Turing's Normal Numbers: Towards Randomness

Verónica Becher

Departamento de Computación, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Pabellón I, Ciudad Universitaria, (1428) Buenos Aires, Argentina
vbecher@dc.uba.ar

Abstract. In a manuscript entitled "A note on normal numbers" and written presumably in 1938 Alan Turing gave an algorithm that produces real numbers normal to every integer base. This proves, for the first time, the existence of computable normal numbers and it is the best solution to date to Borel's problem on giving examples of normal numbers. Furthermore, Turing's work is pioneering in the theory of randomness that emerged 30 years after. These achievements of Turing are largely unknown because his manuscript remained unpublished until its inclusion in his Collected Works in 1992. The present note highlights Turing's ideas for the construction of normal numbers. Turing's theorems are included with a reconstruction of the original proofs.

1 On the Problem of Giving Instances of Normal Numbers

The property of *normality* on real numbers, defined by Émile Borel in 1909, is a form of randomness. A real number is normal to a given integer base if its infinite expansion is seriously balanced: every block of digits of the same length must occur with the same limit frequency in the expansion of the number expressed in that base.[1] For example, if a number is normal to base two, each of the digits '0' and '1' occur in the limit, half of the times; each of the blocks '00', '01', '10' and '11' occur one fourth of the times, and so on. A real number that is normal to every integer base is called *absolutely normal*, or just *normal*. Borel proved that almost all real numbers are normal (that is, the set of normal numbers has Lebesgue measure 1), and he asked for an explicit example. Since then it has been easier to conjecture results on normality than to prove them. In particular, it remains unproved whether the fundamental mathematical constants such as π, $\sqrt{2}$ and e are normal to some integer base. Although its has been proved that there exist numbers that are normal to one base but not to another [9,26], no examples have been given. There are already many particular constructions of numbers

[1] An alternative characterization proves that a real number x is normal to a base b if, and only if, the sequence $(xb^n)_{n\geq 1}$ is uniformly distributed modulo one [6]. Also, a real number is normal to a base b if, and only if, its expansion is compressible by no information lossless finite automaton (injective transducer) [22,18,7].

S.B. Cooper, A. Dawar, and B. Löwe (Eds.): CiE 2012, LNCS 7318, pp. 35–45, 2012.

that are normal to a given base, but no explicit instance has been proved normal to two multiplicatively independent bases; see [6] for up to date references.

It is fair to say that Borel's question on providing an example of a normal number (normal to every integer base) is still unresolved because the few known instances are not completely satisfactory: it is desirable to show that a *known* irrational number is normal, or, at least, to exhibit the number explicitly. We would like an example with a simple mathematical definition and such that, in addition of normality, some extra properties are proved. Considering that *computability* is the acceptable notion of constructiveness since the 1930s, we would also like that the number be easily computable. Let us recall that, as defined by Turing [24], the *computable real numbers* are those whose expansion in some integer base can be generated by a mechanical (finitary) method, outputting each of its digits, one after the other.

There is no evident reason for the normal numbers to have a non-empty intersection with the computable numbers. A measure-theoretic argument is not enough to see that these two sets intersect: the set of normal numbers in the unit interval has Lebesgue measure one, but the computable numbers are just countable, hence they form a null set (Lebesgue measure 0). Indeed, there are computable normal numbers, and this result should be attributed to Alan Turing. His manuscript entitled *"A note on normal numbers"*, presumably written in 1938, presents the best answer to date to Borel's question: an algorithm that produces normal numbers. This early proof of existence of computable normal numbers remained largely unknown because Turing's manuscript was only published in 1997 in his Collected Works, edited by J.L.Britton [25]. The editorial notes say that the proof given by Turing is inadequate and speculate that the theorem could be false. In [1] we reconstructed and completed Turing's manuscript, trying to preserve his ideas as accurately as possible and correcting minor errors.

The very first examples of normal numbers were independently given by Henri Lebesgue and Waclaw Sierpiński[2] in 1917 [16,23]. They also lead to computable instances by giving a computable reformulation of the original constructions [2]. Together with Turing's algorithm these are the only known constructions of computable normal numbers. In his manuscript, Turing alerts the reader that the provided examples of normal numbers are not convenient and he explicitly says that one would like that the expansion of such numbers be actually exhibited. From his wording we suppose that he was aware of the problem that the n-th digit in the expansion of a number output by his algorthm is defined by exponentially many operations in n. Actually, a literal reading of Turing's algorithm yields that at most *simple-exponentially* many operations suffice. Our reconstruction worsens this amount to *double-exponentially* many, due to a modification we had to introduce in one expression that Turing wrote without a proof (see Section 2.2). A theorem of Strauss [27] asserts that normal numbers computable in simple exponential time do exits, but this existential result yields no specific instances.

[2] Both published their works in the same journal issue, but Lebesgue's dates back to 1909, immediately after Borel's question.

There are two other published constructions of normal numbers, one due to W.M.Schmidt in 1962 [26], the other to M.B.Levin in 1979 [17], but it is still unproved whether they yield computable numbers. Bugeaud in [5] demonstrated the existence of Liouville numbers that are normal. It is an open problem whether there are computable instances. Other non constructive examples of normal numbers follow from the theory of algorithmic randomness (recent reference books are [10,20]; for an overview see [11] in this volume). Since randomness implies normality, the particular real numbers that have been proved random are, therefore, normal. For instance, Chaitin's Omega numbers [8], the halting probabilities of optimal Turing machines with prefix-free domain. But random numbers are not computable, so Omega numbers are not the desired examples.[3]

2 Turing's Construction of Normal Numbers

In his manuscript Turing proves two theorems. Here we discuss the main ideas and include the proofs in accordance to our reconstruction in [1] of the original. We intend our curent presentation to be simpler and more readable. Theorem 1 is a computable version of Borel's fundamental theorem that establishes that almost all real numbers, in the sense of Lebesgue measure, are normal [3]. The theorem gives a construction of a set of real numbers as the limit of computably definable finite approximations. This set has arbitrarily large measure and consists only of normal numbers. This construction is valuable in its own right.

Turing's Theorem 1. *There is a computable function $c(k,n)$ of two integer variables with values consisting of finite sets of pairs of rational numbers such that, for each k and n, if $E_{c(k,n)} = (a_1,b_1)\cup(a_2,b_2)\cup...(a_m,b_m)$ denotes the finite union of the intervals whose rational endpoints are the pairs given by $c(k,n)$, then $E_{c(k,n)}$ is included in $E_{c(k,n-1)}$ and the measure of $E_{c(k,n)}$ is greater than $1-1/k$. And for each k, $E(k) = \bigcap_n E_{c(k,n)}$ has measure $1-1/k$ and consists entirely of normal numbers.*

In Theorem 2 Turing gives an algorithm to output the expansion of a normal number in base two. The proof relies on the construction in Theorem 1. The algorithm is a computable functional: it receives an integer value that acts as a parameter to control measure, and an infinite sequence ν in base two to be used as an oracle to possibly determine some digits of the output sequence. When ν is a computable sequence (Turing puts the sequence of all zeros), the algorithm yields a computable normal number. With this result Turing is the first one to prove the existence of computable normal numbers.

Turing's Theorem 2. *There is an algorithm that, given an integer k and an infinite sequence ν of zeros and ones, produces a normal number $\alpha(k,\nu)$ in the unit interval, expressed in base two, such that in order to write down the first n*

[3] The family of Omega numbers coincides with the family of random real numbers that can be approximated by a computable non-decreasing sequence of rationals [14].

digits of $\alpha(k,\nu)$ the algorithm requires at most the first n digits of ν. For a fixed k these numbers $\alpha(k,\nu)$ form a set of measure at least $1-2/k$.

The algorithm can be adapted to intercalate the bits of the input sequence ν at fixed positions of the output sequence. Thus, one obtains non-computable normal numbers in each Turing degree.

Notation. For an integer base $b \geq 2$, a *digit* in base b is an element in $\{0,...,b-1\}$, and a *block* in base b a finite sequence of digits in base b. $|u|$ is the length of a block u, and $u[i..i+r-1]$ is the inner block of r consecutive digits in a block u starting at position i, for $1 \leq i \leq |u|-r+1$. A block w *occurs* in a block u at position i if $u[i...i+|w|-1] = w$. The set of all blocks of length r in base b is denoted by $\{0,...,(b-1)\}^r$. For each real number x in the unit interval we consider the unique expansion in base b of the form $x = \sum_{i=1}^{\infty} a_i b^{-i}$, where the integers $0 \leq a_i < b$, and $a_i < b-1$ infinitely many times. This last condition over a_n is introduced to ensure a unique representation of every rational number. When the base b is fixed, we write $x[i..i+r-1]$ to denote the inner block of length r in the expansion of x in base b, starting at position i. We write μ to denote Lebesgue measure.

Turing uses the following definition of normality, given by Borel in [4] as a characterising property of normal numbers.

Definition 1 (Normality). *For a real number x and an integer base $b \geq 2$, the number of occurrences of a given block w in the first k digits of the expansion of x in base b is $S(x,b,w,k) = \#\{i : 1 \leq i \leq k-|w|+1 \text{ and } x[i..i+|w|-1] = w\}$. The number x is* normal to base b *if for every block w, $\lim_{k\to\infty} \frac{S(x,b,w,k)}{k} = b^{-|w|}$. If x is normal to every base $b \geq 2$ then we say x is* normal.

2.1 Turings's Theorem 1: A Construction via Finite Approximations

The main idea in Turing's Theorem 1 is the construction of a set of normal numbers of arbitrarily large measure, via finite approximations. This is done by pruning the unit interval by stages such that, at the end, one obtains the desired set consisting only of normal numbers. The construction is uniform on a parameter k, whose only purpose is to establish the measure of the constructed set $E(k)$ to be exactly $1-1/k$. At each stage n the construction is a finite set of intervals with rational endpoints determined by a computable function $c(k,n)$. At the initial stage 0, the set $E_{c(k,0)}$ is the whole unit interval. At stage n, the set $E_{c(k,n)}$ is the finite approximation to $E(k)$ that results from removing from $E_{c(k,n-1)}$ the points that are *not* candidates to be normal, according to the inspection of an initial segment of their expansions. At the end of this infinite process all rational numbers are discarded, because of their periodic structure. All irrational numbers with an unbalanced expansion are discarded. But also many normal numbers may be discarded, because their initial segments remain unbalanced for too long.

The construction covers all initial segment sizes, all bases, and all blocks by increasing computable functions of the stage n. And it has a decreasing

bound on the acceptable discrepancy between the actual number of blocks in the inspected initial segments and the perfect number of blocks expected by the property of normality. These functions (initial segment size, base, block length and discrepancy) are such that, at each stage n, the set of discarded numbers has a small measure. The set $E(k)$, obtained in the limit of the construction, is the countable intersection of the sets $E_{c(k,n)}$ and consists just of normal numbers.

The proof of Theorem 1 depends on a constructive version of the strong law of large numbers: for each base there are a few blocks with too many or too few occurrences of any given shorter block. The expected number of occurrences of a given *digit* in a block of length k is k/b plus or minus a small fraction of k. An upper bound for the number of blocks of length k having the expected occurrences of a given *digit* is proved in Hardy and Wright's book[4] [12], Theorem 148 (also in many books as [6,13,15]).

Definition 2. *The number of blocks of length k in base b where a given block of r digits occurs exactly i times is $p_{b,r}(k,i)$.*

In particular, the number of blocks of length k with exactly i occurrences of a given *digit* is $p_{b,1}(k,i) = \binom{k}{i}(b-1)^{k-i}$.

Lemma 1. *Fix a base $b \geq 2$ and a block length $k > 6b$. For every real number ε such that $6/k \leq \varepsilon \leq 1/b$,* $\sum_{i:\ |i-k/b| \geq \varepsilon k} p_{b,1}(k,i) \ < \ 2\ b^k e^{-b\varepsilon^2 k/6}$.

Turing extends this result to count occurrences of *blocks* instead of *digits*. Lemma 2 corresponds to our reconstruction in [1] where we give the full proof. The upper bound used by Turing in his manuscript is smaller but unproved.

Lemma 2. *Let base $b \geq 2$ and and let k and r be block lengths such that $k > r$. For every real number ε such that $6/\lfloor k/r \rfloor \leq \varepsilon \leq 1/b^r$,*

$$\sum_{i:\ |i-k/b^r| \geq \varepsilon k} p_{b,r}(k,i) < 2\ b^{k+2r-2} r\ e^{-b^r \varepsilon^2 k/6r}.$$

Lemma 2 provides a lower bound for the measure of the set of real numbers that are candidates to be normal based upon inspection of an initial segment of their expansion in finitely bases. In the following we define $A(\varepsilon, T, L, k)$ as the set of real numbers such that their initial segment of size k in each base up to T has a discrepancy of frequency below ε for each block of length up to L.

Definition 3. *For a real value ε and integer values T, L and k, let*

$$A(\varepsilon, T, L, k) = \bigcap_{2 \leq b \leq T} \ \bigcap_{1 \leq r \leq L} \ \bigcap_{w \in \{0,\ldots,b-1\}^r} \{x \in (0,1)\ :\ |S(x,b,w,k) - k/b^r| < \varepsilon k\}.$$

Observe that $A(\varepsilon, T, L, k)$ is a finite union of intervals with rational endpoints.

[4] Since the first edition of *Introduction to the Theory of Numbers* was in 1938 we suppose the material was taught by G.H.Hardy in King's College Cambridge at the time Turing was a student.

Proposition 1. *If* $6/\lfloor k/L\rfloor \le \varepsilon \le 1/T^L$, $\mu A(\varepsilon,T,L,k) \ge 1-2L\,T^{3L-1}e^{-\varepsilon^2 k/3L}$.

Proof. By Definition 3, the complement of $A(\varepsilon,T,L,k)$ in the unit interval is $\overline{A}(\varepsilon,T,L,k) = \bigcup_{2\le b\le T}\ \bigcup_{1\le r\le L}\ \bigcup_{w\in\{0,\ldots,b-1\}^r} \overline{B}(\varepsilon,b,w,k)$, where the set $\overline{B}(\varepsilon,b,w,k) = \{x\in(0,1) : |S(x,b,w,k) - k/b^r| \ge \varepsilon k\}$. Observe that if a number x belongs to $\overline{B}(\varepsilon,b,w,k)$ then so does each y such that $x[1..k] = y[1..k]$. Then, the interval $[0.x[1..k]000\ldots, 0.x[1..k](b-1)(b-1)(b-1)\ldots]$, which has measure b^{-k}, is included in $\overline{B}(\varepsilon,b,w,k)$. Recall that $p_{b,r}(k,i)$ (cf. Definition 2) is the number of different blocks of length k in which a given block of length r occurs exactly i times. Letting the block length $r = |w|$ we have $\mu\overline{B}(\varepsilon,b,w,k) \le b^{-k} \sum_{i:\ |i-k/b^r|\ge\varepsilon k} p_{b,r}(i,k)$. Applying Lemma 2, $\mu\overline{B}(\varepsilon,b,w,k) < 2\ b^{2r-2}r\ e^{-b^r\varepsilon^2 k/6r}$. Since $1\le r\le L$, $2r/L \le 2 \le b^r$. Then, $\varepsilon^2 k/3L \le b^r\varepsilon^2 k/6r$. This gives a uniform upper bound $\mu\overline{B}(\varepsilon,w,b,k) < 2\,T^{2L-2}L\,e^{-\varepsilon^2 k/3L}$ for all b,r,w such that $2\le b\le T$, $1\le r\le L$ and $w\in\{0,\ldots,b-1\}^r$. Thus, $\mu\overline{A}(\varepsilon,T,L,k) \ge \sum_{2\le b\le T}\sum_{1\le r\le L}\sum_{w\in\{0,\ldots,b-1\}^r} \mu\overline{B}(\varepsilon,b,w,k)$. In the third sum there are b^r many blocks w. Using $\sum_{2\le b\le T}\sum_{1\le r\le L} b^r = \sum_{2\le b\le T}\frac{b^{L+1}-1}{b-1} \le T^{L+1}$, conclude $\mu\overline{A}(\varepsilon,T,L,k) < 2L\,T^{3L-1}e^{-\varepsilon^2 k/3L}$. The proof is completed by taking the complement.

Turing defines the sets A_k as particular instances of the sets $A(\varepsilon,T,L,k)$ where ε, T and L are computable functions of the initial segment size k such that $\varepsilon(k)$ goes to 0 as k increases, and $T(k), L(k)$ are increasing in k. Turing chose the base $T(k)$ to grow sub-linearly in k, and the block length $L(k)$ to grow sub-logarithmically in k, which would yield the maximum discrepancy $\varepsilon(k)$ (according to the bound of Lemma 1). Other assignments are possible.

Definition 4. Let $A_k = A(\varepsilon,T,L,k)$ for $L = \sqrt{\ln k}/4$, $T = e^L$ and $\varepsilon = 1/T^L$.

Proposition 2. *There is* k_0 *such that for all* $k\ge k_0$, $\mu A_k \ge 1 - 1/k(k-1)$.

Proof. By Definition 4, $L = \sqrt{\ln k}/4$, $T = e^L$ and $\varepsilon = 1/T^L$. Assume $k\ge 2$. Then, $6/\lfloor k/L\rfloor \le \varepsilon$. By Proposition 1, $\mu A_k \ge 1 - 2L\,T^{3L-1}e^{-\varepsilon^2 k/3L}$. To obtain $\mu A_k \ge 1 - 1/k(k-1)$ it suffices to show $2LT^{3L-1}k^2 \le e^{\varepsilon^2 k/3L}$, which can be proved to hold for any $k\ge 1$.

From now on let k_0 be the value established in Proposition 2. Turing recursively defines the set $E_{c(k,n)}$ as a subset of A_k with measure *exactly* $1-1/k+1/(k+n)$.

Definition 5. Let $c(k,n)$ be the function of two integer variables with values in finite sets of pairs of rational numbers such that, for each k and n, $E_{c(k,n)} = (a_1,b_1)\cup(a_2,b_2)\cup\ldots(a_m,b_m)$ denotes the finite union of the intervals whose rational endpoints are given by the pairs in the set $c(k,n)$. For any $k\ge k_0$ let $E_{c(k,0)} = (0,1)$ and $E_{c(k,n+1)} = A_{k+n+1}\cap E_{c(k,n)}\cap(\beta_n,1)$ where $(\beta_n,1)$ is an interval such that $\mu E_{c(k,n+1)} = 1 - 1/k + 1/(k+n+1)$.

The β_n above necessarily exists, it is unique, and it is a rational number computable from the two other sets in the definition. Both are a union of finitely many intervals with rational endpoints, so their respective measure are computable, and they are big enough.

Proof of Turing's Theorem 1. We first prove that $\bigcap_{k\geq k_0} A_k$ contains only normal numbers. By way of contradiction assume $x \in \bigcap_{k\geq k_0} A_k$ and x is not normal to base b. Then, $\lim_{k\to\infty} \frac{S(x,b,w,k)}{k} \neq \frac{1}{b^r}$ for some block w of length r. So, there is $\delta > 0$ and there are infinitely many values k such that $|S(x,b,w,k) - k/b^r| > k\delta$. Let $T(k)$, $L(k)$ and $\varepsilon(k)$ be the assignments of Definition 4 and fix $k_1 \geq k_0$ large enough such that $T(k_1) \geq b$, $L(k_1) \geq r$ and $\varepsilon(k_1) \leq \delta$. This is always possible because $T(k)$ and $L(k)$ are increasing in k, and $\varepsilon(k)$ goes to 0 as k increases. Then, for each $k \geq k_1$, $x \in A_k$ and by Definition 3, $|S(x,b,w,k) - k/b^r| < k\,\varepsilon(k) \leq k\delta$, a contradiction. $E(k) \subseteq \bigcap_{i\geq k} A_i$ for $k \geq k_0$; therefore, all real numbers in $E(k)$ are normal. Since $\mu E_{c(k,n)} = 1 - 1/k + 1/(k+n)$, $\mu E(k) = \lim_{n\to\infty} \mu E_{c(k,n)} = 1 - 1/k$. This completes the proof.

2.2 Turing's Theorem 2: An Algorithm to Output Normal Numbers

Turing's algorithm is uniform in the parameter k and it receives as input an infinite sequence ν of zeros and ones. The algorithm works by stages. The main idea is to split the unit interval by halves, successively. It starts with the whole unit interval and at each stage it chooses either the left half or the right half of the current interval. The sequence $\alpha(k,\nu)$ of zeros and ones output by the algorithm is the trace of the left/right selection at each stage. The invariant condition of the algorithm is that the intersection of the current interval with the set $E(k)$ of normal numbers of Theorem 1 has positive measure. Since $E_{c(k,n)}$ is the finite approximation of $E(k)$ at stage n, the algorithm chooses the half of the current interval whose intersection with $E_{c(k,n)}$ reaches a minimum threshold of measure which avoids running out of measure at any later stage. In case both halves reach this minimum, the algorithm uses the n-th symbol of the input sequence ν to decide. The chosen intervals at successive stages are nested and their measures converge to zero; therefore, their intersection contains exactly one number. This is the sequence $\alpha(k,\nu)$ output by the algorithm. The algorithm is correct if the number denoted by $\alpha(k,\nu)$ is normal to base two. This is proved by induction on the stage n, the only non obvious part is the verification of the invariant condition.

Each sequence output by the algorithm has an explicit convergence to normality: in the initial segment of length ℓ in each base up to base $T(\ell)$, all blocks of length up to $L(\ell)$ occur with the expected frequency plus or minus at most $\varepsilon(\ell)$, where $L(\ell) = \sqrt{\ln \ell}/4$, $T(\ell) = e^L$ and $\varepsilon(\ell) = e^{-L^2} = k^{-1/16}$.

The time complexity of the algorithm is the number of needed operations to produce the n-th digit of the output sequence $\alpha(k,\nu)$. This just requires to compute, at each stage n, the measure of the intersection of the current interval with the set $E_{c(k,n)}$. Turing gives no hints on properties of the sets $E_{c(k,n)}$ that could allow for a fast calculation. The naive way does the combinatorial

construction of $E_{c(k,n)}$ in a number of operations exponential in n. Turing's algorithm verbatim would have simple-exponential time complexity, but we have been unable to verify its correctness. In our reconstruction in [1] the number of intervals we consider in $E_{c(k,n)}$ is exponentially larger than in Turing's literal formulation, so we end up with *double-exponential* time complexity.

Proof of Turing's Theorem 2. Let k be the integer parameter and ν the input infinite sequence of zeros and ones. We write α to denote the output sequence, $\alpha(i)$ for its digit in position i. Similarly for ν. Redefine the computable function $c(k,n)$ of Theorem 1 as follows. Assuming k is big enough, let $E_{c(k,0)} = (0,1)$ and for $n > 0$, $E_{c(k,n)} = A_{k2^{2n+1}} \cap E_{c(k,n-1)} \cap (\beta_n, 1)$, where $(\beta_n, 1)$ is an interval such that $\mu E_{c(k,n)} = 1 - 1/k + 1/k2^{2n+1}$. Here is the algorithm:

`Start with` $I_0 = (0,1)$. `At stage` $n > 0$,
`Split the interval` $I_{n-1} = (a_{n-1}, b_{n-1})$ `into two halves`
$I_n^0 = (a_{n-1}, \frac{a_{n-1}+b_{n-1}}{2})$ `and` $I_n^1 = (\frac{a_{n-1}+b_{n-1}}{2}, b_{n-1})$.
`If` $\mu(E_{c(k,n)} \cap I_n^0) > 1/k2^{2n}$ `and` $\mu(E_{c(k,n)} \cap I_n^1) > 1/k2^{2n}$ `then`
`let` $\alpha(n) = \nu(n)$ `and` $I_n = I_n^{\nu(n)}$.
`Else if` $\mu(E_{c(k,n)} \cap I_n^1) \leq 1/k2^{2n}$ `then`
`let` $I_n = I_n^0$ `and` $\alpha(n) = 0$.
`Else, let` $I_n = I_n^1$ `and` $\alpha(n) = 1$.

To show that α is normal, we prove $\alpha \in E(k) = \bigcap_n E_{c(k,n)}$ by induction on n. For $n = 0$, $E_{c(k,0)} = (0,1)$; so, $\mu(E_{c(k,n)} \cap I_0) = 1 > 1/k$. For $n > 0$, assume the inductive hypothesis $\mu(E_{c(k,n)} \cap I_n) > 1/k2^{2n}$. Since the sets $E_{c(k,n)}$ are nested

$$E_{c(k,n+1)} \cap I_n = \big(E_{c(k,n)} \cap I_n\big) \setminus \big(\big(E_{c(k,n)} \setminus E_{c(k,n+1)}\big) \cap I_n\big).$$

So, $\mu(E_{c(k,n+1)} \cap I_n) = \mu(E_{c(k,n)} \cap I_n) - \mu(\big(E_{c(k,n)} \setminus E_{c(k,n+1)}\big) \cap I_n)$. Then, $\mu(E_{c(k,n+1)} \cap I_n) \geq \mu(E_{c(k,n)} \cap I_n) - \mu(E_{c(k,n)} \setminus E_{c(k,n+1)})$. Using the equality $\mu(E_{c(k,n)} \setminus E_{c(k,n+1)}) = 1/k2^{2n+1} - 1/k2^{2(n+1)+1}$ and the inductive hypothesis, we obtain $\mu(E_{c(k,n+1)} \cap I_n) > 1/k2^{2n} - (1/k2^{2n+1} - 1/k2^{2n+3}) > 2/k2^{2(n+1)}$. It is impossible that both $\mu(E_{c(k,n+1)} \cap I_{n+1}^0)$ and $\mu(E_{c(k,n+1)} \cap I_{n+1}^1)$ be less than or equal to $1/k2^{2(n+1)}$. At least one of the sets $E_{c(k,n+1)} \cap I_{n+1}^i$, for $i \in \{0,1\}$, has measure greater than $1/k2^{2(n+1)}$. The algorithm picks as I_{n+1} the set I_{n+1}^i which fulfills this condition. In case both verify it, the oracle is used to choose left or right. By construction, the expansion of each real number in $E_{c(k,n)} \cap I_n$ starts with $\alpha(0)\ \alpha(1)...\alpha(n)$.

We now prove that for a fixed k, the set of output numbers $\alpha(k,\nu)$ for all possible inputs ν has measure at least $1 - 2/k$. Turing bounds the measure of the unqualified intervals up to stage n, as the n first bits of the sequence ν run through all possibilities. Let $I_m = \big(\frac{m}{2^{n+1}}, \frac{m+1}{2^{n+1}}\big)$, for $m = 0, 1, ..., 2^{n+1} - 1$. The algorithm discards the interval I_m when $\mu(E_{c(k,n)} \cap I_m) \leq 1/k2^{2n}$. The set of intervals that are *not* discarded is recursively defined as follows. Let $M(k,0) = (0,1)$ and for $n > 0$, let $M(k,n+1)$ be the union of the intervals I_m such that $I_m \subseteq M(k,n)$ and $\mu(E_{c(k,n)} \cap I_m) > 1/k2^{2n}$. Then, $\mu(E(k) \cap M(k,n+1))$ equals

$$\mu(E(k) \cap M(k,n)) - \sum_{m=0}^{2^n-1} \mu(E(k) \cap (M(k,n) \setminus M(k,n+1)) \cap \left(\frac{m}{2^n}, \frac{m+1}{2^n}\right)).$$

Each term in the sum is at most $1/k2^{2n}$. Therefore, $\mu(E(k) \cap M(k, n+1)) \geq \mu(E(k) \cap M(k, n)) - 1/k2^n$. Applying this inequality recursively n times, we get $\mu(E(k) \cap M(k, n+1)) \geq \mu(E(k) \cap M(k, 1)) - 1/k \sum_{i=1}^{n} 1/2^n$. Finally, since $E_{c(k,0)} = (0, 1)$ and $k \geq 2$, $M(k, 1) = (0, \frac{1}{2}) \cup (\frac{1}{2}, 1)$; so, $E(k) \cap M(k, 1) = E(k)$. Then, $\mu(E(k) \cap \bigcap_n M(k, n)) > \mu E(k) - 1/k$. Using that $\mu E(k) = 1 - 1/k$, conclude that $E(k) \cap \bigcap_n M(k, n)$ has measure at least $1 - 2/k$.

3 Towards the Theory of Algorithmic Randomness

Turing's manuscript conveys the impression that he had the insight, ahead of his time, that traditional mathematical concepts specified by finitely definable approximations, such as measure or continuity, could be made computational. This point of view has developed under the general name of *effective mathematics*, a part of which is algorithmic randomness. From the modern perspective, Turing's construction of the set of normal numbers in Theorem 1, done via finite approximations, is an instance of a fundamental entity in the theory of algorithmic randomness: a *Martin-Löf test*[5] [19]. Intuitively, a real number is random when when it exhibits the almost-everywhere behavior of all reals, for example its expansion has no predictable regularities. A random real number must pass every test of these properties. Martin-Löf had the idea to focus just in properties definable in terms of computability: a test for randomness is a uniformly computably enumerable sequence of sets whose measure converges to zero. A real number is random if it is covered by no such test. That is to say that it has the almost-everywhere property of avoiding the measure-zero intersection. This definition turned out to be equivalent to the definition of randomness in terms of description complexity [8]. The equivalence between the two been taken as a sign of robustness of the defined notion of randomness.

Definition 6. *1. A Martin-Löf randomness test, hereafter ML-test, is a uniformly computably enumerable sequence $(V_i)_{i\geq 0}$ of sets of intervals with rational endpoints such that, for each i, $\mu V_i \leq 2^{-i}$.*
2. A real number x is random if for every ML-test $(V_i)_{i\geq 0}$, $x \notin \bigcap_{i\geq 0} V_i$.

Turing's set $E(k)$ of Theorem 1 leads immediately to a ML-test[6]. Hence, it provides a direct proof that randomness implies normality.

Corollary 1. *The sequence $(V_k)_{k\geq 0} = ((0, 1) \setminus E(2^k))_{k\geq 0}$ is a ML-test.*

Proof. By Theorem 1, $E(k) = \bigcap_{n\geq 1} E_{c(k,n)}$, where $c(k, n)$ is computable and for each k and n, $E_{c(k,n)}$ is a finite set of intervals with rational endpoints. So, the complement of each $E_{c(k,n)}$ is also a finite set of intervals with rational

[5] Martin-Löf presented the test in terms of sequences of zeros and ones. We give here an alternative formulation in terms of sets of intervals with rational endpoints.

[6] In fact, Theorem 1 yields a *Schnorr test* [21]. This is a ML-test where μV_i is computable uniformly in i. The notion is unchanged if we, instead, let a Schnorr test be a ML-test such that $\mu V_i = 2^{-i}$, for each $i \geq 0$.

endpoints. Then, $(0,1) \setminus E(k) = \bigcup_{n\geq 1}(0,1) \setminus E_{c(k,n)}$ is computably enumerable. Since Turing's construction is uniform in the parameter k, $((0,1) \setminus E(k))_{k\geq 0}$ is uniformly computably enumerable. Finally, since the measure of $E(k)$ is $1-1/k$, $\mu((0,1) \setminus E(k)) = 1/k$. Thus, $(V_k)_{k\geq 0} = ((0,1) \setminus E(2^k))_{k\geq 0}$ is a ML-test.

Corollary 2. *Randomness implies normality.*

Proof. If x is not normal then, by Theorem 1, x belongs to no set $E(k)$, for any k. So, $x \in \bigcap_{k\geq 0}(0,1) \setminus E(k)$. By Corollary 1, $(V_k)_{k\geq 0} = ((0,1) \setminus E(2^k))_{k\geq 0}$ is a ML-test. Hence, $x \in \bigcap_{k\geq 0} V_k$; therefore, x is not random.

References

1. Becher, V., Figueira, S., Picchi, R.: Turing's unpublished algorithm for normal numbers. Theoretical Computer Science 377, 126–138 (2007)
2. Becher, V., Figueira, S.: An example of a computable absolutely normal number. Theoretical Computer Science 270, 947–958 (2002)
3. Borel, É.: Les probabilités dénombrables et leurs applications arithmétiques. Rendiconti del Circolo Matematico di Palermo 27, 247–271 (1909)
4. Borel, É.: Leçons sur la thèorie des fonctions, 2nd edn., Gauthier Villars (1914)
5. Bugeaud, Y.: Nombres de Liouville et nombres normaux. Comptes Rendus de l'Académie des Sciences de Paris 335, 117–120 (2002)
6. Bugeaud, Y.: Distribution Modulo One and Diophantine Approximation. Cambridge University Press (2012)
7. Bourke, C., Hitchcock, J., Vinodchandran, N.: Entropy rates and finite-state dimension. Theoretical Computer Science 349(3), 392–406 (2005)
8. Chaitin, G.: A theory of program size formally identical to information theory. Journal ACM 22, 329–340 (1975)
9. Cassels, J.W.S.: On a paper of Niven and Zuckerman. Pacific Journal of Mathematics 2, 555–557 (1952)
10. Downey, R., Hirschfeldt, D.: Algorithmic Randomness and Complexity. Springer (2010)
11. Downey, R.: Randomness, Computation and Mathematics. In: Cooper, S.B., Dawar, A., Löwe, B. (eds.) CiE 2012. LNCS, vol. 7318, pp. 163–182. Springer, Heidelberg (2012)
12. Hardy, G.H., Wright, E.M.: An Introduction to the Theory of Numbers, 1st edn. Oxford University Press (1938)
13. Harman, G.: Metric Number Theory. Oxford University Press (1998)
14. Kučera, A., Slaman, T.: Randomness and recursive enumerability. SIAM Journal on Computing 31(1), 199–211 (2001)
15. Kuipers, L., Niederreiter, H.: Uniform Distribution of Sequences. Dover (2006)
16. Lebesgue, H.: Sur certaines démonstrations d'existence. Bulletin de la Société Mathématique de France 45, 132–144 (1917)
17. Levin, M.B.: On absolutely normal numbers. English translation in Moscow University Mathematics Bulletin 34, 32–39 (1979)
18. Dai, L., Lutz, J., Mayordomo, E.: Finite-state dimension. Theoretical Computer Science 310, 1–33 (2004)
19. Martin-Löf, P.: The Definition of Random Sequences. Information and Control 9(6), 602–619 (1966)

20. Nies, A.: Computability and Randomness. Oxford University Press (2009)
21. Schnorr, C.-P.: Zufälligkeit und Wahrscheinlichkeit. In: Eine algorithmische Begründung der Wahrscheinlichkeitstheorie. Lecture Notes in Mathematics, vol. 218. Springer, Berlin (1971)
22. Schnorr, C.-P., Stimm, H.: Endliche Automaten und Zufallsfolgen. Acta Informatica 1, 345–359 (1972)
23. Sierpiński, W.: Démonstration élémentaire du théorème de M. Borel sur les nombres absolument normaux et détermination effective d'un tel nombre. Bulletin de la Société Mathématique de France 45, 127–132 (1917)
24. Turing, A.M.: On computable numbers, with an application to the Entscheidungsproblem. Proceedings of the London Mathematical Society Series 2 42, 230–265 (1936)
25. Turing, A.M.: A note on normal numbers. In: Britton, J.L. (ed.) Collected Works of A.M. Turing: Pure Mathematics, pp. 263–265. North Holland, Amsterdam (1992); with notes of the editor in 263–265
26. Schmidt, W.M.: On normal numbers. Pacific Journal of Math. 10, 661–672 (1960)
27. Strauss, M.: Normal numbers and sources for BPP. Theoretical Computer Science 178, 155–169 (1997)