpha_2 \cdots \alpha_n$.

One of the major problems in Algorithmic Information Theory is to define and study (algorithmically) random reals. To this aim one can use the prefix-complexity or constructive measure and dimension theories; remarkably, the class of "random reals" obtained with different approaches remains the same.

In what follows we will adopt the complexity-theoretic approach. Fix a universal machine U. The prefix-complexity induced by U is the function $H_U : \Sigma^* \to \mathbf{N}$ ($\mathbf{N}$ is the set of natural numbers) defined by the formula: $H_U(x) = \min\{|p| : U(p) = x\}$. One can prove that this complexity is optimal up to an additive constant in the class of all prefix-complexities $\{H_C : C \text{ is a machine}\}$.

A c.e. real α is *Chaitin-random* if there exists a constant c such that for all $n \geq 1$, $H_U(\alpha(n)) \geq n - c$. The above definition is invariant with respect to U. Every Chaitin-random real is non-computable, but the converse is not true. Chaitin-random reals abound: they have (constructive) Lebesgue measure one; cf. [1,5].

The standard example of c.e. Chaitin-random real is the halting probability of a universal machine U (Chaitin's Omega number)[3]:

$$\Omega_U = \sum_{U(x)<\infty} 2^{-|x|}.$$

[1] The critique by Pratt [25–27], the response in [24] and the forthcoming paper by Margenstern [18] show the subtlety of the notion of universality.

[2] See more in [1]. The above universal machine, called *prefix-universal* because universality is obtained by adjunction, is quite particular. There are universal machines which are not prefix-universal.

[3] $U(x) < \infty$ means "U is defined on x".

Each Omega number encodes information about halting programs in the most compact way. For example, the answers to the following $2^{n+1}-1$ questions "Does $U(x)$ halt?", for all programs $|x| \leq n$, is encoded in the first n digits of Ω_U. Compared with the real k_U whose ith bit is 1 iff the ith program halts on U, Ω_U achieves an exponential rate of compression. Is this important? For example, to solve the Riemann hypothesis one needs to calculate the first 2,741 bits of a natural Omega number [3,4].

The following result characterises the class of c.e. Chaitin-random reals:

Theorem 1 ([11,7,17]). *The set of c.e. Chaitin-random reals coincides with the set of all halting probabilities of all universal machines.*

The c.e. random reals have been intensively studied in recent years, with many results summarised in [1,13,23].

6. Universal machines simple for PA

We start with the simple question: Can PA certify the universality of a universal machine?

A universal machine U is called *simple for* PA if PA $\vdash$ "U is universal", i.e. PA can prove that a universal U, given by its full description, is indeed universal. For illustration, the results in this section will include full proofs.

As one might expect, there exist universal machines simple for PA:

Theorem 2 ([6]). *One can effectively construct a universal machine which is simple for* PA.

Proof. The set of all machines PA can prove to be prefix-free is c.e., so if $(C_i)_i$ is a computable enumeration of provably prefix-free machines, then the machine U_0 defined by $U_0(0^i 1x) = C_i(x)$ has the property specified in the theorem: PA $\vdash$ "U_0 is universal". □

However, not all universal machines are simple:

Theorem 3 ([6]). *One can effectively construct a universal machine which is not simple for* PA.

Proof. Let $(f_i)_i$ be a c.e. enumeration of all primitive recursive functions $f_i : \mathbf{N} \to \Sigma^*$ and $(C_i)_i$ a c.e. enumeration of all prefix-free machines. Fix a universal prefix-free machine U and consider the computable function $g : \mathbf{N} \to \mathbf{N}$ defined by:

$$C_{g(i)}(x) = \begin{cases} U(x), & \text{if for some } j > 0,\ \#\{f_i(1), f_i(2), \ldots, f_i(j)\} > |x|, \\ \infty, & \text{otherwise.} \end{cases}$$

For every i, $C_{g(i)}$ is a prefix-free universal machine iff $f_i(\mathbf{N})$ is infinite (if $f_i(\mathbf{N})$ is finite, then so is $C_{g(i)}$). Since the set of all indices of primitive recursive functions with infinite range is not c.e. it follows that there is some i such that PA cannot prove that $C_{g(i)}$ is universal. □

Both results above are true for plain universal machines too. The above proofs work for plain universal machines, but a simpler proof can be given for the negative result. In fact the above results work also for ε-universal machines ($\varepsilon \in (0, 1]$ is computable) studied in [8]. A machine U is ε-universal if for every machine T there exists a constant $c = c_{U,T}$ such that for each program $p \in \Sigma^*$, there exists a program $q \in \Sigma^*$ such that $U(q) = T(p)$ and $\varepsilon \cdot |q| \leq |p| + c$.

7. Universal machines simple for ZFC

Assume that the binary expansion of Ω_U is $0.\omega_1\omega_2\cdots$. For each digit ω_i we can consider two arithmetic sentences in ZFC, "$\omega_i = 0$", "$\omega_i = 1$". How many sentences of the above type can ZFC prove?

Theorem 4 ([11]). *Assume that* ZFC *is arithmetically sound (that is, each sentence of arithmetic proved by* ZFC *is true). Then, for every universal machine U,* ZFC *can determine the value of only finitely many bits of the binary expansion of Ω_U, and one can calculate a bound on the number of bits of Ω_U which* ZFC *can determine.*[4]

Actually, we can precisely describe the"moment" ZFC fails to prove any bit of Ω_U:

Theorem 5 ([2]). *Assume that* ZFC *is arithmetically sound. Let $i \geq 1$ and consider the c.e. random real $\alpha = 0.1^{i-1}0\alpha_{i+1}\cdots$ Then, we can effectively construct a universal machine U (depending upon* ZFC *and α) such that* PA *proves the universality of U,* ZFC *can determine at most i initial bits of Ω_U and $\alpha = \Omega_U$.*

In other words, the moment the first 0 appears (and this is always the case because α is random) ZFC cannot prove anything about the values of the remaining bits.

By taking $\alpha < 1/2$ we get Solovay's most "opaque" universal machine:[5]

Theorem 6 ([31]). *One can effectively construct a universal machine U such that* ZFC *(if arithmetically sound) cannot determine any bit of Ω_U.*

We say that a universal machine is *n-simple for* ZFC if ZFC can prove n digits and no more of the binary expansion of Ω_U. In view of Theorem 5, for every $n \geq 1$ there exists a universal machine which is *n-simple for* ZFC. By Theorem 6 there exists a universal machine which is not 1-simple for ZFC.

---

[4] This means that ZFC can prove only finitely many sentences of the form "$\omega_i = 0$", "$\omega_i = 1$" and one can calculate a natural N such that no sentence of the above type with $i \geq N$ can be proved in ZFC.

[5] Theorem 6 was obtained before Theorem 5.

8. Universal machines PA-simple for randomness

We first express Chaitin randomness in PA. A c.e. real α is *provably Chaitin-random* if there exists a universal machine simple for PA and a constant c such that PA $\vdash$ "$\forall n(H_U(\alpha(n)) \geq n - c)$".

In this context it is natural to ask the question: Which universal machines U "reveal" to PA that Ω_U is Chaitin-random?

Theorem 7 ([6]). *The halting probability of a universal machine simple for* PA *is provably Chaitin-random.*

In fact, Theorem 1 can be proved in PA:

Theorem 8 ([6]). *The set of c.e. provably Chaitin-random reals coincides with the set of all halting probabilities of all universal machines simple for* PA.

Based on Theorem 7 we define another (seemingly more general) notion of randomness in PA. A c.e. real is *provably random* (in PA) if there is a universal machine simple for PA and PA $\vdash$ "$\Omega_U = \alpha$".

Theorem 9 ([6]). *A c.e. real is provably random iff it is provably Chaitin-random.*

In contrast with the case of finite random strings where ZFC (hence PA) cannot prove the randomness of more than finitely many strings, for c.e. reals we have:

Theorem 10 ([6]). *Every c.e. random real is provably random.*

We say that a universal machine U is PA-*simple for randomness* if PA $\vdash$ "Ω_U is random".

Theorem 7 says that every universal machine simple for PA is PA-simple for randomness. It is an *open problem* whether the converse implication is true. In view of Theorem 10 we get:

Corollary 11. *For every c.e. random real α there exists a* PA*-simple for randomness universal machine U_0 such that $\alpha = \Omega_{U_0}$.*

A stronger version of Theorem 2 is true:

Theorem 12 ([6]). *There exists a universal machine which is not* PA*-simple for randomness (hence not simple for* PA*).*

Recall that a c.e. real α is *provably Chaitin ε-random* if there exists universal machine simple for PA and a constant c such that PA $\vdash$ "$\forall n(H_U(\alpha(n)) \geq \varepsilon \cdot n - c)$". Theorem 1 holds true for ε-random reals: the set of c.e. ε-random reals coincides with the set of all halting probabilities of all ε-universal machines. As a consequence, the definition and results presented in this section extend for universal machines PA-simple for ε-randomness [8].

9. Conclusion

We have introduced three new criteria of simplicity for prefix-free universal machines based on their "openness" in revealing information to a formal system, PA or ZFC. The type of encoding is essential for these criteria. Primarily, the concepts and results in this note are relevant to computability theory; secondarily, the notions of simplicity proposed can be useful for automatic theorem proving, testing and verification. It would be interesting to "actually construct" the universal machines discussed in this paper.

Acknowledgements

I thank D. Woods whose invitation to CSP08 stimulated these thoughts and H. Zenil who helped me with recent references. I am indebted to the anonymous referees for their comments which substantially improved the presentation.

References

[1] C.S. Calude, Information and Randomness. An Algorithmic Perspective, 2nd edition, Revised and Extended, Springer Verlag, Berlin, 2002.
[2] C.S. Calude, Chaitin Ω numbers, Solovay machines and incompleteness, Theoretical Computer Science 284 (2002) 269–277.
[3] C.S. Calude, Elena Calude, Evaluating the complexity of mathematical problems. Part 1, Complex Systems 18 (2009) 267–285.
[4] C.S. Calude, Elena Calude, Evaluating the complexity of mathematical problems. Part 2, Complex Systems 18 (2010) 387–401.
[5] C.S. Calude, G.J. Chaitin, What is . . . a halting probability? Notices of the AMS 57 (2) (2010) 236–237.
[6] C.S. Calude, N.J. Hay, Every Computably enumerable random real is provably computably enumerable random, Logic Journal of the IGPL 17 (2009) 325–350.
[7] C.S. Calude, P. Hertling, B. Khoussainov, Y. Wang, Recursively enumerable reals and Chaitin Ω numbers, in: M. Morvan, C. Meinel, D. Krob (Eds.), Proceedings of the 15th Symposium on Theoretical Aspects of Computer Science, Paris, Springer-Verlag, Berlin, 1998, pp. 596–606. Full paper in Theoretical Computer Science 255 (2001) 125–149.
[8] C.S. Calude, N.J. Hay, F. Stephan, Representation of left-computable ε-random reals, CDMTCS Report 365, 2009, 11 pp.
[9] C.S. Calude, A. Nies, L. Staiger, F. Stephan, Universal recursively enumerable sets of strings, in: M. Ito, M. Toyama (Eds.), Developments in Language Theory, DLT'08, in: Lectures Notes in Comput. Sci., vol. 5257, Springer-Verlag, Berlin, 2008, pp. 170–182.

[10] C.S. Calude, L. Staiger, On universal computably enumerable prefix codes, Mathematical Structures in Computer Science 19 (2009) 45–57.
[11] G.J. Chaitin, A theory of program size formally identical to information theory, Journal of the Association for Computing Machinery 22 (1975) 329–340.
[12] M. Cook, Universality in elementary cellular automata, Complex Systems 15 (1) (2004) 1–40.
[13] R. Downey, D. Hirschfeldt, Algorithmic Randomness and Complexity, Springer, Heidelberg, 2010.
[14] J. Harrison, Formal proof—theory and practice, Notices of the American Mathematical Society (December) (2008) 1395–1406.
[15] R. Herken, The Universal Turing Machine: A Half-Century Survey, Oxford University Press, Oxford, 1992.
[16] R. Kaye, Models of Peano Arithmetic, Oxford Press, Oxford, 1991.
[17] A. Kučera, T.A. Slaman, Randomness and recursive enumerability, SIAM Journal on Computing 31 (1) (2001) 199–211.
[18] M. Margenstern, Turing machines with two letters and two states, Complex Systems (in press).
[19] M. Minsky, Size and structure of universal Turing machines using Tag systems, in: Recursive Function Theory, Proc. Symp. Pure Mathematics, vol. 5, AMS, Providence, RI, 1962, pp. 229–238.
[20] T. Neary, D. Woods, Four small universal Turing machines, in: J. Durand-Lose, M. Margenstern (Eds.), Machines, Computations and Universality 2007, in: LNCS, vol. 4664, Springer, 2007, pp. 242–254. Full paper in Fundamenta Informaticae 91 (1) (2009) 123–144.
[21] D. Woods, T. Neary, Small semi-weakly universal Turing machines, in: J. Durand-Lose, M. Margenstern (Eds.), Machines, Computations and Universality 2007, in: LNCS, vol. 4664, Springer, 2007, pp. 303–315. Full paper in Fundamenta Informaticae 91 (1) (2009) 179–195.
[22] D. Woods, T. Neary, The complexity of small universal Turing machines, in: S.B. Cooper, B. Loewe, A. Sorbi (Eds.), Computability in Europe 2007, in: LNCS, vol. 4497, CIE, Springer, 2007, pp. 791–798.
[23] A. Nies, Computability and Randomness, Oxford University Press, 2009.
[24] NKS Forum, http://forum.wolframscience.com/showthread.php?s=&threadid=1472.
[25] V. Pratt, Simple turing machines, universality, encodings, etc., http://cs.nyu.edu/pipermail/fom/2007-October/012156.html.
[26] V. Pratt, Definition of universal Turing machine, http://cs.nyu.edu/pipermail/fom/2007-October/012148.html.
[27] V. Pratt, Complexity of (universal) Turing machines, http://www.cs.nyu.edu/pipermail/fom/2008-April/012828.html.
[28] Y. Rogozhin, A universal Turing machine with 22 states and 2 symbols, Romanian Journal of Information Science and Technology 1 (3) (1998) 259–265.
[29] C. Shannon, A universal Turing machine with two internal states, in: Automata Studies, Princeton, Princeton University Press, NJ, 1956, pp. 157–165.
[30] A. Smith, Wolfram's 2,3 Turing machine is universal, Complex Systems (in press).
[31] R.M. Solovay, A version of Ω for which *ZFC* can not predict a single bit, in: C.S. Calude, G. Păun (Eds.), Finite Versus Infinite. Contributions to an Eternal Dilemma, Springer-Verlag, London, 2000, pp. 323–334.
[32] A. Turing, On computable numbers, with an application to the Entscheidungsproblem, Proceedings of the London Mathematical Society 42 (2) (1936) 230–265.
[33] A. Turing, On computable numbers, with an application to the Entscheidungsproblem: a correction, Proceedings of the London Mathematical Society 2 (43) (1937) 544–546.
[34] S. Watanabe, 4-symbol 5-state universal Turing machine, Information Processing Society of Japan Magazine 13 (9) (1972) 588–592.
[35] S. Wolfram, A new kind of science, Wolfram Research, 2002, 706–714.
[36] Wolfram 2,3 Turing Machine, http://demonstrations.wolfram.com/TheWolfram23TuringMachine/.