Studying randomness through computation

André Nies*

January 16, 2011

1 How were you initially drawn to the study of computation and randomness?

My first contact with the area was in 1996 when I still worked at the University of Chicago. Back then, my main interest was in structures from computability theory, such as the Turing degrees of computably enumerable sets. I analyzed them via coding with first-order formulas. During a visit to New Zealand, Cris Calude in Auckland introduced me to algorithmic information theory, a subject on which he had just finished a book [3]. We wrote a paper [4] showing that a set truth-table above the halting problem is not Martin-Löf random (in fact the proof showed that it is not even weakly random [33, 4.3.9]). I also learned about Solovay reducibility, which is a way to gauge the relative randomness of real numbers with a computably enumerable left cut. These topics, and many more, were studied either in Chaitin's work [6] or in Solovay's visionary, but never published, manuscript [35], of which Cris possessed a copy.

In April 2000 I returned to New Zealand. I worked with Rod Downey and Denis Hirschfeldt on the Solovay degrees of real numbers with computably enumerable left cut. We proved that this degree structure is dense, and that the top degree, the degree of Chaitin's Ω, cannot be split into two lesser degrees [9]. During this visit I learned about K-triviality, a notion formalizing the intuitive idea of a set of natural numbers that is far from random.

To understand K-triviality, we first need a bit of background. Sets of natural numbers (simply called *sets* below) are a main topic of study in computability theory. Sets can be "identified" with infinite sequences of bits. Given a set A, the bit in position n has value 1 if n is in A, otherwise its value is 0. A *string* is a finite sequence of bits, such as 11001110110. Let $K(x)$ denote the length of a shortest prefix-free description of a string x (sometimes called the prefix-free Kolmogorov complexity of x even though Kolmogorov didn't introduce it). We say that $K(x)$ is the *prefix-free complexity* of x. Chaitin [6] defined a set $A \subseteq \mathbb{N}$ to be *K-trivial* if each initial segment of A has prefix-free complexity no greater than the prefix-free complexity of its length. That is, there is $b \in \mathbb{N}$ such that, for each n,

*partially supported by the Marsden Fund of New Zealand, grant no. 08-UOA-187.

$$K(A\upharpoonright_n) \leq K(n) + b.$$

(Here $A\upharpoonright_n$ is the string consisting of the first n bits of A. On the right hand side the number n is represented in base 2 by a string.)

Martin-Löf [22] introduced a mathematical notion of randomness that is nowadays regarded as central. It is commonly referred to as Martin-Löf (ML-) randomness, and sometimes as 1-randomness. We will discuss this notion in detail in the next section. K-triviality of sets is the opposite of ML-randomness: K-trivial sets are "antirandom". For, the Levin-Schnorr Theorem says that Z is Martin-Löf random if and only if there is a constant d such that for each n, we have $K(Z\upharpoonright_n) \geq n - d$; on the other hand, Z is K-trivial if the values $K(Z\upharpoonright_n)$ are within a constant of their lower bound $K(n)$, which is at most $2\log n$.

Chaitin showed that each K-trivial set is Δ^0_2, that is, the set is Turing below the halting problem. Downey, Hirschfeldt and I worked our way through Chaitin's proof, the construction in Solovay's manuscript of an incomputable K-trivial, and Calude and Coles' improvement where the set constructed is also computably enumerable (c.e.), which means that one can effectively list its elements in some order. Downey realized that there is a connection between K-triviality of A and a notion introduced by Zambella [37]: a set $A \subseteq \mathbb{N}$ is *low for ML-randomness* if

each ML-random set is already ML-random relative to A.

The phrase "relative to A" means that we can include queries to A in the computations determining, for instance, a ML-test. In this context A is called an "oracle set". Kučera and Terwijn [20] proved that such set can be c.e. but incomputable. Thus, the notion of relative ML-randomness does not always distinguish the oracles with some computational power.

A *lowness property* of a set expresses that the set is, in some specific sense, close to being computable. Unlike K-triviality, which expresses being far from random, Zambella's property is a lowness property defined in terms of relativized randomness.

At first these topics seemed exceedingly strange to me. Starting from April 2000, it took me almost exactly two years to understand the notions of K-triviality, and being low for ML-randomness.

Downey realized, during his visit to the University of Chicago in February 2001, that the dynamics of the Kučera-Terwijn construction of a set that is low for ML-randomness can be adapted for an easy construction of an incomputable, but c.e. K-trivial set. From 2002 on, the language of cost functions was developed as an abstract framework for such constructions; see [33, Section 5.3].

End of May 2001 I left the University of Chicago. I had more than a year ahead of me for pure research, knowing that I would safely start at the University of Auckland in mid-2002. In July 2001, Denis Hirschfeldt and I together travelled to Italy. After that, we met with Rod Downey at the Vienna summer conference of the Association for Symbolic Logic. Progress was slow but steady. For a while, we believed that a K-trivial set can be Turing complete!

However, in discussions of Downey and Hirschfeldt, obstructions to building a Turing complete K-trivial emerged.

In August 2001 a group of researchers including Denis Hirschfeldt, Frank Stephan and Jan Reimann met at the University of Heidelberg. Denis had the crucial idea how to turn these obstructions into a proof that each K-trivial set is Turing incomplete. Eventually we published these findings in [10]. The mechanism in Hirschfeldt's construction has been described by a stack of decanters holding precious wine [11]. The height of the stack is essentially given by the constant b in the definition of K-triviality of the set A. Wine is first poured into the top decanter (in smaller and smaller quantities). A decanter that is not at the bottom can be emptied into the next lower decanter. The purpose is to fill the bottom decanter up to a certain amount, while spilling as little as possible; this yields a contradiction to the Turing completeness of A. An elaborate argument one could call a "garbage lemma" shows that the amount one spills is indeed bounded. Such garbage lemmas recur in several related results.

After the stay in Heidelberg I went to Novosibirsk for a month, and worked with Andrei Morozov on questions related to algebra and its interaction with logic. I returned to the topics discussed above during an epic trip to Lake Baikal, Mongolia and then China on the Trans-Siberian Railway. I remember working on this in Goryachinsk, a resort for Soviet war veterans on the remote Eastern side of the lake 3 hours by minibus from Ulan-Ude, the capital of the Russian province of Buryatia. I also remember a hotel room in Southern China where I was trying to write up a proof that in the c.e. Turing degrees, each proper Σ^0_3 ideal has a low$_2$ upper bound [1]. This is the mix of old and new I was immersed in. As the class of K-trivials is closed under effective join $\oplus$ and every K-trivial is Turing incomplete [10], the c.e. K-trivials seemed to be a natural candidate for such an ideal, except that we didn't know yet whether the K-trivials are closed downward under Turing reducibility $\leq_T$.

2 What have we learned?

A set of natural numbers can be studied under two aspects, its *randomness* and its *computational complexity*. We now understand both aspects. We also know that they are closely related. There are strong interactions from computability to randomness, and conversely, from randomness to computability.

2.1 The randomness aspect of a set

For infinite sequences of bits, there is no single formal notion corresponding to our intuition of randomness. Our intuition is simply too vague for that. Instead, there is a hierarchy of formal randomness notions, determined by the strength of the algorithmic methods that are allowed for defining a test concept. This can be traced back to the admissible selection rules of von Mises [36].

The infinite sequences of bits form the points of a topological space called *Cantor Space*. Martin-Löf [22] defined a set to be random in a formal sense if it passes each test in a certain collection of effective tests: a *ML-test* is a sequence $(G_m)_{m \in \mathbb{N}}$ of uniformly c.e. open sets in Cantor space of "size" at most 2^{-m} (formally, the size is the product measure λG_m). *Z passes* this test if Z is not in all G_m. Z is *Martin-Löf random* if it passes all ML-tests.

Many notions in the hierarchy of formal randomness notions can be defined via modifying Martin-Löf's test notion. If passing effective statistical tests such as the law of large numbers is all you want, then your notion might be Schnorr randomness, which is weaker than ML-randomness. A test $(G_m)_{m \in \mathbb{N}}$ has to satisfy the additional condition that the size of G_m is a computable real number uniformly in m.

If computability theoretic criteria matter to you, then Schnorr randomness is not enough because there is a Schnorr random set Z where the sequence of bits in the even positions is Turing equivalent to the bits in the odd positions [33, 3.5.22]. This cannot happen any longer for a ML-random set. But maybe you also think that a real with computably enumerable left cut, such as Ω, should not be called random (when viewed in its binary representation). In that case, try weak 2-randomness. A Π^0_2 class in Cantor Space has the form $\{Z \colon Z \restriction_n \in R \text{ for infinitely many } n\}$, where R is some computable set of strings. Z is *weakly 2-random* if Z is in no Π^0_2 class of measure 0. Next is 2-randomness, namely ML-randomness relative to the halting problem. This notion was already studied in 1981 by Kurtz [21]. He showed that each 2-random Z is c.e. in some set $Y <_T Z$. Similar to ML-randomness, it has a characterization via incompressibility of initial segments: Z is 2-random $\Leftrightarrow$ for infinitely many n the initial segment $Z \restriction_n$ is incompressible in the sense of plain Kolmogorov complexity C (see [33, 3.6.20]).

So far, all tests were definable in arithmetic. If such tests are not sufficient, your notion might be Δ^1_1 randomness, surprisingly proposed by Martin-Löf in [23] as "the" formal randomness notion. A Δ^1_1 class is a sort of effective Borel class, and a set is Δ^1_1 random if it is in no Δ^1_1 class of measure 0. Martin-Löf's main result in that short paper states that there is no universal test in this sense (also see [33, after 9.3.5]). The strongest effective notion is Π^1_1 randomness, studied in [15]. All null Π^1_1 classes are now tests. There is a largest one ([16], or see [15] for a direct proof). Interestingly, this notion is relevant in effective model theory: if a countable structure has with probability 1 a presentation computable in an oracle, then it already has a presentation computable in each Π^1_1 random oracle. (This is due to Kalimullin and Nies, slightly extending work of Greenberg, Montalban, and Slaman. See the the March 2010 entry in the Logic Blog on my web site.)

To summarize, the intuitive idea of randomness for sets corresponds to a whole hierarchy of formal notions. We mentioned most of the main notions:

Π^1_1-random $\Rightarrow$ Δ^1_1-random $\Rightarrow$
2-random $\Rightarrow$ weak 2-random $\Rightarrow$ ML-random $\Rightarrow$ Schnorr random.

For three notions, there is a universal test. Do you know which ones they are?

2.2 The computational complexity aspect of a set

In contrast to randomness, we have a clear intuition of what a computable set (or function) is. The Church-Turing thesis states that this intutive notion has a clear-cut formal counterpart, the sets computable by a Turing machine. If we search our mind for an intution on the complexity of *in*computable sets, things become less clear. Perhaps there is an intuition what it means to be "close to computable". However, the formal notions that have been proposed, the so-called lowness properties already mentioned in Section 1, are rather disparate. They can even exclude each other outside the computable sets. For instance, a set A is called *computably dominated* if each function that can be computed with A as an oracle is dominated by a computable function. The only computably dominated sets that are Turing below the halting problem are the computable sets. A diagram of 12 lowness properties is given on page 361 of [33].

2.3 Using computability to understand randomness

In the beginning of this section I explained how computability theoretic tools are used to introduce formal randomness notions. Once defined formally, one can also *study* randomness, or its absence, via computability. Let A be a set of natural numbers. There are theorems supporting each of the following principles.

(1) A is far from random $\Leftrightarrow$ A is close to computable.

(2) Suppose A already has a certain randomness property. Then
A is more random $\Leftrightarrow$ A is closer to computable.

I will now give mathematical evidence for both principles. At present the main evidence for (1) is the following.

Theorem 2.1 *A is K-trivial $\Leftrightarrow$ A is low for Martin-Löf randomness.*

After the China visit at the end of 2001, I went on to Thailand, and then took a plane to the US to work with Richard Shore at Cornell. Now, at the beginning of 2002, I found myself travelling by bus, starting from the South of Mexico, through all the countries of Central America. I ended up on a small island called Isla Grande off the North coast of Panama, where I began working on the question posed in [20] whether each set that is low for ML-randomness is Δ^0_2. Eventually, I was able to obtain an affirmative answer. Having left Isla Grande, I wrote up a 7 page paper with this result in the internet cafés of Panama City and submitted it to the 2002 FOCS conference, where it was promptly rejected. See [28] for this proof. Later on, I improved the methods to obtain the implication "$\Leftarrow$" of Theorem 2.1. In 2003 I obtained an even stronger result involving computable randomness, which concludes this line of argument (if each set that is ML-random is still computably random in A, then A is K-trivial). The result appeared in [29]; see the unpopular Theorem 8.3.10 in my book [33].

I hitched a ride on a yacht from Ciudád Colon in Panama to Isla Mujeres in Mexico. I spent two weeks in the beautiful ocean off the Eastern Coast of Central America. My company consisted of two elderly gentlemen who hated each other, one paranoid cat, and an adventurer from the US who had the project to hide his savings in a bank on the Caribbean Island of St. Thomas because of a paternity lawsuit that awaited him at home. These were the conditions under which I started thinking about the implication "$\Rightarrow$" of Theorem 2.1. After a stop at Playa Tulum I went on to visit friends in Jalapa, the capital of the Mexican state of Veracruz. Staying there for a month, I got closer and closer to proving that remaining implication, without believing it was true at that time. I started from the decanter proof in [10] that each K-trivial is Turing incomplete. As an intermediate result, I proved that each c.e. K trivial has a lowness property called c.e. traceability, which for c.e. sets is equivalent to being array recursive. Next, I showed that the K-trivial are closed downward under Turing reducibility (which is not at all clear from the definition). Given a K-trivial set A and a set B that is Turing below A, I built a prefix-free machine showing that B is K-trivial. Unlike the previous decanter proof, this construction has a tree of runs of decanters. There must be a "golden run", namely a run that does not return while all the runs it calls do return. At the golden run node the required object is constructed, in this case, a prefix-free machine showing that B is K-trivial.

From Jalapa I went to Chicago to meet Denis Hirschfeldt. He realized that the golden run method shows the stronger result that each K-trivial set A is low for K: using A as an oracle does not yield shorter prefix-free descriptions of strings. This property, introduced by Muchnik Jr. in 1999, easily implies being low for ML-randomness. To define it formally, A is low for K if there is a constant d such that $K^A(y) \geq K(y) - d$ for each string y. Interestingly, we cannot find the golden run node in this construction, we can only prove that it exists. This is necessarily so: there is no effective way to obtain, from an index for the c.e. set A and the constant b for K-triviality of A, the constant d via which A is low for K [33, 5.5.5]. (In the construction, only the double jump $\emptyset''$ can find this golden run.)

Next I will discuss the principle (2) above: if A already has a randomness property, then

$$A \text{ is more random} \Leftrightarrow A \text{ is closer to computable.}$$

This almost seems to contradict the principle (1), but note that (1) is about sets that are far from random, while (2) is about sets that are already (somewhat) random. The implication "$\Leftarrow$" has been called *randomness enhancement*: satisfying a lowness property enhances the degree of randomness of A [27]. There are numerous instances of the principle (2).

- Randomness properties stronger than ML-randomness are usually closed downwards under $\leq_T$ within the ML-random sets, so they are given by an "abstract" lowness property. Further, if A is ML-random, then A is weakly 2-random $\Leftrightarrow$ every Δ^0_2 set below A is computable (Hirschfeldt and Miller; see [33, p. 135]), and A is 2-random $\Leftrightarrow$ Ω is ML-random in A [32].

- Let Z be Schnorr random set. If Z it is not high, then it is already ML-random. If Z is even computably dominated, then in fact Z is weakly 2-random [32].
- If A is Δ^1_1 random, then A is Π^1_1 random $\Leftrightarrow$ each function f that is hyperarithmetical in A is dominated by a hyperarithmetical function (Kjos-Hanssen, Nies, Stephan and Yu [17]; also see [33, 9.4.6]).

2.4 Using randomness to understand computability

Computability theory is all about the computational complexity of sets of natural numbers. One can gauge the complexity of a set A by locating A in classes of sets that all have a similar complexity. Examples of such classes are the computable sets, the high sets (i.e., $\emptyset'' \leq_T A'$), the Δ^0_2 sets (i.e., A is Turing below the halting problem), and the ω-c.e. sets (A is Turing below the halting problem, and, in addition, the reduction has a computably bounded use). The Limit Lemma of Shoenfield says that a set A is $\Delta^0_2 \Leftrightarrow$ the bit values $A(x)$ can be computably approximated with a finite number of mind changes; A is ω-c.e. $\Leftrightarrow$ in addition, the number of mind changes is computably bounded. Randomness-related concepts can be used both to introduce, and to study classes of similar complexity.

Mostly the classes of similar complexity are lowness properties. A common paradigm for lowness is the *weak-as-an-oracle* paradigm: A is weak in a specific sense when used as an oracle set in a Turing machine computation. Via randomness-related concepts, two new lowness paradigms have emerged [33, 27, 13].

The *Turing-below-many* paradigm says that A is close to computable because A is easy to obtain from an oracle set, in the sense that the class of oracles computing A is large. Here, a class of oracles is considered large if it contains random sets of a certain kind. So far, all sets satisfying an instance of the Turing-below-many paradigm are Δ^0_2 sets.

The *inertness* paradigm says that a set A is close to computable because it is computably approximable with a small number of mind changes. In particular, such a set is Δ^0_2 by the Limit Lemma. For a mathematical formulation of the inertness paradigm, one can use so-called *cost functions*. A cost function $c(x, s)$ is a computable function defined on pairs of natural numbers x, s. The values $c(x, s)$ are non-negative rationals. Cost functions are used to bound the total quantity of changes of a Δ^0_2 set, and especially that of a computably enumerable set. At a stage s, if x is least such that I change my guess at $A(x)$, then I have to pay $c(x, s)$. To achieve lowness, my goal is to build a set A that *obeys* c in the sense that the total cost of changes is finite.

K-triviality has been characterized via the inertness paradigm [29]. Let $c_\Omega(x, s)$ be the measure of descriptions entering the domain of the universal prefix-free machine between stages x and s; thus, $c_\Omega(x, s) = \Omega_s - \Omega_x$. The single cost function c_Ω does the job: In [30] it is shown that A is K-trivial $\Leftrightarrow$

some computable approximation of A obeys c_Ω. Just like c_Ω, most of the other examples of cost functions are based on randomness-related concepts.

A further lowness property of a set is *strong jump traceability*. It was discovered when Santiago Figueira visited Auckland for 3 months in 2003 [12]. Cholak, Downey and Greeberg [7] showed that the c.e. strongly jump traceable sets form a proper subclass of the c.e. K-trivials. The class has now been characterized via all three lowness paradigms. The original definition is by the weak-as-an-oracle paradigm, expressing that the jump $J^A(x)$ has very few possible values (it is equivalent to require [12] that the relative Kolmogorov complexity $C^A(y)$ of a string y is not far below $C(y)$, which makes the notion an analog of being low for K). The characterization via the Turing-below-many paradigm says that A is Turing below each ML-random set Z that is ω-c.e. If A itself is ω-c.e., this also expresses being far from random. For, there are many ω-c.e. ML-random sets: besides Ω, we have the ones obtained via the Low Basis Theorem. They can even be ML-random relative to one other. If an ω-c.e. A is "known" to all of them, it must be far from random itself. (Are you convinced by this argument?) The characterization via the inertness paradigm says that a c.e. set A is strongly jump traceable $\Leftrightarrow$ A obeys all so-called benign cost functions (for Δ^0_2 sets in general, at present only "$\Leftarrow$" is known).

The Turing-below-many paradigm seems to be more powerful than the weak-as-an-oracle paradigm, because it allows us to get closer to being computable. It even brings to light proper subclasses of the strongly jump traceable sets, for instance the sets Turing below all ω^2-c.e. ML-random sets [31].

3 What don't we know (yet)?

We hope that new developments add to the present body of knowledge, and that, in the worst case, they supersede known results. However, new developments may also render previous results irrelevant.

The field of computability and randomness has now reached a state of "early maturity". Some notions, and results involving them, are generally agreed to be fundamental, for instance Martin-Löf randomness and K-triviality. For other notions, time will show. Let me assess the present knowledge critically.

3.1 Are we studying the right randomness notions?

There are various criteria for a good notion. The criteria (b)-(d) below work for all of mathematics, while criterion (a) only applies to some of it.

(a) *The notion corresponds to some intuitive idea.* This is true for most of the randomness notions of Subsection 2.1, and also for computable randomness, which is between ML-randomness and Schnorr randomness. Usually these notions formalize at least one of the three intuitive "randomness paradigms" introduced in [8] (typicalness, unpredictability, incompressibility of initial segments).

(b) *There are natural examples, or constructions, of instances of the notion.* This is true for ML-randomness and 2-randomness, where the examples are Ω, and Ω relative to $\emptyset'$, respectively.

(c) *The notion interacts richly with other sub-areas.* Again, this is true for several randomness notions of Subsection 2.1, for instance because of their interaction with computability.

(d) *The notion is a "sink".* That is, one reaches the same notion from different directions. To support this, there are coincidence results for Schnorr randomness, ML-randomness, and 2-randomness. Originally defined via tests, these three notions can be characterized via incompressibility of initial segments. Computable randomness can also be characterized in different ways: by definition, Z is computably random if no computable betting strategy wins on Z. However, recent research of Brattka, Miller and myself [2] shows that it is equivalent to require that each non-decreasing computable function defined on the unit interval is differentiable at (the real corresponding to) Z.

Of course, there may be undiscovered randomness notions that perform just as well with these criteria.

3.2 Do the randomness notions really form a hierarchy?

For the notions presently known, the answer is "yes, mostly". One notable exception is Demuth randomness (see Section 3.6 of [33]), which is between 2-randomness and ML-randomness, but is incomparable with weak 2-randomness. This notion is interesting because, unlike weak 2-randomness, it is compatible with being Δ^0_2. It interacts strongly with lowness properties. For instance, each c.e. set A below a Demuth random set Y is strongly jump traceable [18]. If Y is Δ^0_2 then such a set A can be incomputable. Polynomial randomness is incomparable with Schnorr randomness, because the proof of [32] that some Schnorr random is not computably random actually produces a Schnorr random that is not polynomially random (also see [33, Thm. 7.5.10]).

If the answer to the hierarchy question eventually becomes "no", it would be harder to claim that these notions have anything to do with our intuition of randomness.

3.3 Are we studying the right lowness properties?

Let us apply the criteria for good notions (a)-(d) in Subsection 3.1. Many lowness properties perform quite well in (b)-(d). As for (a), each of them catches a bit of our intuition of being "close to computable", but none of them formalizes an intuitive idea just by itself. I will check the criteria (b)-(d) for two lowness properties: lowness for ML-randomness, and superlowness.

Lowness for ML-randomness = K-triviality (b) has a natural construction: a c.e. set obeying the cost function c_Ω (see Subsection 2.4). It (c) interacts well with randomness, and (d) coincides with heaps of other classes, such as being low for K, being a base for ML-randomness, and being low for weak 2-randomness (see [33, Section 5]).

We say that a set A is superlow if A' is truth-table below the halting problem $\emptyset'$. Superlow c.e. sets can be built via finite injury. Non-c.e. superlow sets with interesting properties, such as being ML-random or PA-complete, can be built via the low basis theorem. So (b) is satisfied. For c.e. sets, superlowness is equivalent to a property called jump traceability (see [33, 8.4.23]), which gives us (d).

3.4 So, again, are our notions intrinsic, or accidental?

They are the former, hopefully. The randomness and lowness notions would seem less accidental if they were introduced in different ways. For instance, they could be specializations of more general formal notions. To obtain these more general notions one could try to formalize the randomness and the lowness paradigms discussed earlier on, which so far are informal meta-notions.

To be even more heretical, we could ask whether the whole distinction between the randomness and the computational complexity aspect of sets is more than a historical accident, caused by the fact that people with different backgrounds were working at different times and in different places. Currently we develop these two aspects separately and then find interactions. Perhaps one day they will be unified into a single theory. Perhaps one day there will only be a general theory of access to the information of a set of natural numbers.

4 What are the most important open problems in the field?

I expect there will be many interesting new problems in areas that are just being developed, in particular the interaction of algorithmic randomness with computable analysis [2], and ergodic theory. Instead, I will discuss two major problems on randomness or its interaction with computability that have been around for a while.

4.1 Covering a K-trivial by an incomplete random

Kučera [19] built an incomputable but computably enumerable set A below any given Δ^0_2 ML-random set Z. If Z does not compute the halting problem (for instance, Z is the bits of Ω in even positions), this yields an injury-free solution to Post's problem [34] whether some incomputable c.e. set is Turing incomplete.

The cost function construction of a K-trivial set yields a further injury-free solution to Post's problem ([10], or see [33, Section 5.3]). If Z is a Martin-Löf random set that does not compute the halting problem, then every c.e. set A Turing below Z is K-trivial [14]. Thus, in a sense, Kučera's solution to Post's problem is a special case of the solution via building a K-trivial. It is open whether the converse holds: given A, build Z. Essentially we are asking whether the two injury-free solutions to Post's problem are equivalent. Since every K-trivial set is below a c.e. K-trivial [29], we may omit the hypothesis that A is computably enumerable.

Question 4.1 *[14, 25, 8] Is every K-trivial set Turing below an incomplete Martin-Löf random set?*

Countless people have worked on this. There is no consensus which way the answer will go.

There are several variants of Question 4.1. For instance, we say that A is *ML-noncuppable* if $A \oplus Z \geq_T \emptyset'$ implies $Z \geq_T \emptyset'$ for ML-random Z. Every c.e. ML-noncuppable is K-trivial (see [33, 8.5.15]).

Question 4.2 *[25] Is every K-trivial set ML-noncuppable?*

For background on the next variant of Question 4.1, see Subsection 3.2.

Question 4.3 *[18] Is every strongly jump traceable (c.e.) set Turing below a Demuth random?*

4.2 Kolmogorov-Loveland randomness

An infinite sequence of bits (i.e., a set) is computably random if no computable betting strategies succeeds on it. Such a strategy places a bet on the next bit position in the usual ascending fashion. We say that a set Z is *Kolmogorov-Loveland random* (KL-random) if no computable betting strategy succeeds even when it is allowed to choose the next bit position on which it places a bet. The implications are

Martin-Löf random $\Rightarrow$ KL-random $\Rightarrow$ computably rd. $\Rightarrow$ Schnorr random.

All implications except the leftmost one are known to be strict. The strictness of that implication is a major open question.

Question 4.4 *[26, 25]*
Does KL-randomness differ from Martin-Löf randomness?

A negative answer would defeat Schnorr's critique of ML-randomness, because KL-randomness is defined using a computable test concept. In [24] we obtained various results showing that KL-randomness is, at the very least, much closer to Martin-Löf-randomness than the other notions. For instance, the computable dimension of a KL-random set is 1.

Lowness for KL-randomness implies K-triviality [29]. Separating the two lowness properties would basically give an affirmative answer to Question 4.4. The following would be interesting to begin with.

Question 4.5 *Is some incomputable set low for KL-randomness?*

Recall that co-infinite sets can be identified with reals in the unit interval via the representation in base 2. When viewed as notions about reals, computable randomness and KL-randomness appear to rely on the representation of the real in base 2. However, strategies that bet on rational intervals can be used to show the base invariance for computable randomness. We do not know whether KL-randomness of a real actually depends on the choice of the base 2.

Question 4.6 *Is KL-randomness of a real number base-invariant?*

This is probably just as hard to answer as Question 4.4. Most likely, for each base, KL-randomness induces a distinct class of reals; these notions are all incomparable, and therefore all different from Martin-Löf randomness on reals.

5 What are the prospects for progress?

Back in the 16th century, a prospect was an extensive view of a landscape. Imagine you stand on a mountain top and see the immense area of computability and randomness stretched out before you. Where do you want to go?

Let's say you are a young researcher with some knowledge of the books [5, 33, 8], but not too much respect for the results in there. Then you might make progress on the new directions suggested in Section 3.

We would make progress on the open problems in Section 4 by convening a group of researchers, young and old, on that mountain top.

Acknowledgments. I thank Christopher Porter for comments on earlier drafts of this paper, and the editor Hector Zenil for getting me to write this.

References

[1] G. Barmpalias and A. Nies. Upper bounds for ideals in the computably enumerable degrees. Submitted.

[2] V. Brattka, J. Miller, and A. Nies. Computable randomness and differentiability. To appear.

[3] C. Calude. *Information and randomness.* Monographs in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 1994. With forewords by Gregory J. Chaitin and Arto Salomaa.

[4] C. Calude and A. Nies. Chaitin Ω numbers and strong reducibilities. *J. UCS*, 3(11):1162–1166, 1997.

[5] Cristian S. Calude. *Information and randomness.* Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, second edition, 2002. With forewords by Gregory J. Chaitin and Arto Salomaa.

[6] G. Chaitin. A theory of program size formally identical to information theory. *J. Assoc. Comput. Mach.*, 22:329–340, 1975.

[7] P. Cholak, R. Downey, and N. Greenberg. Strongly jump-traceability I: the computably enumerable case. *Adv. in Math.*, 217:2045–2074, 2008.

[8] R. Downey and D. Hirschfeldt. *Algorithmic randomness and complexity.* Springer-Verlag, Berlin. To appear.

[9] R. Downey, D. Hirschfeldt, and A. Nies. Randomness, computability and density. *SIAM J. Computing*, 31:1169–1183, 2002.

[10] R. Downey, D. Hirschfeldt, A. Nies, and F. Stephan. Trivial reals. In *Proceedings of the 7th and 8th Asian Logic Conferences*, pages 103–131, Singapore, 2003. Singapore University Press.

[11] R. Downey, D. Hirschfeldt, A. Nies, and S. Terwijn. Calibrating randomness. *Bull. Symbolic Logic*, 12(3):411–491, 2006.

[12] S. Figueira, A. Nies, and F. Stephan. Lowness properties and approximations of the jump. *Ann. Pure Appl. Logic*, 152:51–66, 2008.

[13] N. Greenberg, D. Hirschfeldt, and A. Nies. Characterizing the strongly jump traceable sets via randomness. To appear.

[14] D. Hirschfeldt, A. Nies, and F. Stephan. Using random sets as oracles. *J. Lond. Math. Soc. (2)*, 75(3):610–622, 2007.

[15] G. Hjorth and A. Nies. Randomness via effective descriptive set theory. *J. London Math. Soc.*, 75(2):495–508, 2007.

[16] A. Kechris. The theory of countable analytical sets. *Trans. Amer. Math. Soc.*, 202:259–297, 1975.

[17] B. Kjos-Hanssen, A. Nies, F. Stephan, and A. Nies. Higher kurtz randomness. To appear.

[18] A. Kučera and A. Nies. Demuth randomness and computational complexity. To appear.

[19] A. Kučera. An alternative, priority-free, solution to Post's problem. In *Mathematical foundations of computer science, 1986 (Bratislava, 1986)*, volume 233 of *Lecture Notes in Comput. Sci.*, pages 493–500. Springer, Berlin, 1986.

[20] A. Kučera and S. Terwijn. Lowness for the class of random sets. *J. Symbolic Logic*, 64:1396–1402, 1999.

[21] S. Kurtz. *Randomness and genericity in the degrees of unsolvability.* Ph.D. Dissertation, University of Illinois, Urbana, 1981.

[22] P. Martin-Löf. The definition of random sequences. *Inform. and Control*, 9:602–619, 1966.

[23] Per Martin-Löf. On the notion of randomness. In *Intuitionism and Proof Theory (Proc. Conf., Buffalo, N.Y., 1968)*, pages 73–78. North-Holland, Amsterdam, 1970.

[24] W. Merkle, J. Miller, A. Nies, J. Reimann, and F. Stephan. Kolmogorov-Loveland randomness and stochasticity. *Ann. Pure Appl. Logic*, 138(1-3):183–210, 2006.

[25] J. Miller and A. Nies. Randomness and computability: Open questions. *Bull. Symbolic Logic*, 12(3):390–410, 2006.

[26] Andrei A. Muchnik, A. Semenov, and V. Uspensky. Mathematical metaphysics of randomness. *Theoret. Comput. Sci.*, 207(2):263–317, 1998.

[27] A. Nies. Applying randomness to computability. Series of three lectures at the ASL summer meeting, Sofia, 2009.

[28] A. Nies. Low for random sets: the story. Preprint, available at `http://www.cs.auckland.ac.nz/nies/papers/`, 2005.

[29] A. Nies. Lowness properties and randomness. *Adv. in Math.*, 197:274–305, 2005.

[30] A. Nies. Calculus of cost functions. To appear.

[31] A. Nies. Subclasses of the c.e. strongly jump traceable sets. To appear.

[32] A. Nies, F. Stephan, and S. Terwijn. Randomness, relativization and Turing degrees. *J. Symbolic Logic*, 70(2):515–535, 2005.

[33] André Nies. *Computability and randomness*, volume 51 of *Oxford Logic Guides*. Oxford University Press, Oxford, 2009.

[34] E. Post. Recursively enumerable sets of positive integers and their decision problems. *Bull. Amer. Math. Soc.*, 50:284–316, 1944.

[35] R. Solovay. Handwritten manuscript related to Chaitin's work. IBM Thomas J. Watson Research Center, Yorktown Heights, NY, 215 pages, 1975.

[36] R. von Mises. Grundlagen der Wahrscheinlichkeitsrechnung. *Math. Zeitschrift*, 5:52–99, 1919.

[37] D. Zambella. On sequences with simple initial segments. ILLC technical report ML 1990-05, Univ. Amsterdam, 1990.