Contents

Chapter 1

Algorithmic Chaos and the Incompressibility Method

By *Paul Vitanyi*

Many physical theories like chaos theory are fundamentally concerned with the conceptual tension between determinism and randomness. Kolmogorov complexity can express randomness in determinism and gives an approach to formulate chaotic behavior. As a technical tool to quantify the unpredictability of chaotic systems we use the Incompressibility Method. We introduce the method by examples: the distribution of prime numbers and largest clique size in random graphs.

1.1 Introduction

Ideally, physical theories are abstract representations – mathematical axiomatic theories for the underlying physical reality. This reality cannot be directly experienced, and is therefore unknown and even in principle unknowable. Instead, scientists postulate an informal description which is intuitively acceptable, and subsequently formulate one or more mathematical theories to describe the phenomena.

Deterministic Chaos: Many phenomena in physics (like the weather) satisfy well accepted deterministic equations. From initial data we can extrapolate and compute the next states of the system. Traditionally it was thought that increased precision of the initial data (measurement) and increased computing power would result in increasingly accurate extrapolation (prediction) for futures of lengths that linearly scaled inversely with the precision. But it has turned out that for many (that is, the chaotic) systems it scales not better than logarithmic inversely with the accuracy. In fact, it turns out that any long range prediction with any confidence better than what we would get by flipping a fair coin is practically impossible: this phenomenon is known as chaos (see [3] for an introduction). There are two, more or less related, causes for this:

Instability In certain deterministic systems, an arbitrary small error in initial conditions can exponentially increase during the subsequent evolution of the system, until it encompasses the full range of values achievable by the system. This phenomenon of instability of a computation is in fact well known in numerical analysis: computational procedures inverting ill-conditioned matrices (with determinant about zero) will introduce exponentially increasing errors.

Unpredictability Assume we deal with a system described by deterministic equations which can be finitely represented (see below). Even if fixed-length initial segments of the infinite binary representation of the real parameters describing past states of the system are perfectly known, and the computational procedure used is perfectly error free, for many such systems it will still be impossible to effectively predict (compute) any significantly long extrapolation of system states with any confidence higher than using a random coin flip. This is the core of chaotic phenomena: randomness in determinism.

REMARK 1. In the following we use the notion of "effective computation" in the well-known mathematical sense of "computability by Turing machine." Similarly, we use the notion of "(partial) recursive function" and "(partial) computable function" interchangeably. In recursion theory such functions are mappings from a subset of $\mathbb{N}$ into $\mathbb{N}$ (or into $\mathbb{Q}$, after composition by an explicit numbering of rationals). In the current context we may want to consider the extension to real numbers. A function $f : \mathbb{N} \to \mathbb{R}$ is *upper semi-computable* if there is a Turing machine T computing a total function $\phi : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}$ such that $\phi(x, t+1) \leq \phi(x, t)$ and $\lim_{t\to\infty} \phi(x, t) = f(x)$. This means that f can be computably approximated from above. If $-f$ is upper semi-computable, then f is called lower semi-computable. If f is both upper semi-computable and lower semi-computable, then we call f *computable* (or recursive, if the range is integer or rational). (We can similarly consider computable functions with a real domain: $f : \mathbb{R} \to \mathbb{R}$. This requires careful definitions and it turns out that computability implies continuity. But this sophistication is not needed in the current treatment.) The extension of the notion of computable functions to domain and range to vectors is straightforward. For details see any textbook on computability or [8].

REMARK 2. It is perhaps useful to stress that instability and unpredictability, although close companions, are not always the same. A trivial example of instability without unpredictability is a system that makes a first choice in an instable manner but afterwards sticks to that choice. (Such a system is equivalent, for instance, to a "dictatorial coin" that gives outcome 0 or 1 with equal probability when flipped the first time, but at every next flip will give the same outcome it gave the first time.) An example of unpredictability without instability is a function $f_r : \mathbb{N} \to \{0, 1\}$ defined by $f_r(n) = r_n$, with $r = r_1 r_2 \ldots$ an infinite binary sequence that is random in Martin-Löf's sense (see below) and hence unpredictable. (Here $n \in \mathbb{N}$ plays the rôle of time.)

Probability: Classical probability theory deals with randomness in the sense of *random variables*. The concept of random individual data cannot be expressed. Yet our intuition about the latter is very strong: an adversary claims to have a true random coin and invites us to bet on the outcome. The coin produces a hundred heads in a row. We say that the coin cannot have been fair. The adversary, however, appeals to probability theory which says that each sequence of outcomes of a hundred coin flips is equally likely, $1/2^{100}$, and one sequence had to come up. Probability theory gives us no basis to challenge an outcome *after* it has happened. We could only exclude unfairness in advance by putting a penalty side-bet on an outcome of 100 heads. But what about $1010\ldots$? What about an initial segment of the binary expansion of π?

Regular sequence

$$\Pr(00000000000000000000000000) = \frac{1}{2^{26}}$$

Regular sequence

$$\Pr(01000110110000010100111001) = \frac{1}{2^{26}}$$

Random sequence

$$\Pr(10010011011000111011010000) = \frac{1}{2^{26}}$$

The first sequence is regular, but what is the distinction of the second sequence and the third? The third sequence was generated by flipping a quarter. The second sequence is very regular: $0, 1, 00, 01, \ldots$ The third sequence will pass (pseudo) randomness tests.

In fact, classical probability theory cannot express the notion of *randomness of an individual sequence*. It can only express expectation of properties of the total set of sequences under some distribution.

This is analogous to the situation in physics above: *"how can an individual object be random?"* is as much a probability theory paradox as *"how can an individual sequence of states of a deterministic system be random?"* is a paradox of deterministic physical systems.

In probability theory the paradox has found a satisfactory resolution by combining notions of computability and information theory to express the complexity of a finite object. This complexity is the length of the shortest binary program from which the object can be effectively reconstructed. It may be called the *algorithmic information content* of the object. This quantity turns out to be an attribute of the object alone, and recursively invariant. It is the *Kolmogorov complexity* of the object. It turns out that this notion can be brought to bear on the physical riddle too, as we shall see below.

1.2 Kolmogorov Complexity

To make this paper self-contained we briefly review notions and properties required. For details and further properties see the textbook [8]. We identify the natural numbers $\mathbb{N}$ with the finite binary sequences as

$$(0,\epsilon),(1,0),(2,1),(3,00),(4,01),\ldots,$$

where ϵ is the empty sequence. The *length* $l(x)$ is the number of bits in the binary sequence x (for instance, $l(\epsilon)=0$). That defines also the "length" of the corresponding natural integer. If A is a set, then $|A|$ denotes the cardinality of A. Let $\langle\cdot,\cdot\rangle : \mathbb{N}\times\mathbb{N}\to\mathbb{N}$ denote a standard computable bijective "pairing" function. Throughout this paper, we will assume that $\langle x,y\rangle = 1^{l(x)}0xy$.

Define $\langle x,y,z\rangle$ by $\langle x,\langle y,z\rangle\rangle$.

We need some notions from the theory of algorithms, see [10]. Let $\phi_1,\phi_2,\ldots$ be a standard enumeration of the partial recursive functions. The (Kolmogorov) *complexity* of $x\in\mathbb{N}$, given y, is defined as

$$C(x|y)=\min\{l(\langle n,z\rangle) : \phi_n(\langle y,z\rangle)=x\}.$$

This means that $C(x|y)$ is the *minimal* number of bits in a description from which x can be effectively reconstructed, given y. The unconditional complexity is defined as $C(x)=C(x|\epsilon)$. These notions were originally introduced in [6].

An alternative definition is as follows. Let

$$C_\psi(x|y)=\min\{l(z) : \psi(\langle y,z\rangle)=x\} \tag{1.1}$$

be the conditional complexity of x given y with reference to a decoding function ψ. Then $C(x|y)=C_\psi(x|y)$ for a universal partial recursive function ψ that satisfies $\psi(\langle y,n,z\rangle)=\phi_n(\langle y,z\rangle)$.

We need the following properties. For each $x,y\in\mathbb{N}$ we have[1]

$$C(x|y)\le l(x)+O(1). \tag{1.2}$$

For each $y\in\mathbb{N}$ there is an $x\in\mathbb{N}$ of length n such that $C(x|y)\ge n$. In particular, we can set $y=\epsilon$. Such x's may be called *random*, since they are without regularities that can be used to compress the description: intuitively, the shortest effective description of such an integer x is x itself. In general, for each n and y, there are at least $2^n-2^{n-c}+1$ distinct x's of length n with

$$C(x|y)\ge n-c. \tag{1.3}$$

In some cases we want to encode x in *self-delimiting* (s.d.) form x', in order to be able to decompose $x'y$ into x and y. Then we will make use of the *prefix* complexity $K(x)$, introduced in [7], which denotes the length of the shortest *self-delimiting* description. To this end, we consider so called *prefix* Turing

[1] Throughout $O(1)$ (resp. $o(1)$) will denote a bounded quantity (resp. a quantity that converges to 0), whatever its sign, and $O(f(n))$ will mean $f(n)\times O(1)$ (resp. $o(f(n))$ will mean $f(n)\times o(1)$).

machines, which have only 0's and 1's on their input tape, and thus cannot detect the end of the input. Instead we define an input as that part of the input tape which the machine has read when it halts. When $x \neq y$ are two such input, we clearly have that x cannot be a prefix of y (that is, y cannot have the form xz), and hence the set of inputs forms what is called a *prefix code* or *prefix-free code*. We define $K(x|y), K(x)$ similarly to $C(x|y), C(x)$ above, but with reference to a universal prefix machine that first reads 1^n0 from the input tape and then simulates prefix machine n on the rest of the input.

Good upper bounds on the prefix complexity of x are obtained by iterating the simple rule that a self-delimiting description of the length of x followed by x itself is a s.d. description of x. For example, $x' = 1^{l(x)}0x$ and $x'' = 1^{l(l(x))}0l(x)x$ are both s.d. descriptions for x, and this shows that $K(x) \leq 2l(x) + O(1)$ and $K(x) \leq l(x) + 2l(l(x)) + O(1)$.

Similarly, we can encode x in a self-delimiting form of its shortest program $p(x)$ (of length $l(p(x)) = C(x)$) in $2C(x)+1$ bits. Iterating this scheme, we can encode x as a self-delimiting program of $C(x) + 2\log C(x) + 1$ bits[2], which shows that $K(x) \leq C(x) + 2\log C(x) + 1$, and so on.

1.2.1 The Incompressibility Method

The secret of the successful use of Kolmogorov complexity arguments as a proof technique lies in a simple fact: the overwhelming majority of strings have almost no computable regularities. We have called such a string "random." There is no shorter description of such a string than the literal description: it is incompressible.

Traditional proofs often involve all instances of a problem in order to conclude that some property holds for at least one instance. The proof would be more simple, if only that one instance could have been used in the first place. Unfortunately, that instance is hard or impossible to find, and the proof has to involve all the instances. In contrast, in a proof by the incompressibility method, we first choose a random (that is, incompressible) individual object that is known to exist (even though we cannot construct it). Then we show that if the assumed property did not hold, then this object could be compressed, and hence it would not be random. Let us give some simple examples.

Distribution of prime numbers. A prime number is a natural number that is not divisible by natural numbers other than itself and 1. In the nineteenth century, Chebychev showed that the number of primes less than n grows asymptotically like $n/\log n$.[3] Using the incompressibility method we cannot (yet) prove this statement precisely, but we can come remarkably close with a minimal amount of effort. We first prove that for infinitely many n, the number of

[2]Throughout log denotes the binary logarithm.

[3]More precisely, Chebychev showed (in 1850) that the quotient of $\pi(n)$ (the number of primes less than n) by $n/\log n$ is bounded by explicit positive constants. In fact, this ratio tends to $\log e$ when n tends to ∞: this is the "prime number theorem" proved (in 1896) by Hadamard and la Vallée Poussin. One can show that this theorem is equivalent to this statement: *if p_n denotes the n-th prime number, then $\frac{p_n \log e}{n \log n} \to 1$ when $n \to \infty$.* [Note added by the translator in the French edition.]

primes less than or equal to n is at least $\log n / \log\log n$. The proof method is as follows. For each n, we construct a description from which n can be effectively retrieved. This description will involve the primes less than n. For some n this description must be long, which will give the desired result.

Assume that $p_1, p_2, \ldots, p_m$ is the list of all the primes less than n. Then,

$$n = p_1^{e_1} p_2^{e_2} \cdots p_m^{e_m}$$

can be reconstructed from the vector of the exponents. Each exponent is at most $\log n$ and can be represented by $\log\log n$ bits. The description of n (given the maximal order of magnitude $\log n$ of the exponents) can be given in (approximately) $m \log\log n$ bits. But it can be shown that each n that is random (given $\log n$) cannot be described in fewer than $\log n$ bits, whence the result follows. Can we do better? This is slightly more complicated. Let $l(x)$ denote the length of the binary representation of x. We shall show that p_m (the m-th prime number) is $\leq m \log^2 m$. (One can show that this is equivalent to state that the number of primes less or equal to n is greater than $n/\log^2 n$.)

Firstly, any given integer n is completely determined by giving the index m of its greatest prime factor p_m, together with the (integral) quotient n/p_m. Thus we can describe n by $E(m)n/p_m$, where $E(m)$ is a prefix-free encoding of m. (The description of m needs to be self-delimiting or else we wouldn't know where the description of m ends, and where the description of n/p_m starts.) For random n, the length of this description, $l(E(m)) + \log n - \log p_m$, must exceed $\log n$. Therefore, $\log p_m < l(E(m))$. It is known (and straightforward from the earlier discussion) that we can set $l(E(m)) \leq \log m + 2\log\log m$. Hence, $p_m < m \log^2 m$: we have proven our claim.

Random graphs. The interpretation of strings as more complex combinatorial objects leads to a new set of properties and problems that have no direct counterpart in the "flatter" string world. Here we derive topological, combinatorial, and statistical properties of graphs with high Kolmogorov complexity. Every such graph possesses simultaneously all properties that hold with high probability for randomly generated graphs. They constitute "almost all graphs" and the derived properties a fortiori hold with probability that goes to 1 as the number of nodes grows unboundedly.

Each labeled graph $G = (V, E)$ on n nodes $V = \{1, 2, \ldots, n\}$ (at most one non-oriented edge for each pair of different nodes) can be represented (up to an automorphism) by a binary string $E(G)$ of length $\binom{n}{2}$: we simply assume a fixed ordering of the $\binom{n}{2}$ possible edges in an n-node graph, e.g. lexicographically, and let the ith bit in the string indicate presence (1) or absence (0) of the i'th edge. Conversely, each binary string of length $\binom{n}{2}$ encodes an n-node graph. Hence we can identify each such graph with its binary string representation.

We are going to prove that G does not contain a clique (complete graph) on more than $2\log n + 1 + o(1)$ nodes.

Let m be the number of nodes of the largest clique $\mathcal{K}$ in G. We try to compress $E(G)$, to an encoding $E'(G)$, as follows:

1. Prefix the list of nodes in $\mathcal{K}$ to $E(G)$, each node using $\lceil \log n \rceil$ bits[4],

[4] $\lceil x \rceil$ denotes the smallest integer that is greater than x.

adding $m\lceil \log n \rceil$ bits.

2. Delete all redundant bits from the $E(G)$ part, representing the edges between nodes in $\mathcal{K}$, saving $m(m-1)/2$ bits.

Then

$$l(E'(G)) = l(E(G)) + m\lceil \log n \rceil - \binom{m}{2}. \tag{1.4}$$

Let p be a program which, from n and $E'(G)$, reconstructs $E(G)$. Then,

$$C(E(G)|n,p) \leq l(E'(G)). \tag{1.5}$$

Since there are $2^{\binom{n}{2}}$ labeled graphs on n nodes and at most $2^{\binom{n}{2}} - 1$ binary descriptions of length less than $\binom{n}{2}$, we can choose a labeled graph G on n nodes that satisfies

$$C(E(G)|n,p) \geq \binom{n}{2} + o(\log n). \tag{1.6}$$

Equations 1.6, 1.4, and 1.5 are true only when $m \leq 2\log n + 1 + o(1)$.

In fact, the discerning reader will by now understand that while the information in the new prefix of $E'(G)$ is used by the program p to insert "1"s in the appropriate slots in the old suffix of $E'(G)$ to reconstruct the edges of the clique, using another program p' we show that the largest set of nodes with no pairwise edges is bounded by the same upper bound. Indeed, every easily (in $O(\log n)$ bits, given the labeled nodes) describable subgraph of G cannot have more than $2\log n + 1$ nodes. This includes virtually all properties we can conceivably be interested in. Moreover, the set of graphs G that satisfy (1.6) is very large: an easy counting argument shows that of the $2^{\binom{n}{2}}$ labeled graphs on n nodes, at least $(1-1/n)2^{\binom{n}{2}}$ graphs do so. That is, flipping a fair coin to determine presence or absence of the $\binom{n}{2}$ edges of a labeled graph on n nodes, with an overwhelming probability of $1-1/n$ we will flip a graph satisfying (1.6). Hence our conclusion about the maximal size of easily describable subgraphs holds with probability almost 1 (that is, probability $\geq 1 - 1/n$) for randomly generated graphs.

1.2.2 Random Sequences

We would like to call an infinite sequence $\omega \in \{0,1\}^\infty$ random if $C(\omega_{1:n}) \geq n + O(1)$ for all n (where $\omega_{1:n}$ denotes the sequence composed by the n first bits of ω). It turns out that such sequences do not exist. This remark led P. Martin-Löf [9] to create its celebrated theory of randomness. That ω is random in Martin-Löf's sense means, roughly, that it will pass *all* effective tests for randomness: both the tests which are known now and the ones which are as yet unknown [9].

Later it turned out, [1], that we can yet precisely define the Martin-Löf random sequences, but using prefix Kolmogorov complexity:

Theorem 1.2.1. *An infinite binary sequence ω is random in the sense of Martin-Löf iff there is an n_0 such that $K(\omega_{1:n}) \geq n$ for all $n > n_0$,*

Similar properties hold for high-complexity finite strings, although in a less absolute sense. For every finite set $S \subseteq \{0,1\}^*$ containing x we have $K(x|S) \leq \log |S| + O(1)$. Indeed, consider the self-delimiting code of x consisting of its $\lceil \log |S| \rceil$ bit long index of x in the lexicographical ordering of S. This code is called *data-to-model code*. The lack of typicality of x with respect to S is the amount by which $K(x|S)$ falls short of the length of the data-to-model code. The *randomness deficiency* of x in S is defined by

$$\delta(x|S) = \log |S| - K(x|S), \tag{1.7}$$

for $x \in S$, and ∞ otherwise. If $\delta(x|S)$ is small, then x may be considered as a *typical* member of S. There are no simple special properties that single it out from the majority of elements in S. This is not just terminology: if $\delta(x|S)$ is small, then x satisfies *all* properties of low Kolmogorov complexity that hold with high probability for the elements of S. For example: Consider strings x of length n and let $S = \{0,1\}^n$ be the set of such strings. Then $\delta(x|S) = n - K(x|n) + O(1)$. Let $\delta(n)$ be an appropriate function like $\log n$ or $\sqrt{n}$. Then, the following properties are a finitary analog of Martin-Löf randomness of infinite sequences, [8]:

(i) If P is a property satisfied by all x with $\delta(x|S) \leq \delta(n)$, then P holds with probability at least $1 - 1/2^{\delta(n)}$ for the elements of S.

(ii) Let P be any property that holds with probability at least $1 - 1/2^{\delta(n)}$ for the elements of S. Then, every such P holds simultaneously for every $x \in S$ with $\delta(x|S) \leq \delta(n) - K(P|n) + O(1)$.

Let us go one step further. The notion of randomness of infinite sequences and finite strings can only exist in the context of a probabilistic ensemble with respect to which they are a random element. In the above case, for the infinite sequences this ensemble is the set $\{0,1\}^\infty$ supplied with the uniform measure λ, also called the "coin-flip" measure, since $\lambda(\omega_1 \ldots \omega_n) = 1/2^n$ is the probability of producing the n-bit string $\omega_1 \ldots \omega_n$ with n flips of a fair coin. One can generalize the randomness approach to an arbitrary computable measure μ. This is a measure such that there is a Turing machine T such that for every n and $\varepsilon > 0$, on every input $\omega_1 \ldots \omega_n, \varepsilon$, the machine T halts with output r such that $|\mu(\omega_1 \ldots \omega_n) - r| \leq \varepsilon$. (For "Turing machine" we can also read "computer program" in a universal computer language like LISP or Java.) We can now talk about *μ-random sequences*, that is, sequences that satisfy every property (in the appropriate effective Martin-Löf sense) that holds with μ-probability 1 for the sequences in $\{0,1\}^\infty$. The following theorem is taken from [8]:

Theorem 1.2.2. *Let μ be a computable measure. An infinite binary sequence ω is μ-random in the sense of Martin-Löf iff there is an n_0 such that $K(\omega_{1:n}) \geq -\log \mu(\omega_{1:n})$ for all $n > n_0$.*

Note that for $\mu = \lambda$, the uniform distribution, we have $-\log \lambda(\omega_{1:n}) = n$, retrieving Theorem 1.2.1 again. We can extend the notion of μ-randomness to finite strings in the appropriate manner.

1.3 Algorithmic Chaos Theory

When physicists deal with a chaotic system, they believe that the whole thing is based on an underlying deterministic system but that the *fixed trajectory* is *unpredictable* on the basis of the observable states. Unfortunately, in the classical framework this cannot be expressed and therefore one has to use the kludge of an ensemble of states and trajectories, and to go through the rigmarole of probabilistic reasoning which is essentially besides the point. But using Kolmogorov complexity we can express directly the chaoticity of the system and the unpredictability properties of single trajectories, which is the intuition one wants to express. This is this idea that we shall advocate now.

For convenience assume that time is discrete: $\mathbb{N}$. In a deterministic system X the *state* of the system at time t is X_t. The *orbit* of the system is the sequence of subsequent states $X_0, X_1, X_2, \ldots$. For convenience we assume the states are elements of $\{0,1\}$. The definitions below are easily generalized. For each system, be it deterministic or random, we associate a measure μ with the space $\{0,1\}^\infty$ of orbits. That is, $\mu(x)$ is the probability that an orbit starts with $x \in \{0,1\}^*$.

Given an initial segment $X_{0:t}$ of the orbit we want to compute X_{t+1}. Even if it would not be possible to compute X_{t+1}, we would like to compute a prediction of it which does better than a random coin flip.

Definition 1.3.1. Let the set of orbits of a system X be $S = \{0,1\}^\infty$ with measure μ. Let ϕ be a partial recursive function and let $\omega \in S$. Define

$$\zeta_i = \begin{cases} 1 & \text{if } \phi(\omega_{1:i-1}) = \omega_i \\ 0 & \text{otherwise} \end{cases}$$

A system is *chaotic* if, for every computable function ϕ, we have

$$\lim_{t\to\infty} \frac{1}{t} \sum_{i=0}^{t-1} \zeta_i = \frac{1}{2},$$

with μ-probability 1.

Remark 3. For a chaotic system, no computable function ϕ predicts the outcomes of the system better than a fair coin flip. In this definition of chaoticity the essential requirement has been formulated as algorithmic unpredictability of the orbits. The instability properties of the system are expressed by the measure μ (as in Definition 1.3.1) the system induces. For example, let μ be the uniform measure (usually denoted as λ). An orbit like $\omega = \omega_1 \ldots \omega_n 11 \ldots$ is perfectly predictable after the first n bits. In fact, predictability by an appropriate computable function holds for all ω that are computable sequences, such as the binary expansions of the rationals but also transcendental numbers such as $\pi = 3.14\ldots$. However, for every $\omega \in S$, and every $\varepsilon > 0$, the ω' such that $|\omega - \omega'| \leq \varepsilon$, that is, the ω' in the ε-ball around ω, are unpredictable with uniform probability 1. This is because the set of Martin-Löf random sequences in the ε-ball has the same uniform measure as the set of all sequences in the ε-ball.

So the slightest random perturbation of an orbit will result in an unpredictable orbit, with probability 1.

It is not the case, however, that unpredictability implies instability. For example, if μ concentrates all its probability like $\mu(\omega) = 1$ for an $\omega = \omega_1\omega_2\ldots$ such that $K(\omega_{1:n}) \geq n$ for all n, that is, ω is Martin-Löf random, then the subsequent elements of ω are completely unpredictable given the preceding elements. Yet the orbit is completely stable, in fact, it is deterministic. The crux is of course that the orbit is a fixed sequence albeit a quite noncomputable one. We leave it to the reader to construct similar examples where the orbit is not completely fixed, not instable, but yet completely unpredictable.

REMARK 4. In chaos theory one typically considers deterministic systems X with states x from some domain R evolving in discrete steps according to $x_{n+1} = f(x_n)$, where the x_n's are real numbers, or vectors of real numbers, x_0 is given as initial value and f is a function mapping the current system state to the next system state. Physically it makes no sense to consider arbitrary real numbers since they require infinite precision – and that is not accessible to physical measurement. Moreover, no physical law or constant is known to hold with a precision of more than, say ten, decimals, and therefore the same will hold for the precision of the system evolution operator f. Hence, in analysis of the system behavior one replaces the actual values x_n by a finitary approximation represented by an equivalence class containing x_n. These equivalence classes represent the different states we can actually "distinguish", "observe" or "measure." For example, if the x_n are taken from the domain $[0, 1]$ then we can choose to divide the domain $[0, 1]$ into two equal parts, $R_0 = [0, \frac{1}{2})$ and $R_1 = [\frac{1}{2}, 1]$. Subsequently, we consider a system (X_n) defined by $X_n = i$ if $x_n \in R_i$ ($n = 0, 1, \ldots$ and $i \in \{0, 1\}$). Note that this defines both the initial value X_0 and the subsequent system states $X_1, X_2, \ldots$, from the original system X with initial value x_0. "Chaos" is defined for the derived system (X_n) that represents the evolution of "distinguishable" states. Now it becomes clear that for different initial states x_0 and x_0', even if they are taken from the same equivalence class, say R_0, so $X_0 = 0$, the orbits $X_0 = 0, X_1, \ldots$ may be quite different from X_1 onwards. If this happens so that the orbit $X_0 = 0, X_1, \ldots$ is in the appropriate sense unpredictable, even though x_0 and the evolution operator f are known, then we call the system "chaotic."

Our Definition 1.3.1 is based on the following: Let X be a system defined by $x_{n+1} = f(x_n)$ Suppose we randomly select the initial state x_0 from its domain R according to a measure ρ. That is, if $x_0 = x_{0,1}x_{0,2}, \ldots$ then the probability of selecting $x_{0,1} \ldots x_{0,r}$ is $\rho(x_{0,1} \ldots x_{0,r})$. (This still allows us to select a particular x_0 with probability 1 by concentrating all probability of ρ on x_0.) Considering the derived system of distinguishable states, the probability of the initial state $X_0 = i$ is $\rho(R_i)$ ($i \in \{0, 1\}$), but although the probability of the next state X_1 is determined completely by ρ and f, it is sensitive to change of either, and similarly for the states $X_2, X_3, \ldots$ after that. Nonetheless, f, ρ completely determine the probability of every initial segment of every orbit of distinguishable states. That is, for an initial segment $\omega_0 \ldots \omega_n$ we denote this probability as $\mu(\omega_0 \ldots \omega_n)$, and this defines the measure μ in Definition 1.3.1. Note that if

f, ρ are computable in an appropriate manner, then so is μ. For example, if $f(\omega_0\omega_1\ldots) = \omega'_0\omega'_1\ldots$ is such that there exists a computable monotonic increasing function g and a computable function h such that $h(\omega_0\ldots\omega_n) = \omega'_0\ldots\omega'_{g(n)}$, and moreover ρ is computable, then

$$\mu(X_0\ldots X_m) = \rho\{\omega_0\ldots\omega_n \ : g(n) = m \text{ and } h(\omega_0\ldots\omega_n) = \omega'_0\ldots\omega'_m \\ \text{and } \omega'_j \in R_i \text{ if and only if } X_j = i \ (j = 0,\ldots,m, i \in \{0,1\})\}.$$

The system is *uniformly instable* if for every ω and every $\varepsilon > 0$, the ε-ball $\{\omega' : |\omega - \omega'| \leq \varepsilon\}$ has a corresponding set of orbits $\{X'_0X'_1\ldots\}$ that is in an appropriate sense "dense" in the set of all possible orbits of the system. For example, by that set being equal to the set of possible orbits of the system.

The system is *uniformly unpredictable* if for every ω and every $\varepsilon > 0$, the ε-ball $\{\omega' : |\omega - \omega'| \leq \varepsilon\}$ produces a set of orbits in which the subset of Martin-Löf random sequences has full measure. (Here we mean Martin-Löf randomness with respect to the uniform distribution, and the "full measure" with respect to the induced measure μ on the set of orbits $X_0, X_1, \ldots$ of distinguishable states of the system.)

Clearly, systems can be both uniformly unpredictable and uniformly instable, but they can also be either one without being the other. *Chaoticity* as in Definition 1.3.1 can be the result of any of these three possibilities.

1.3.1 Doubling Map

A well-known example of such a chaotic system is the *doubling map*, see [5]. Consider the *deterministic* system D with initial state $x_0 = 0.\omega$ a real number in the interval $[0, 1]$ where $\omega \in S$ is the binary representation.

$$x_{n+1} = 2x_n \pmod 1 \tag{1.8}$$

where (mod 1) means droping the integer part. Thus, all iterates of x_0 under the transformation 1.8 lie in the unit interval $[0, 1]$. This interval corresponds to what is called the "phase space" in physics. We can partition this phase space into two cells, a left cell $R_0 = [0, \frac{1}{2})$ and a right cell $R_1 = [\frac{1}{2}, 1]$. Thus x_n lies in the left cell R_0 if and only if the nth digit of ω is 0.

One way to derive the doubling map is as follows: In chaos theory, [3], people have for years being studying the discrete-time logistic system L_α

$$y_{n+1} = \alpha y_n(1 - y_n)$$

which maps the unit interval upon itself when $0 \leq \alpha \leq 4$. When $\alpha = 4$, setting $y_n = \sin^2 \pi x_n$, we obtain:

$$x_{n+1} = 2x_n \pmod 1.$$

Theorem 1.3.2. *There are chaotic systems (like the doubling map D and the logistic map L_α for certain values of α, like alpha $= 4$), with μ in Definition 1.3.1 being the uniform distribution (the coin-flip measure λ, with $\lambda(x) = 2^{-l(x)}$: the probability of flipping the finite binary string x with a fair coin).*

Proof. We prove that D is a chaotic system. Since L_4 reduces to D by specialization, this shows that L_4 is chaotic as well. Assume ω is random. Then by Theorem 1.2.1,

$$C(\omega_{1:n}) > n - 2\log n + O(1). \tag{1.9}$$

Let ϕ be any partial recursive function. Construct ζ from ϕ and ω as in Definition 1.3.1.

Assume by way of contradiction that there is an $\varepsilon > 0$ such that

$$\left| \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \zeta_i - \frac{1}{2} \right| \geq \varepsilon.$$

Then, there is a $\delta > 0$ such that

$$\lim_{n\to\infty} \frac{C(\zeta_{1:n})}{n} \leq 1 - \delta. \tag{1.10}$$

We prove this as follows. The number of binary sequences of length n where the numbers of 0's and 1's differ by at least an εn is

$$N = 2 \cdot 2^n \sum_{m=(\frac{1}{2}+\varepsilon)n}^{n} b(n, m, \frac{1}{2}) \tag{1.11}$$

where $b(n, m, p)$ is the probability of m successes out of n trials in a $(p, 1-p)$ Bernoulli process: the binomial distribution. A general estimate of the tail probability of the binomial distribution, with m the number of successful outcomes in n experiments with probability of success $0 < p < 1$ and $q = 1 - p$, is given by Chernoff's bounds, [4, 2],

$$\Pr(|m - np| \geq \varepsilon n) \leq 2e^{-(\varepsilon n)^2/3n}. \tag{1.12}$$

Therefore, we can describe any element $\zeta_{1:n}$ concerned by giving n and εn in $2\log n + 4\log\log n$ bits self-delimiting descriptions, and pointing out the string concerned in a constrained ensemble of at most N elements in $\log N$ bits, where

$$N = 2^n \Pr(|m - np| \geq \varepsilon n) \leq 2^{n+1} e^{-(\varepsilon n)^2/3n}.$$

Therefore,

$$C(\zeta_{1:n}) \leq n - \varepsilon^2 n \log e + 2\log n + 4\log\log n + O(1).$$

That is, we can choose

$$\delta = \varepsilon^2 \log e - \frac{2\log n + 4\log\log n + O(1)}{n}.$$

Next, given ζ and ϕ we can reconstruct ω as follows:

`for` $i := 1, 2, \ldots$ `do:`
`if` $\phi(\omega_{1:i-1}) = a$ `and` $\zeta_i = 0$ `then` $\omega_i := \neg a$

`else` $\omega_i := a$.

Therefore,

$$C(\omega_{1:n}) \leq C(\zeta_{1:n}) + K(\phi) + O(1). \tag{1.13}$$

Now Equations 1.9, 1.10, 1.13 give the desired contradiction. By Theorem 1.2.1, one has

Claim 1. The set of ω's satisfying Equation 1.9 has uniform measure one.

In the definition of the doubling map we have already noted that: Starting from an initial state $x_0 = 0.\omega_1\omega_2\ldots$ the doubling map yields the trajectory $X_0, X_1, \ldots$ with $X_0 = i$ iff $x_0 \in R_i$ and $X_j = \omega_j$ for $j = 1, 2, \ldots$ and $i \in \{0, 1\}$. Therefore,

Claim 2. If we select the initial state x_0 with uniform probability from $[0, 1]$, then the induced measure on the resulting set of trajectories of distinguishable states $X_0, X_1, \ldots$ of the doubling map is the uniform measure.

Together, Claims 1, 2, prove the theorem. □

In [5] the argument is as follows. Assuming that the initial state is randomly drawn from $[0, 1)$ according to the uniform measure λ, we can use complexity arguments to show that the doubling map's observable orbit cannot be predicted better than a coin toss. Namely, with λ-probability 1 the drawn initial state will be a Martin-Löf random infinite sequence. Such sequences by definition cannot be effectively predicted better than a random coin toss, see [9].

But in this case we do not need to go to such trouble. The observed orbit essentially consists of the consecutive bits of the initial state. Selecting the initial state randomly from the uniform measure is isomorphic to flipping a fair coin to generate it. The approach we have taken above, however, allows us to treat chaoticity under nonuniform measures of selecting the initial condition. Moreover, we can think of initial states that are computable but pseudorandom versus prediction algorithms that are polynomially time bounded. Such extensions will be part of a future work.

From a practical viewpoint it may be argued that we really are not interested in infinite sequences: in practice the input will always be finite precision. Now an infinite sequence which is random may still have an arbitrary long finite initial segment which is completely regular. Therefore, we analyse the theory for finite precision inputs in the following section.

1.3.2 Chaos with Finite Precision Input

In the case of finite precision real inputs, the distinction between chaotic and non-chaotic systems can be precisely drawn, but it is necessarily a matter of degree. This occasions the following definition.

Definition 1.3.3. Let S, μ, ϕ, ω and ζ be as in Definition 1.3.1. A deterministic system with *input precision* n is (ε, δ)-*chaotic* if, for every computable function

ϕ, we have

$$\left| \frac{1}{n} \sum_{i=1}^{n} \zeta_i - \frac{1}{2} \right| \leq \varepsilon,$$

with μ-probability at least $1 - \delta$.

So systems are chaotic in the sense of Definition 1.3.1, like the doubling map above, iff they are $(0, 0)$-chaotic with precision ∞. The system is *probably approximately unpredictable*: a *pai*-chaotic system.

Theorem 1.3.4. *Systems D and L_α (for certain values of α, like $\alpha = 4$) above are $(\sqrt{(\delta(n) + O(1)) \ln 2/n}, 1/2^{\delta(n)})$-chaotic for every function δ such that $0 < \delta(n) < n$, with μ in Definition 1.3.3 being the uniform measure λ.*

Proof. We prove that D is (ε, δ)-chaotic. Since L_4 reduces to D, this implies that L_4 is (ε, δ)-chaotic as well. Assume that x is a binary string of length n with

$$C(x) \geq n - \delta(n). \tag{1.14}$$

Let ϕ be a polynomial time computable function, and define z by:

$$z_i = \begin{cases} 1 & \text{if } \phi(x_{1:i-1}) = x_i \\ 0 & \text{otherwise} \end{cases}$$

Then, x can be reconstructed from z and ϕ as before, and therefore:

$$C(x) \leq C(z) + K(\phi) + O(1).$$

By Equation 1.14 this means

$$C(z) \geq n - \delta(n) - K(\phi) + O(1). \tag{1.15}$$

We analyse the number of zeros and ones in z (we shall denote by $\#\,\text{ones}(z)$ the number of ones in z). Using Chernoff's bounds, Equation 1.12, with $p = q = \frac{1}{2}$, the number N of z's which have an excess of ones over zeros such that

$$|\#\,\text{ones}(z) - \frac{n}{2}| \geq \varepsilon n,$$

is such that:

$$N \leq 2^{n+1} e^{-(\varepsilon n)^2/n}.$$

Then, we can give an effective description of z by giving a description of ϕ, δ and z's index in the set of size N in this many bits

$$n - \varepsilon^2 n \log e + K(\phi) + K(\delta) + O(1). \tag{1.16}$$

From Equations 1.15, 1.16 we find

$$\varepsilon \leq \sqrt{\frac{\delta(n) + 2K(\phi) + K(\delta) + O(1)}{n \log e}}. \tag{1.17}$$

Making the simplifying assumption that $K(\phi), K(\delta) = O(1)$ this yields

$$|\#\,\text{ones}(z) - \frac{n}{2}| \leq \sqrt{(\delta(n) + O(1))n \ln 2}\,. \tag{1.18}$$

The number of binary strings x of length n with $C(x) < n - \delta(n)$ is at most $2^{n-\delta(n)} - 1$ (there are not more programs of length less than $n - \delta(n)$). Therefore, the uniform probability of a real number starting with an n-length initial segment x such that $C(x) \geq n - \delta(n)$ is given by:

$$\lambda\{\omega : C(\omega_{1:n}) \geq n - \delta(n)\} > 1 - \frac{1}{2^{\delta(n)}}. \tag{1.19}$$

Therefore, since we use the same doubling map D as in Theorem 1.3.2, the initial uniform distribution on the inputs induces a uniform distribution $\mu = \lambda$ on the corresponding set of trajectories of distinguishable states. Moreover, each such trajectory is the same bit sequence as the decimal expansion of the initial state. Then, by the almost equivalent two claims as in the proof of Theorem 1.3.2, the system D is (ε, δ) chaotic with $\varepsilon = \sqrt{(\delta(n) + O(1)) \ln 2/n}$ and $\delta = 1/2^{\delta(n)}$.

□

Acknowledgement

This paper is based on a talk by the author at the University of Waterloo, Canada, in 1991. Partially supported by EU through the NeuroColt II Working Group, the PASCAL Network of Excellence and the QAIP, RESQ Projects.

The idea of connecting primality and prefix code-word length, in the Incompressibility method proof of (a weaker version of) the Prime Number Theorem, is due to P. Berman, and the presented proof is due to J. Tromp.

Bibliography

[1] G. J. Chaitin, *A theory of program size formally identical to information theory*, J. Assoc. Comp. Mach. **22** (1975), p. 329-340.

[2] T. H. Cormen, C. E. Leiserson and R. L. Rivest, *Introduction to algorithms*, MIT Press, Cambridge, 1990.

[3] R. L. Devaney, *An Introduction to Chaos Dynamical Systems*, Addison-Wesley, 2nd Edition, 1989.

[4] P. Erdös and J. Spencer, *Probabilistic Methods in Combinatorics*, Academic Press, New York, 1974.

[5] J. Ford, *How random is a random coin toss?* Physics Today **36** (1983), April, p. 40-47.

[6] A. N. Kolmogorov, *Three approaches to the definition of the concept of 'quantity of information'*, Problems in Information Transmission **1** (1965), p. 1-7.

[7] L. A. Levin, *Laws of information conservation (non-growth) and aspects of the foundation of probability theory*, Problems in Information Transmission **10** (1974), p. 206-210.

[8] M. Li and P. M. B. Vitányi, *An Introduction to Kolmogorov Complexity and its Applications*, 2nd Edition, Springer-Verlag, New York, 1997.

[9] P. Martin-Löf, *On the definition of random sequences*, Information and Control **9** (1966), p. 602-619.

[10] H. J. Rogers, Jr., *Theory of Recursive Functions and Effective Computability*, McGraw-Hill, 1967.