PREDICTION, COMPLEXITY AND RANDOMNESS

by

Giuseppe Trautteur

Department of Computer Science
University of British Columbia
Vancouver 8, British Columbia

May , 1972

Prediction, Complexity and Randomness

Giuseppe Trautteur

Our starting consideration is based upon a conjecture of Von Neumann [1] which we will refer to as VNHp. This conjecture seems to be one of the few perspicuous explorations of the notion of complexity besides the common statements about the feeling that some new phenomena will emerge given some sufficiently complex system. VNHp is bound both to factual issues and to epistemological issues. Von Neumann never formulated it explicitly but a fair rendering is as follows: there exists a certain level of complexity, $\bar{n}$, such that for objects of complexity less than $\bar{n}$ it is simpler to describe what they do (the behavior), than how they are made (the structure); whereas for objects of complexity larger than $\bar{n}$ the opposite occurs. Any object such that it is simpler to say how it is made rather than what it does will be said to have the Von Neumann property (VNP). Thus VNHp is simply the statement that all but the objects of complexity less than $\bar{n}$ have VNP. The context in which these ideas appeared was a discussion about the shape a logical theory of complex automata should have and the epistemological side of them is best expounded by an example by Von Neumann himself: "It is absolutely not clear <u>a priori</u> that there is a simpler description of what constitutes a visual analogy (what the visual brain does) than a description of the visual brain (how the visual brain is made)" (parenthetical notes are mine). On the other hand, the factual side of the discussion was concerned with the possibility of a self-reproducing-machine and the feeling was present, based on the assumption that a self reproducing machine should contain a description of itself, that the complexity of such a machine should be the critical value $\bar{n}$. However, the concepts appearing in VNHp were never formalized. The concepts and the new outlook which have been developed in the field of computational complexity may help to make a precise formulation of VNHp.

The universe in which we will be interested is one in which all the objects and structures are Turing machines of the Hartmanis & Stearns type [3], that is multi-tape, multi-head machines, with a non erasable one way output tape, which will be interpreted either as the means of communication with the outside or as representing the observable quantities associated with the machine. Although, as is well known, these machines are capable of computing the partial recursive functions, I will rather regard them as potentially infinite processes with the output tape as a record of the evolution of the system. Description will be identified with Gödel number in a fixed, fully effective Gödel numbering in the sense of Rogers [2]. This Gödel numbering, in a sense, will constitute the "language" in which "descriptions" are meaningful. The complexity will be the "size of machines" defined by Blum [4] as any finite-one recursive function ρ which assigns to any machine an integer which rates its complexity.

The choice of a particular ρ will depend on external circumstances of fact and will vary from field of application to field of application. However, it is required of ρ that there exists an effective procedure to tell which machines have a given complexity (ρ-value). Thus ρ explicates the phrase: how difficult it is to describe the structure of an object. It remains to define exactly the phrase: how difficult (complex) it is to say what an object does. I introduce here a notion of prediction. It seems obvious that if one is capable of saying what a machine does one must be capable of answering questions about the object ahead of time, or of some other resource, since otherwise it might have been more expedient to simply look at the object and our alleged knowledge would have appeared singularly useless. The definition of <u>prediction</u> is as follows: with each machine A_i there is naturally defined the function $T_i(n)$ which is the number of operations (or amount of some other resource) from start up to the printing of the n-th symbol on the output tape.
A_j predicts A_i ($A_i <_p A_j$) iff when A_i starts on some initial configuration α and A_j is started on some initial configuration β which encodes in a fixed manner i, α, m, d, the output tape of A_j, for some k, is identical between the k-th and the (k+d)-th symbol with the output tape of A_i between the m-th and the (m+d)-th symbol for all α, almost all m and all d <u>and</u> $T_j(k+d) < T_i(m+d)$. Of course if A_i stops we consider its tape as completed by a string of blank symbols as long as necessary. The <u>behavioral complexity</u> $\hat{\rho}(i)$ of A_i is defined as

the structural complexity, that is the ρ-value, of the smallest machine, if any, which predicts it, if this is smaller than $\rho(i)$, and $\rho(i)$ itself otherwise:

$$\hat{\rho}(i) = \min_{z} [[A_i <_p A_j \;\&\; \rho(j) = z < \rho(i)] \vee \rho(i) = z]$$

A machine has VNP if $\hat{\rho}(i) = \rho(i)$. This implies that all its predictors, if any, are structurally no simpler than A_i itself. It is immediate by padding in the enumeration of machines that if $<_p$ is not empty then there are infinitely many machines without VNP. On the other hand, using a theorem of Hartmanis & Stearns [3], it is easy to build for any machine another machine which predicts it, since prediction in this very restricted sense, only requires a linear speed-up. In this setup VNHp is not true because there are machines of arbitrary complexity without VNP, but something interesting happens instead. It is clear that all machines with the lowest possible ρ-value have VNP. A simple induction shows that there are infinitely many more and in fact not only all their predictors are no simpler but all machines in the transitive closure of $<_p$ are not. We therefore have this structure of intermingled machines, some with VNP, some without, which extends infinitely upwards.

A parallel phenomenon has been exploited to give epistemological substance to the notion of finite random string. As is now well known, Kolmogorov [5] and Chaitin [6] independently suggested, albeit following different formal developments, that a random object might be one such as to be very difficult to predict. Kolmogorov especially pursued the goal, along ideas sprung from Von Mises' approach to probability, to put probability theory on an algorithmic base, via a definition of random string which satisfied both the formal requests of probability theory and the intuitive explicandum which lies behind. In particular, the finite random string which has a strong intuitive and practical appeal and which could not be situated anywhere in classical probability theory, was the natural approach. We will consider only these and try to show the generality and allusive, if not explicative, power of VNP. Both Kolmogorov and Chaitin suggest that a good candidate to the status of finite random string would be a string such that its inherent complexity is less than or about the same as the complexity of a device capable of predicting it. This is very similar to the VNP for machines with the following changes: the function ρ becomes the inherent complexity of the string and is usually taken as the length of the string itself; while the behavioral complexity $\hat{\rho}$ becomes the Kolmogorov' complexity $K_A(x) = \rho(\min_y[A(y) = x])$. Under the simple condition on ρ, which must rate also the complexity of pairs of strings and satisfy $\rho(x,y) < c_x + \rho(y)$ for all strings y, the main theorem of Kolmogorov assures us of the existence of a universal programming system such that the complexity computed with respect to that system is not much larger than the complexity computed with respect to any other algorithm: $K(x) \leq K_A(x) + c$. It is also an immediate consequence of the definitions that K has a simple upper bound $K(x) \leq \rho(x)+c$.

It is natural to say in this framework that a string has VNP if the ρ-complexity of all the algorithms which compute it fall within a fixed constant of the ρ-complexity of the string itself: $\rho(x) - c \leq K(x) \leq \rho(x) + c$. This coincides with the definition of finite random string given by Martin-Lof [7] who shows that these strings pass any statistical test of randomness, and when ρ is interpreted as the length of the string it turns out that for each n the vast majority of strings are random in this sense. In fact there are strong similarities between a size measure as used before and this measure of complexity for strings. Using a standard one-one numbering, with no loss of generality, according to Rogers [2] we have

$$\begin{tikzcd} \underset{\text{algorithms}}{A} \arrow[r, leftrightarrow, "\text{standard numbering}"'] & \underset{\text{indices}}{N} \arrow[r, "\rho", "\text{(finite-one)}"'] & \underset{\text{sizes}}{N} \end{tikzcd}$$

$$\begin{tikzcd} \underset{\text{strings}}{S} \arrow[r, leftrightarrow, "\text{1-1 numbering}"'] & \underset{\text{binary strings as indices}}{B} \arrow[r, "\text{length}"] & N \text{ sizes} \\ & B \times B \arrow[r, "\text{length}"] & N^2 \arrow[u, "\text{pairing function}"'] \end{tikzcd}$$

Thus apart from an ambiguity because an integer may be the size of both a string and a pair, the situation is the same. In practice since the work of Martin-Lof one has dispensed with the intermediate coding, assuming that a string is a description of itself and taking ρ as the length of the string. The various recursiveness requirements are easily met. However, the length, or the length of the code, does not seem to be the only measure of interest, because: (i). for such strings as proteins, capable of very refined folded structures it would certainly be convenient to assign different complexities, possibly not monotone with length, to strings of the same length; and (ii) it may also be convenient to assign complexities to objects not obviously unidimensional. In such cases the number of objects of given complexity may be any function and the previous result does not obtain. Therefore it seems not entirely useless to observe that nevertheless for any ρ satisfying the finite-one requirement there are in fact infinitely many strings with VNP. The proof is the same as the one for machines when the relation $<_p$ is replaced by the relation $<_k$ defined by $x <_k y$ iff $U(y) = x$ where U is the universal programming system associated with K.

Now this rather pervasive phenomenon has a certain allusive power which expands in various directions. a) Simon [8] attempted to illustrate the fact that most complex systems of interest have a hierarchical structure and this permits their analysis by much simpler means than the systems themselves. This may be interpreted by saying that in fact most systems of interest do not have VNP so that they can be predicted or explained away by simpler systems. b) In the case of machines, VNP implies that a machine with VNP can be explained with advantage in some fixed resource at the expense of machine complexity while in the case of strings it is affirmed that no algorithm, independently of the amount of resource used, has a complexity sensibly lower than that of the object itself. It is this stronger fact that proposes on epistemological grounds the strings with VNP as random strings. c) The VNP might be improved to the r-VNP where r is a recursive function and a machine has r-VNP if

$$r(\hat{\rho}(i)) \le \rho(i).$$

d) The prediction scheme could be put on a more general axiomatic basis which should describe formally all the intermediate steps of a computation. e) An epistemological hope might be offered by the fact that even those systems with VNP which, being non-hierarchical, would have been considered by Simon not knowable, might possess approximations not necessarily simpler but without VNP. f) One might surmise that a well-known object without VNP might suddenly acquire it by being broken and thus explain why diagnostic problems are generally very difficult. g) Recently, Chaitin [9] proposed a new criterion for life which seems to fit in the VNP scheme. His main point is that a living organism is an object with an inherent complexity less than the sum of the complexities of its components. Elaborating a little upon Chartin's idea one might think of the decomposition of the object as a deduction of a formal grammar with the given object represented by the sentence symbol, every part of the object represented by some node and the elementary components of the object as terminal symbols. The branching at each node might be taken to represent the action of some operator putting together the higher node from more elementary ones. One may well think that the physical world may involve only a finite number of types of elementary objects and of ways of mutual interaction. Now a measure of inherent complexity is imposed on every object as a Blum size. But to every decomposition of an object one may assign a complexity value equal to the sum of the complexities of the parts in which the object can be exhaustively decomposed in accord with that decomposition. Thus, besides its own inherent complexity, to every object there belongs (with the exception of the elementary components) a set of - let them be called - decomposition complexities. Objects such that their inherent complexity is no larger than the lowest decomposition complexity would naturally be said to possess VNP and, according to Chaitin, life.

References

1) Von Neumann, J., Self Reproducing Automata, (Ed. A.W. Burks) Univ. of Illinois Press, (1966) pp. 47-56, p. 78.

2) Rogers, Hartley Jr., Gödel numberings of partial recursive functions, JSL 23, 331 (1958).

3) Hartmanis, J. & Stearns, R.E., On the computational complexity of algorithms, TAMS 117, 285 (1965).

4) Blum, M., On the size of machines, Info & Control 11, 257 (1967).

5) Kolmogorov, A.N., Three approaches to the quantitative definition of information, Int. J. of Comp. Math. 2, 157 (1968).

6) Chaitin, G.J., On the difficulty of computations, IEEE IT 16, 5 (1970).

7) Martin-Löf, P., Algorithms & randomness, Rev. Int. Stat. Inst. 37, 3 (1969).

8) Simon, H.A., The architecture of Complexity, Proc. Am. Phil. Soc. 106, 467 (1962).

9) Chaitin, G.J., To a mathematical definition of 'life' SIGACT News #4, 12 (1970) 106, 467 (1962).