The Homogenous Capture of Random Strings

B. Natarajan
TR 85-672
April 1985

Department of Computer Science
Cornell University
Ithaca, NY 14853

THE HOMOGENEOUS CAPTURE OF RANDOM STRINGS

B.Natarajan

Department of Computer Science
Cornell University
Ithaca, NY 14853

ABSTRACT

It is well known that a set of strings that are random in the Kolmogorov sense is immune to all computable enumerations. In this paper, we discuss the generalization of this property to the computational resource hierarchies. We then introduce the notion of homogeneous capture of sets and show that sets of random strings are not homogeneously captured by any computable enumeration. Again, we discuss the extension of this property to the resource hierarchies. Finally, we discuss the relationship between the notion of homogeneous capture and the traditional concept of randomness.

INTRODUCTION

The Kolmogorov complexity K(x) of a string x is the length of the shortest program that computes it [1]. If M_u is the universal Turing machine,

$$K(x) = \min \{ |y| \; M_u(y) = x \}.$$

It is easily seen [1] that if M_u and M_v are two universal Turing machines, the Kolmogorov complexity of a string with respect to M_u will differ by no more than a constant value from that with respect to M_v, the constant depending only on M_u and M_v. With this in mind, we shall refer to the Kolmgorov complexity of a string without mentioning the universal machine on which it is measured. The set $K[n/2] = \{x | K(x) \leq |x|/2\}$ is the set of strings that can be compressed by a factor of two or more. In general, for monotonic and unbounded function $r(n) < n$, the set

$$K[r(n)] = \{x | K(x) \leq r(|x|)\}$$

is the set of strings that can be compressed by r(n) or more. Correspondingly, the set $\mathbf{K}_c[r(n)] = \Sigma^*\text{-}K[r(n)]$ is the set of r(n)-random (or just random) strings.

Recently , resource bounded versions of Kolmogorov complexity have been introduced . See [2] for a discussion. For resource R, string x is in the set KR[r(n),R(n)] if there exists a string of length r(|x|) or less from which x can be computed using R(|x|) or less of resource R.

$$KR[r(n),R(n)] = \{x \mid \exists y{:}|y| \leq r(|x|) \wedge M_u^{[R(|x|)]}(y) = x\}$$

where $M_u^{[R(|x|)]}$ is the universal Turing machine running on a resource bound of R(n). If time is the resource of interest, we have the set KT[r(n),T(n)] of those strings x that can be generated from strings of length r(|x|) or less in T(|x|) steps. Similarly, KS[r(n),S(n)] is the set of strings x that can be generated from strings of length r(|x|) or less in S(|x|) tape.

IMMUNITY OF RANDOM STRINGS IN THE RESOURCE HIERACHIES

An infinite set A is *immune* to a collection of sets C if no element of C is an infinite subset of A.

$$\forall c \in C : c \subseteq A \Rightarrow |c| < \infty.$$

It is well known that for any recursive, monotonic and unbounded $r(n) < n$, $K_c[r(n)]$ is immune to the computable sets. We now extend this property to the resource hierachies. We shall consider time and space, noting that similar results follow for other valid measures of computational complexity.

We define the class of languages TIME[T(n)] to be those languages that are accepted by T(n)-time bounded Turing machines. Correspondingly, we call a set A TIME[T(n)]-immune if no T(n)-time bounded Turing machine accepts an infinite subset of A. Consider Turing machines with a fixed number of tapes.

Theorem I. If $r(n) < n$ is time constructible and $r(n) > \log n$, then $KT_c[r(n),T(n)] = \Sigma^*\text{-}KT[r(n),T(n)]$ is immune to TIME[$(T_1(n)$] if

$$\lim_{n\to\infty} 2^{r(n)}T_1(n)/T(n) = 0.$$

Proof: Suppose there exists a $T_1(n)$ -time bounded Turing machine M that accepts an infinite subset of $KT_c[r(n),T(n)]$. We now construct Turing machine M' which behaves as follows. M' takes a binary integer y as input and runs M on strings of length y or more until it finds one accepted by M. M' then prints this string and halts. We now pick an input integer y_0 large enough so that

$$|y_0| + |M'| < r(y_0) \text{ and}$$

$$2^{r(y_0)}T_1(y_0) < T(y_0)$$

We also adjust y_0 if necessary, while maintaining the above inequalities so that M accepts some string of length y_0. Such a choice of y_0 exists by virtue of the asymptotic behaviour of the functions r ,T_1 and T.

Since there are at most $2^{r(y_0)}$ strings of length y_0 that are not in $KT_c[r(n),T(n)]$, we know that M' takes at most $2^{r(y_0)}T_1(y_0)$ time on input y_0. Let x be the string printed by M' on input y_0. Now $M_u(\langle y_0,M'\rangle)$ generates x in less than $T(|x|)$ steps. Also,

$$|M'| + |y_0| < r(|x|).$$

This implies that $x \notin KT_c[r(n),T(n)]$, a contradiction. Hence, M cannot enumerate an infinite subset of $KT_c[r(n),T(n)]$.

□

The corresponding result for space is stated by the following theorem. Here we are not limited to Turing machines with fixed number of tapes.

Theorem II. If $r(n)<n$ is monotonic, unbounded and space-constructible, then $KS_c[r(n),S(n)]$ is immune to $SPACE[S_1(n)]$ if

$$\lim_{n\to\infty} S_1(n)/S(n) = 0.$$

Proof: Similar to that of Theorem I.

□

HOMOGENEOUS CAPTURE

We know that sets of random strings are immune to the computable sets. Is it possible to approximate these sets by computable sets even if they cannot be captured exactly? In order to discuss this possibility, we have to make precise what we mean by an approximation. We could define a set A to be an approximation to a set B if at least a fixed fraction of its elements of any particular length are also in B. We will use a more general definition of approximation that we call homogeneous capture. Our motivation is the following. When we consider a Turing machine enumerator that approximates a set A, the natural question to ask is how many strings that are not in A are listed between two successive strings that are in A? Is there a bound on this number? Loosely speaking, we say an enumeration homogeneously captures A if there is a bound on the number of strings enumerated between any two successive strings in A. Before we make this precise, we need a few supporting definitions. We define a sequence of strings to be an *enumeration* if its elements are distinct. Further, if the sequence can be listed by some Turing machine, we call it a *computable* enumeration. We say S' is an *interval* of an enumeration $S = x_1, x_2, x_3, \ldots$, if there exist $x_j, x_k \in S$ such that $S' = x_j, x_{j+1}, \ldots, x_k$.

A set A is homogeneously captured by an enumeration S at the fraction α if S has at least as many elements as A and every sufficiently long subsequence of S contains at least a fraction α of elements from A. To be precise,

Definition: A set A is *homogeneously captured* by an enumeration S at the fraction $\alpha > 0$ if $|S| \geq |A|$ and for any interval S' of S

$$|S'| \geq 1/\alpha \Rightarrow \frac{|S' \cap A|}{|S|} \geq \alpha$$

In the following theorem we show that sets containing random strings resist enumerations that distribute the random strings uniformly. The set of all strings in an alphabet, Σ^*, contains all the random strings and is trivially

computable. Yet, no computable enumeration of Σ^* can distribute the random strings unformly to guarantee a minimum fraction of them in every sufficiently long sample of the enmumeration.

Theorem III. If $r(n)<n$ is recursive, monotonic and unbounded, $K_c[r(n)]$ is not homogeneously captured by any computable enumeration.

Proof: Let M be a Turing machine that enumerates S. We construct a machine M' from M that behaves as follows: On input j, M' generates a string y such that $r(|y|) \geqslant |M'| + 1/\alpha$. It then runs M to generate a sequence of $1/\alpha$ strings, all of which are longer than y. M' then prints the j^{th} string of this sequence and halts. Consider the family of strings

$\langle 1,M'\rangle,\langle 2,M'\rangle,\ldots,\langle 1/\alpha,M'\rangle$.

Consider also the family

$M_u(\langle 1,M'\rangle),M_u(\langle 2,M'\rangle),\ldots,M_u(\langle 1/\alpha,M'\rangle)$.

Since S homogeneously captures A at the fraction α, at least one of these strings must be in $K_c[r(n)]$. But each of these srings is generated from a short string. Hence none of them can be in $Kc[r(n)]$, a contradiction. Therefore either S is not computable or S does not homogeneously capture $K_c[r(n)]$.

□

HOMOGENEOUS CAPTURE IN THE RESOURCE HIERARCHIES

Theorem III can be readily extended to the resource hierarchies. We say a Turing machine enumerator operates in resource bound r(n) on resource R if it enumerates a sequence $x_1,x_2,x_3,\ldots$, and for all $i>0$, it uses no more than $r(|x_{i+1}|)$ units of resource R between x_i and x_{i+1}. Consider Turing machines with a fixed number of tapes.

Theorem IV. If $r(n)<n$ is monotonic, unbounded and time constructible, $KT_c[r(n),T(n)]$ is not homogeneously captured by any $T_1(n)$-time bounded Turing machine enumerator if

$$\lim_{n\to\infty} T_1(n)/T(n) = 0.$$

Proof: Straightforward extension of the proofs of Theorems I and III.

□

In the following we are not restricted to Turing machines with a fixed number of tapes.

Theorem IV. If $r(n)<n$ is monotonic, unbounded and space constructible, $KS_c[r(n),S(n)]$ is not homogeneously captured by any $S_1(n)$-space bounded Turing machine enumerator if

$$\lim_{n\to\infty} S_1(n)/S(n) = 0.$$

Proof: Straightforward extension of the proofs of Theorems II and III.

□

HOMOGENEOUS CAPTURE AND RANDOMNESS

The Kolmogorov complexity of a string is a measure of its randomness or 'hardness'. See [1,3]. for a discussion. What is the relevance of our theorems to randomness testing and the generation of random numbers? Qualitatively, Theorem III tells us that a random number generator cannot claim that "at least five out every hundred numbers it generates are random". There will be arbitrary and unbounded gaps between any two random numbers generated by it. Theorem V extends this to the space heirarchy. If we consider $KS_c[cn,S(n)]$, $0<c<1$, then the theorem says that no enumeration in an asymptotically smaller space can guarantee a minimum fraction of elements in $KS_c[cn,S(n)]$ in every sufficiently long sample. Loosely speaking, if we view the enumeration as random number generation, this implies that the generator will fail the Kolmogorov randomness test if the test is conducted

in an asymptotically larger space class. Theorem IV is the corresponding result for time.

CONCLUSION

In conclusion, we observe that arbitrary enumerations of even the simplest sets are not computable if they contain hard strings. The hardness has to fluctuate smoothly in the enumeration, with the hard strings occuring in lumps that cannot be dissipated. This property persists in the resource hierarchies as well. With reference to random number generation, we see that generators cannot satisfy randomness tests conducted in asymptotically larger resource classes.

REFERENCES

[1] Kolmogorov, A.N., Three Approaches for Defining the Concept of Information Quantity. *Prob. Inform. Trans.* 1, (1965), 1-7

[2] Hartmanis, J., Generalized Kolmogorov Complexity and the Structure of Feasible Computations. *Symposium on the Foundations of Computer Science*, 1983, 439-445.

[3] Martin Löf, P., The Definition of Random Sequences. *Inf. and Control.* 9, (1966), 602-619.