Doctopic: Theory of natural computing ARTICLE IN PRESS TCS:11059

Theoretical Computer Science ••• (••••) •••–•••

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Random numbers as probabilities of machine behavior ☆

George Barmpalias [a,b,*], Douglas Cenzer [c], Christopher P. Porter [d]

[a] *State Key Lab of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, China*
[b] *School of Mathematics, Statistics and Operations Research, Victoria University of Wellington, New Zealand*
[c] *Department of Mathematics, University of Florida, Gainesville, FL 32611, United States*
[d] *Department of Mathematics and Computer Science, Drake University, Des Moines, IA 50311, United States*

ARTICLE INFO

Article history:
Received 21 May 2016
Received in revised form 13 January 2017
Accepted 1 February 2017
Available online xxxx
Communicated by C.S. Calude

Keywords:
Random oracles
Probability
Machine
Random numbers

ABSTRACT

A fruitful way of obtaining meaningful, possibly concrete, algorithmically random numbers is to consider a potential behavior of a Turing machine and its probability with respect to a measure (or semi-measure) on the input space of binary codes. In this work we obtain characterizations of the algorithmically random reals in higher randomness classes, as probabilities of certain events that can happen when an oracle universal machine runs probabilistically on a random oracle. Moreover we apply our analysis to several machine models, including oracle Turing machines, prefix-free machines, and models for infinite online computation. We find that in many cases the arithmetical complexity of a property is directly reflected in the strength of the algorithmic randomness of the probability with which it occurs, on any given universal machine. On the other hand, we point to many examples where this does not happen and the probability is a number whose algorithmic randomness is not the maximum possible (with respect to its arithmetical complexity). Finally we find that, unlike the halting probability of a universal machine, the probabilities of more complex properties like totality, cofinality, computability or completeness do not necessarily have the same Turing degree when they are defined with respect to different universal machines.

1. Introduction

In this work we examine the probabilities of certain outcomes of a universal Turing machine that runs on random input from the point of view of algorithmic complexity. The input of a Turing machine can be a finite binary string that is written on the input tape, or even an infinite binary stream whose bits are either written on the input tape at the start of the computation or are provided upon demand as the computation progresses. Such computations can be interpreted in terms of probabilistic machines and randomized computation or as oracle computations with a random oracle. The study of probabilistic Turing machines goes back to [19], where it was shown that if a randomized machine computes a function f with positive probability, then f is computable. Similarly, if a randomized machine produces an enumeration of a set W

☆ This research was partially supported by NSF DMS-1362273. Barmpalias was also supported by the 1000 Talents Program for Young Scholars from the Chinese Government, grant no. D1101130, the Chinese Academy of Sciences (CAS) and the Institute of Software of the CAS. We thank Veronica Becher and the referees for many corrections and suggestions that improved the presentation of this article.

* Corresponding author.
E-mail addresses: barmpalias@gmail.com (G. Barmpalias), cenzer@math.ufl.edu (D. Cenzer), cp@cpporter.com (C.P. Porter).
URLs: http://barmpalias.net (G. Barmpalias), http://people.clas.ufl.edu/cenzer (D. Cenzer), http://cpporter.com (C.P. Porter).

http://dx.doi.org/10.1016/j.tcs.2017.02.001

with positive probability, then W is computably enumerable, i.e. W can be enumerated by a deterministic machine without any oracle.

One way to introduce algorithmic randomness to the study of randomized computation is to consider what a machine can do given an algorithmically random oracle of a certain strength. This line of research has produced a large body of work and a sub-discipline in the area between algorithmic randomness and computability; see [4,10] for a presentation and a bibliography. A common theme in this topic is that algorithmically random oracles of sufficient strength cannot compute useful sets or functions. For example, oracles that are random relative to the halting problem do not compute any complete extension of Peano Arithmetic (Stephan [32]).

In this paper we apply the theory of algorithmic randomness to randomized computation in a different way. Given a property of the outcome of a randomized computation, we consider the probability with which this property occurs on a given universal machine. Then we consider the algorithmic randomness of this probability as a real number in $[0, 1]$. It turns out that in this way we can characterize the real numbers that are probabilities of certain key properties (like halting, totality, computability, etc.), as the algorithmically random numbers of certain well known classes in algorithmic information theory.

1.1. Previous work on the topic and outline of our results in historical context

One can trace this methodology back to Chaitin [12] where it was shown that the probability that a universal self-delimiting machine halts is a Martin-Löf random real. Becher, Daicz, and Chaitin [3] applied this idea to a model for infinite computations which was introduced earlier in Chaitin [13] and exhibited a probability which is 2-random, i.e. Martin-Löf random relative to the halting problem. Becher and Chaitin [1] studied a related model for infinite computations in order to exhibit a probability which is 3-random, i.e. Martin-Löf random relative to 2-quantifier sentences in arithmetic. A subsequent series of papers by Becher and Grigorieff [7–9] as well as Becher, Figueira, Grigorieff and Miller [5], followed this idea with the main aim of exhibiting algorithmically random numbers that are concrete, in the sense that they can be expressed as probabilities that a universal machine will have a certain outcome when it is run on a random input. Sureson [33] continues this line of research, showing that many natural open sets related to the universal machine have random probabilities of different algorithmic strength, while his arguments appeal to completeness phenomena as opposed to machine arguments.

A crucial development in the study of halting probabilities was the cumulative work of Solovay [31], Calude, Hertling, Khoussainov, and Wang [14], and Kučera and Slaman [21]. In these papers it was established that a real number is the halting probability of a universal self-delimiting machine if and only if it is Martin-Löf random and has a computably enumerable (in short, c.e.) left Dedekind cut. Moreover, as we explain in the following, this analysis shows that the same characterization remains true for most commonly used types of Turing machines and is not specific to the self-delimiting model which is sometimes used in algorithmic information theory. A similar approach was followed in [2] in order to provide a characterization of the *universality probability* of a self-delimiting machine, a notion that was introduced in [34].

In the present paper we take the aforementioned work as a starting point and develop a methodology for characterizing algorithmically random numbers as probabilities of certain outcomes of a universal machine like totality, computability, and co-finiteness. There are three main ways that our work stands out from previous attempts on this topic. We provide

(1) characterizations of algorithmically random numbers as probabilities;
(2) results that hold uniformly for several types of Turing machine models;
(3) examples of probabilities which are *not* as random as their arithmetical complexity would suggest;
(4) demonstrations that the probability of properties that are not definable with one quantifier is not Turing degree-invariant with respect to different universal machines.

We have a few remarks for each of these clauses. With respect to (1), the only such characterizations in the literature are the two that we mentioned, namely the halting probability and the universality probability. However, even in these cases the characterizations were proved for the specific model of self-delimiting machines although, as we will see, they apply in a much more general class of models. With respect to (2) we would like to note that all examples of random probabilities from the literature that we have encountered refer to self-delimiting models and there is a reason for this preference. Self-delimiting models correspond to prefix-free domains, which means that the probabilities considered turn out to be measures of open sets, which are easily represented and manipulated as sets of strings. In the present paper we free ourselves from this restriction, considering probabilities that are measures of classes that have higher Borel complexity, so they are not necessarily open or closed or even G_δ. Moreover, as we will see, this generality does not introduce an extra burden in our analysis. The reason for this pleasing fact can be traced to the theory of 1-random reals with left Dedekind cut (also known as left-c.e. reals) and the fact that the measures of Σ^0_n classes are the same as the measures of Σ^0_1 classes relative to $\mathbf{0}^{(n-1)}$, i.e. the Turing degree of the halting problem iterated $n-1$ times. As a result of this generality, we can look at classic properties of computability theory like totality, cofinality, completeness or computability, and characterize their probabilities in various standard models of Turing machines.

Clause (3) deserves a somewhat more elaborate remark. There is a direct contrast between the definability of a mathematical object and its algorithmic complexity in terms of effective randomness. Indeed, according to Martin-Löf [25] a sequence is random if it is not contained in a null set of a certain arithmetical complexity. So, for instance, since each