Algorithmic Randomness I

Rod Downey

Victoria University

Wellington

New Zealand

The Basic Refs are van Lambalgen's Thesis, Solovay's unpublished notes, and Li-Vitanyi. Also a new book "to appear" by Downey and Hirschfeldt prelim version on my home page.

And *Calibrating Randomness* (with Hirschfeldt, Nies and Terwijn) for BSL, soon on my web page.

Some Computability-Theoretical Aspects of Reals and Randomness, to appear, in a *Lecture Notes in Logic* volume edited by Cholak. et. al.

Some of the papers can be found in

`www.mcs.vuw.ac.nz/`

`research/math-pubs.shtml`

Nies home page, Hirschfeldt's home page.

Motivation

- What is “random”?
- How can we calibrate levels randomness? Among randoms?, Among non-randoms?
- How does this relate to classical computability notions, which calibrate levels of computational complexity?
- Von Mises, Church, Solomonoff, Levin, Chaitin, Kolmogorov, Shannon, etc.

Notation

- Real is a member of Cantor space 2^ω with topology with basic clopen sets $[\sigma] = \{\sigma\alpha : \alpha \in 2^\omega\}$ whose measure is $2^{-|\sigma|}$.
- for uniformity, a real is always nonrational.
- Strings = members of $2^{<\omega} = \{0, 1\}^*$.

Kolmogorov Complexity

- Capture the incompressibility paradigm. Random means hard to describe, incompressible: e.g. 1010101010.... (10000 times) would have a short program.

- A string σ is random iff the only way to describe it is by hardwiring it. (Formalizing the Berry paradox)

- For a fixed machine N, we can define

- The *Kolmogorov complexity* $C(\sigma)$ of $\sigma \in \{0,1\}^*$ with respect to N, is $|\tau|$ for the shortest τ s.t. $N(\tau)\downarrow = \sigma$. (Kolmogorov)

- A string σ is N-random iff $C_N(\sigma) \geq |\sigma|$.
- A machine U is called weakly universal iff for all N, there is a d such that for all σ, $C_U(\sigma) \leq C_N(\sigma) + d$.
- Actually we will always use universal machines where the e-th machine is coded in a computable way.
- They exist (Kolmogorov). Hence there is a notion of Kolmogorov randomness for strings up to a constant.
- Proof: We can enumerate the Turing machines $\{M_e : e \in \mathbb{N}\}$. Define

$$U(1^e 0\sigma) = M_e(\sigma).$$

This particular coding gives $C(\tau) \leq M_e(\tau) + e + 1$.

- Thus we can define the plain Kolmogorov complexity of a string σ as $C(\sigma)$ for a fixed universal machinei U.
- We can similarly do an oracle version of this and can define $C(x|y)$ as the Kolmogorov complexity of x given y.
- The unique string τ which first occurs of length $C(\sigma)$ is denoted by x^* (really x^*_C).

- Here are some basic facts about C-complexity:

  (i) $C(x, C(x)) = C(x^*)$.

  (i) $C(x|x^*) = O(1)$

  (iii) $C(x, C(x)|x^*) = C(x^*|C(x), x) = O(1)$.

  (iv) $C(xy) \leq C(x, y) + O(1)$ where xy denotes the concatenation of x and y and $C(x, y)$ denotes $C(\langle x, y\rangle)$.

Plain Counting Thm

- The following is the basic fact that makes the theory work.
- (Plain Counting Theorem-Kolmogorov) $|\{\tau : C(\tau) \leq |\tau| - d\}| \leq O(1)2^{|\tau|-d}$.
- Proof: pigeonhole principle.
- We say that σ is C-*random* iff $C(\sigma) \geq |\sigma|$.

Compression functions

- Thus plain complexity is a *combinatorial fact*
- (Nies, Stephan Terwijn) We say that $F : \Sigma^* \mapsto \Sigma^*$ is a compression function if for all x $|F(x)| \leq C(x)$ and F is 1-1.
- Note that the counting theorem works for compression functions.
- Now we can form a Π^0_1 class of compression functions. We can apply then various basis Theorems, for instance, the Low Basis Theorem.

- There is a infinite low set of C-random strings.
- In some sense this is the best you could hope for. The collection of C-random strings is easily seen to be immune.
- To see this, let $A = \{x : C(x) \geq \frac{|x|}{2}\}$. Then A is immune. Suppose that A has an infinite c.e. subset B. Let $h(n)$ be defined as the first element of B to occur in its enumeration of length above n. Then
$$C(h(n)) \geq \frac{|h(n)|}{2} \geq \frac{n}{2}, \text{ but,}$$
$C(h(n)) \leq C(n) + \mathcal{O}(1) \leq |n| + O(1)$. For large enough n this is a contradiction.

C-overgraphs

- We can easily see that R_C, the collection of C-randoms is wtt complete.
- For each n, choose a length $f(n)$ and, at each stage s point at a string $\sigma(n, s)$ which is C_e-random.
- Should $\sigma(n, s)$ become nonrandom due to a play by our opponent RED choose the next string of this length. Should we see n enter $\emptyset'$ at s, we (BLUE) drops the complexity of $\sigma(n, s)$.

Kummer's Theorem

- It was a question whether R_C could be tt-complete, so that the reduction above was non-adaptive.
- Theorem (Kummer) R_C and hence the *overgraph* $M_C = \{(x, y) : C(x) < y\}$ is tt-complete.

- The proof is tricky and nonuniform. It used *blocks* instead of the $\sigma(n, s)$ above and is a conjunctive tt-reduction. The nonuniformity comes from the combinatorics. A finite number of tries occur for these blocks, but this will be bounded and the number that occurs infinitely often is the one.

Muchnik's Theorem

- The following is easier and along the same lines.
- Theorem (An. A. Muchnik) The conditional overgraph $M = \{(x, y, n) : C(x|y) < n\}$ is creative

- The proof. We need $\emptyset' \leq_m M$.
- Parameter d known in advance.
- Construct possible g_x for $x \in [1, 2^d]$.
- Either we know $z \in \emptyset'$, or there is a unique y such that $g_x(z) = (x, y, d)$ and $x \in \emptyset'$ iff $g_x(z) \in M$.
- For some maximal x which enumerates elements infinitely often, g_x works.

- **Construction, stage** $s+1$ For each active $y \leq s$, find the least $q \in [1, 2^p]$ with
$$(q, y, d) \notin M_s.$$
(Notice that such an x needs to exist since $\{q : (q, y, d) \in M\} < 2^d$.)

- Now for any v, if v enters $\emptyset'[s+1]$, find the largest r, if any, with $g_r(z)$ defined. If one exists, enumerate $g_r(z)$ into M. Find $\hat{y}$ with $g_r = (r, \hat{y}, d)$. Declare that $\hat{y}$ is no longer active.

- Let x be tha maximal r for which we put $g_r(z)$ into M infinitely often. (any y can only compress so many of $[1, 2^d]$) It works.

- There is a lot of very interesting work by Allender and others about what is *efficiently* reducible to R_C, and this (apparently) relates to standard classes like PSPACE, NP, etc. The point is that here the reductions are big.

- For instance, Allender, Buhrmann, Koucký look at the hypothesis

$$PSPACE = \cap_V P^{R_C^V}$$

(R_C^V is R_C for universal V.)

Complexity Oscillations

- Tempting but false $C(xy) \leq C(x) + C(y) + O(1)$. The false argument says : concatenate the machines

- The problem is where does x^* stop and y^* begin.

- Martin-Löf showed that the formula always fails for long enoug srings and hence reals.

- Why? Take any α. Then, as a string $\alpha \upharpoonright n$ corresponds to some number which we can interpret as a string using llex ordering: $\alpha \upharpoonright n$ is the m-th string.

- Now consider the program that does the following. It takes a strings ν, interprets its length $m_\nu = |\nu|$ as a string, $\sigma = \sigma_m$ and outputs $\sigma\nu$.

- Apply this to the string τ whose length is m th code of $\alpha \upharpoonright n$.

- The output would be much longer, and would be $\alpha \upharpoonright m + n$, with input having length m. Thus $C(\alpha \upharpoonright m + n) < m + n - O(1)$.

- This phenomenom is fundamental in our understanding of Kolmogorov complexity and is called *complexity oscillations*.
- There are several known ways to get round this problem to cause only to get the information provided by the *bits* of the strings.

Symmetry of Information

- The *information content* of a string y in a string x is defined as

$$I(x:y) = C(y) - C(y|x).$$

- (Levin-Kolmogorov)

$$\begin{aligned} I(x:y) &= I(y:x) \pm O(\log n) \\ &= I(y:x) \pm O(\log C(x,y)) \end{aligned}$$

where $n = \max\{|y|, |x|\}$.

- (restated) $C(x,y) = C(x) + C(y|x) + O(\log C(x,y))$

Prefix free

universal computers

- Levin, Gaćs, Chaitin.
- Computers have alphabet $\{0, 1\}$.
- A computer M is *prefix-free* if

$$(M(\sigma)\downarrow \ \wedge \ \sigma' \supsetneq \sigma) \Rightarrow M(\sigma')\uparrow .$$

- A prefix-free machine is universal if every other one is coded in it.
- They exist, same proof.
- Building them uses Kraft-Chaitin.

Kraft-Chaitin

- Theorem(Kraft)
  - (i) If A is prefix-free then $\sum_{n \in A} 2^{-|n|} \leq 1$.
  - (ii) (This part is now called Kraft-Chaitin, or Chaitin simulation) Let $d_1, d_2, \cdots$ be a collection of lengths, possibly with repetitions, Then $\Sigma 2^{-d_i} \leq 1$ iff there is a prefix-free set A with members σ_i and σ_i has length d_i. Furthermore from the sequence d_i we can effectively compute the set A.
- Proof: On direction of Kraft-Chaitin is clear. This is because of the

topological correspondence
$\Delta : [\sigma] \mapsto [0.\sigma, 0.\sigma + 2^{-|\sigma|})$ taking the string σ to an interval of size $2^{-|\sigma|}$, gives a correspondence between a set of disjoint intervals in $[0, 1)$ and a prefix-free set.

- (noneffective) Given lengths $\{d_i : i \in \mathbb{N}\}$ in some random order.

- Arrange in increasing order, say $l_1 \leq l_2 \leq \dots$.

- Choose disjoint intervals I_j, with the right end-point of I_n as the left endpoint of I_{n+1} and the length of I_{n+1} being $2^{-l_{n+1}}$. Then we can again use the correspondence by setting $[\sigma_n] = \Delta^{-1}(I_n)$.

- Pippinger's (Chaitin's) process: (Using a trick of Joe Miller) The idea is that, at each stage n, we have a mapping $d_i \mapsto [\sigma_i]$, $|\sigma_i| = d_i$, together with a binary string $x[n] = .x_1x_2 \ldots x_m$ representing the length $1 - \sum_{j \leq n} 2^{-d_j}$.
- Ensure for 1 in the expansion that there is a string of precisely that length in $2^{<\omega} - \{\sigma_j : j \leq n\}$.
- To continue the induction, at stage $n+1$, when a new length d_{n+1} enters,
- position $x_{d_{n+1}}$ is a 1. Then we can find the corresponding string $\tau_{d_{n+1}}$ in $2^{<\omega} - \{\sigma_j : j \leq n\}$ and set $\sigma_{n+1} = \tau_{d_{n+1}}$. Then of course we

make $x_{d_{n+1}} = 0$ in $x[n+1]$.

- If position $x_{d_{n+1}}$ is a 0, find the largest $j < d_{n+1}$ with $x_j = 1$, find the lexicographically least string τ extending τ_j of length d_{n+1}, let $\sigma_{n+1} = \tau$, and let $x[n+1] = x[n] - .\nu$ where ν is the string which is zero except for 1 in position d_{n+1}.

- Notice that nothing changes in $x[n+1]$ from $x[n]$ except in positions j to d_{n+1}, and these all change to 1, with the exception of x_j which changes to 0. Since τ was chosen as the lexicographically least string in the cone $[\tau_j]$, there will be corresponding strings in $[\tau_j]$ of lengths $j-1, \ldots, d_{n+1}$, as required to complete the induction.

- (Restatement) Suppose that we are effectively given a set of "requirements" $\langle n_k, \sigma_k \rangle$ for $k \in \omega$ with $\sum_k 2^{-n_k} \leq 1$. Then we can (primitive recursively) build a prefix-free machine M and a collection of strings τ_k with $|\tau_k| = n_k$ and $M(\tau_k) = \sigma_k$.

Prefix-free randomess

- Prefix freeness gets rid of the use of length as extra information: Machines concatenate!
- The *prefix-free complexity* $K(\sigma)$ of $\sigma \in \{0,1\}^*$ is $|\tau|$ for the shortest τ s.t. $M(\tau)\downarrow = \sigma$.
- Note now $K(\sigma) \leq |\sigma| + K(|\sigma|) + d$, about $n + 2\log n$, for $\sigma| = n$.
- Build M, $M(z\sigma) = \sigma$ if $U(z) = |\sigma|$.

K-Counting Theorem

- (Counting Theorem-Chaitin) $|\{\sigma : |\sigma| = n \wedge K(\sigma) \leq n + K(n) - c\}| \leq O(1)2^{n+K(n)-c}$.
- The easiest proof uses semimeasures. A partial function $\widehat{K} : 2^{<\omega} \mapsto \mathbb{N}$ such that

  (i) $\sum_{\sigma \in 2^{<\omega}} 2^{-\widehat{K}(\sigma)} \leq 1$, and,

  (ii) $\{\langle \sigma, k \rangle : \widehat{K}(\sigma) \leq k\}$ is c.e..
- There is a universal minimal one:

$$\widehat{K}(x) = \min_{k \geq 0} \{\widehat{K}_k(x) + k + 1\}.$$

- Using KC K is the same thing!
- Namely, at stage s, if we see $K_s(\sigma) = k$ and $K_{s+1}(\sigma) = k' < k$ enumerate a Kraft-Chaitin axiom $\langle 2^{-(k'+1)}, \sigma \rangle$ to describe M, and hence generate $\widehat{K} = K_M$.

- Many proofs exploit the minimality of K.
- Strictly speaking, A discrete semimeasure is function $m : 2^{<\omega} \mapsto \mathbb{R}^+ \cup \{0\}$ such that
$$\sum_{\sigma \in 2^{<\omega}} m(\sigma) \leq 1.$$
- NB Discrete Lebesgue measure is $\lambda(\sigma) = 2^{-2|\sigma|-1}$.
- Let m denote the minimal universal discrete semimeasure. Then
- $K(\sigma) = -\log m(\sigma) + O(1)$.

Proof of Counting

-

- (Counting Theorem-Chaitin) $|\{\tau : |\sigma| = |\tau| = n \wedge K(\sigma) \leq K(\tau) + d - c\}| \leq O(2^d)2^{n+K(n)-c}$.

- Note: $\sum 2^{-K(n)} = \sum_n \sum_{|\sigma|=n} 2^{-K(\sigma)}$.

- Now, as K is minimal, we have
$$2^{-K(n)+O(1)} \geq \sum_{|\sigma|=n} 2^{-K(\sigma)}.$$

- suppose that there are more than 2^{n-k+c} strings of length n with $K(\sigma) < n + K(n) - k$.

- Let $F = \{\sigma : |\sigma| = n \wedge K(\sigma) < n + K(n) - k\}$. (the good)
- Then

$$2^{-K(n)+c} \geq \sum_{|\sigma|=n} 2^{-K(\sigma)} \geq$$

$$\sum_{\sigma \notin F} 2^{-K(\sigma)} + \sum_{\sigma \in F} 2^{-K(\sigma)}$$

$$> (1+\epsilon) 2^{n-k+c} 2^{n-K(n)-k} > 2^{-K(n)+c},$$

a contradiction. (There are too many bads)

The Coding Theorem

- Let $Q_D(\sigma) = \mu(D^{-1}(\sigma))$, the probability tht σ is output.
- (The Coding Theorem) $-\log m(\sigma) = -\log Q(\sigma) + O(1) = K(\sigma) + O(1)$.

- (Proof) $Q(\sigma) \geq 2^{-K(\sigma)} = 2^{-|\sigma^*|}$, since $D(\sigma^*) = \sigma$.
- So $-\log Q(\sigma) \leq K(\sigma)$.
- But: $\sum 2^{-logQ(\sigma)} \leq \sum_\sigma Q(\sigma) \leq 1$.
- Now use minimality of K.
- (Remark) It is not hard to show that for any σ $Q(\sigma)$ is random.

An Application

- One nice applications shows that within a fixed diameter there are relatively few descriptions.
- Theorem (Chaitin, Levin) There is a constant d such that for all c and all σ,

$$|\{\nu : U(\nu) = \sigma \wedge |\nu| \leq K(\sigma)+c\}| \leq d2^c.$$

- The point here is that d is independent of $|\nu|$ and depends only on the Recursion Theorem, and c

- Proof: Trivially,

$$\mu(\{\nu : U(\nu) = \sigma \wedge |\nu| \leq K(\sigma)+c\}) \geq$$
$$2^{-(K(\sigma)+c)} \cdot |\{\nu : U(\nu) = \sigma \wedge |\nu| \leq K(\sigma)+c\}|.$$

  But also, $\mu(\{\nu : U(\nu) = \sigma \wedge |\nu| \leq K(\sigma) + c\}) \leq d \cdot 2^{-K(\sigma)}$, by the Coding Theorem.

- Thus,

$$d2^{-K(\sigma)} \geq 2^{-c}2^{-K(\sigma)}|\{\nu : U(\nu) = \sigma \wedge |\nu| \leq$$

  Hence, $d2^{c} \geq |\{\nu : U(\nu) = \sigma \wedge |\nu| \leq K(\sigma) + c\}|$.

Symmetry of Information

- $K(xy) \leq K(x) + K(y) + O(1)$.
- Define $I(x : y) = K(y) - K(y|x)$.
- Levin and Gács, Chaitin $I(\langle x, K(x)\rangle : y) = I(\langle y, K(y)\rangle : x) + O(1)$.
- (restated)
  $K(x, y) = K(x) + K(y|x^*) = K(x) + K(x|x, K(x))$.
- The proof uses KC again. And the Coding Theorem.

- Clearly $K(x,y) \leq K(x) + K(y|x^*)(+O(1))$.
- RTP $K(y|x^*) \leq K(x,y) - K(x)$
- At each stage s, have a unique p_s, $U(p_s)\downarrow$.
- $U(p_s) = (x_s, y_s)$.
- by Coding Thm $2^{K(x)-c}\sum_y Q(x,y) \leq 1$. for all x as $\sum_y Q(x,y)$ is an information content measure of x.
- We build a machine. M. With x' on tape, M first simulates $U(x')$. So with x^* on tape M will simuate $U(x^*) = x$.

- Then M simulates M_x described by the set W KC axioms:

  $(|p_t| - |x^*| + c, y_t)$, for each $p_t = (x, y_t)$.

- $$\sum_{t \in W} 2^{-(|p_t| - |x^*| + c)}$$

  $$\leq 2^{|x^*| - c} \sum_t 2^{-|p_t|} \leq 2^{K(x) - c'} (\sum_y Q(\langle x, y \rangle))$$

- Finally, for each p with $U(p) = (x, y)$, there is a $\hat{p}$ with $U(\hat{p}|x^*) = M_x(\hat{p}) = y$, and $|\hat{p}| = |p| - K(x) + c$.

- Thus $K(y|x^*) \leq K(x, y) - K(x) + O(1)$.

Prefix free randomness

- Levin-Chaitin random
  $K(x) \geq |x| + O(1)$.
- Strongly $K(x) \geq |x| + K(|x|) + O(1)$.
- Strongly K-random implies
  C-random implies K-random.
- NO reversals (the first is nontrivial
  and due to Solovay)

- As with life, relationships here are complex (Solovay)

$$K(x) = C(x) + C^{(2)}(x) + \mathcal{O}(C^{(3)}(x)).$$

and

$$C(x) = K(x) - K^{(2)}(x) + \mathcal{O}(K^{(3)}(x)).$$

- These 3's are *sharp* (Solovay) That is, for example,
$K = C + C^2 + C^3 + O(C^4)$ is NOT true.

- Is there a infinite low collection of strongly K-random strings. Joe Miller showed that the set is not co-c.e..

- Theorem. (An A Muchnik) There exist universal prefix-free machines V and U such that

  (i) M_K^V is tt-complete.

  (ii) M_K^U (and hence $\overline{R}_K^U$) is not tt-complete.

- The proof of (ii) is very interesting, using strategies for finite games do diagonalize against tt-reductions.

- Thus, the overgraph may or may not be tt-complete depending on the universal machine. Open for monotone complexity, open for the nonrandoms.

Monotone Complexity

- Levin's original idea here was to try to assign a complexity to the *real itself.* That is, think of the complexity of the real as the shortest machine that outputs the real. Hence now we are thinking of machines that take a program σ and might perhaps output a real α. (Nonsense unless α is computable)
- The following definition can be applied to Turing machines with potentially infinite output, and to discrete ones mapping strings to strings. In this definition, we regard

$M(\sigma)\downarrow$ to mean that at some stage s, $M(\sigma)\downarrow[s]$.

- We say that a machine M is *monotone* if its action is continuous. That is, for all $\sigma \preceq \tau$, if $M(\sigma)\downarrow$ and $M(\tau)\downarrow$ then

$$M(\sigma) \preceq M(\tau).$$

- Levin's (standard) monotone complexity Km is defined as follows. Fix a universal monotone machine U.

$$Km(\sigma) = \min\{|\tau| : \sigma \preceq U(\tau)\}.$$

Continuous Semimeasures

- The coding theorem relates K to *discrete semimeasures*. Here we would like an analog.
- Continuous semimeasures.
- A *continuous semimeasure* is a function $\delta : [2^{<\omega}] \mapsto \mathbb{R}^+ \cup \{0\}$ satisfying

  (i) $\delta([\lambda]) \leq 1$, and

  (ii) $\delta([\sigma]) \geq \delta([\sigma 0]) + \delta([\sigma 1])$.

- There is a minimal optimal continuous semimeasure δ. (Actually $\delta([\sigma]) = 2^{-|\sigma|}F(\sigma)$ where F is the optimal supermartingale, for those who know.)
- $KM(\sigma) = -\log \delta([\sigma])$.
- The analog of the Coding Theorem would state $KM = Km$. That is the probability that a string is output (KM) is the same as its Kolmogorov complexity (Km). Note $2^{-Km(\sigma)}$ is a semimeasure.

Gács Theorem

- (i) There exists a function f with $\lim_s f(s) = \infty$, such that for infinitely many σ,

$$Km(\sigma) - KM(\sigma) \geq f(|\sigma|).$$

  (ii) Indeed, we may choose f to be the inverse of Ackkermann's function.

- This shows $\leq_{Km}$ is not the same as $\leq_{KM}$. (Miller observation). Is this true for c.e. reals?

- Find a reasonable proof of Gács Theorem. (Here reasonable=one I can understand)