

Johanna N.Y. Franklin

Schnorr trivial reals

A construction

Received: date / Revised: date

Abstract A real is Martin-Löf (Schnorr) random if it does not belong to any effectively presented null Σ_1^0 (recursive) class of reals. Although these randomness notions are very closely related, the set of Turing degrees containing reals that are K -trivial has very different properties from the set of Turing degrees that are Schnorr trivial. Nies proved in [12] that all K -trivial reals are low. In this paper, we prove that if $\mathbf{h}' \geq_T \mathbf{0}''$, then \mathbf{h} contains a Schnorr trivial real. Since this concept appears to separate computational complexity from computational strength, it suggests that Schnorr trivial reals should be considered in a structure other than the Turing degrees.

Keywords randomness · triviality · Schnorr trivial

Mathematics Subject Classification (2000) 03D15

1 Preliminaries

We will say that a real is an element of 2^ω and that a tree is a subset of $2^{<\omega}$ that is closed under substrings. The set of all infinite branches through a tree T will be represented by $[T]$, and the set of all $\tau \in 2^{<\omega}$ extending a binary string σ will be represented by $[\sigma]$.

Throughout this paper, μ will represent Lebesgue measure.

This material is based upon work supported under a National Science Foundation Graduate Research Fellowship. A preliminary version of this paper appeared in Electronic Notes in Theoretical Computer Science.

Johanna N.Y. Franklin
Department of Mathematics
National University of Singapore
2, Science Drive 2
Singapore 117543
SINGAPORE
E-mail: franklin@math.nus.edu.sg

Definition 1 We will define a Turing machine to be a recursive function from $2^{<\omega}$ to $2^{<\omega}$. A Turing machine M is said to be *prefix free* if all distinct σ and τ in $\text{dom}(M)$ are incomparable. A prefix-free Turing machine U is *universal* if for each prefix-free machine M , there is some $\tau \in 2^{<\omega}$ such that

$$(\forall \sigma \in 2^{<\omega})[U(\tau \hat{\ } \sigma) = M(\sigma)].$$

Such a machine can easily be constructed by taking an effective list $\langle M_i \rangle_{i \in \omega}$ of all prefix-free Turing machines and setting $U(1^i 0 \hat{\ } \sigma) = M_i(\sigma)$ [4].

We will only consider prefix-free Turing machines. Since we will often consider the measure of the domain of a Turing machine but never the range, we will write $\mu(M)$ for $\mu(\text{dom}(M))$. Note that for a prefix-free Turing machine M , if we list the elements of M as $\langle \tau_i, \sigma_i \rangle$, $\mu(M) = \sum \tau_i \frac{1}{2^{|\tau_i|}}$.

Definition 2 We say that a Turing machine M is *computable* if the measure of its domain is a recursive real.

We will use the same notation for Kolmogorov complexity as Downey et al. do in [4].

Definition 3 Let M be a Turing machine, and let $\sigma \in 2^{<\omega}$. The *prefix-free Kolmogorov complexity of σ with respect to M* is $K_M(\sigma) = \min\{|\tau| \mid M(\tau) = \sigma\}$.

When the machine is unspecified (for instance, when we write $K(\sigma)$), a universal Turing machine is being used. Note that the particular universal Turing machine is irrelevant up to an additive constant.

We will say that $\sigma \in M$ for $\sigma \in 2^{<\omega}$ and a Turing machine M when there is a $\tau \in 2^{<\omega}$ such that $\langle \tau, \sigma \rangle \in M$.

2 Martin-Löf randomness, lowness, and triviality

Before we begin analyzing the reals that are “far from Schnorr random,” we review what is known for the most heavily studied notion of randomness, Martin-Löf randomness.

Definition 4 A *Martin-Löf test* is a uniformly r.e. sequence $\langle V_i \rangle_{i \in \omega}$ of Σ_1^0 classes such that $\mu(V_i) \leq \frac{1}{2^i}$. A real A is *Martin-Löf random* if for all Martin-Löf tests $\langle V_i \rangle_{i \in \omega}$, $A \notin \bigcap_{i \in \omega} V_i$.

Martin-Löf proved that there is a universal Martin-Löf test; i.e., a Martin-Löf test $\langle U_i \rangle_{i \in \omega}$ such that a real A is Martin-Löf random if and only if $A \notin \bigcap_{i \in \omega} U_i$ [10]. Later, Schnorr demonstrated that Martin-Löf randomness can also be defined in terms of computational complexity [14].

Theorem 1 [14] *A is Martin-Löf random if and only if $(\exists c \in \omega)(\forall n \in \omega)[K(A \upharpoonright n) \geq n - c]$.*

It can be seen easily from the Martin-Löf test definition of Martin-Löf randomness that there are Π_1^0 classes that contain only Martin-Löf random reals. Therefore, we can apply the Low Basis Theorem and the Hyperimmune-free Basis Theorem to obtain a low Martin-Löf random real and a hyperimmune-free Martin-Löf random real, respectively. Among the results about the Turing degrees of Martin-Löf random reals is the following theorem due to Kučera.

Theorem 2 [9] *If $\mathbf{a} \geq \mathbf{0}'$, then \mathbf{a} contains a Martin-Löf random real.*

We now turn our attention to the reals that are “far from Martin-Löf random.” As with most types of randomness, we can develop a notion of relative Martin-Löf randomness. Such a reducibility can then be used to define a triviality notion.

Definition 5 [6] Let A and B be reals. We say that $A \leq_K B$ if $(\exists c \in \omega)(\forall n \in \omega)[K(A \upharpoonright n) \leq K(B \upharpoonright n) + c]$. We define A to be K -trivial if $A \leq_K \mathbf{0}^\omega$; i.e., if $(\exists c \in \omega)(\forall n \in \omega)[K(A \upharpoonright n) \leq K(\mathbf{0}^n) + c]$.

Another way in which a real A can be “far from Martin-Löf random” is if relativizing a definition of Martin-Löf randomness to A generates the same class of sets. There are two ways in which this has been done.

Definition 6 Let \mathcal{ML} be the class of Martin-Löf-random sets. A real A is *low for Martin-Löf randomness* if $\mathcal{ML}^A = \mathcal{ML}$.

Definition 7 A real A is *low for Martin-Löf tests* if for every Martin-Löf test relative to A $\langle U_i^A \rangle_{i \in \omega}$, there is a Martin-Löf test $\langle V_i \rangle_{i \in \omega}$ such that $\bigcap_{i \in \omega} U_i^A \subseteq \bigcap_{i \in \omega} V_i$.

We immediately note that the existence of a universal Martin-Löf test implies that a real is low for Martin-Löf randomness if and only if it is low for Martin-Löf tests. In fact, it has been shown by Nies that all three of these notions coincide for Martin-Löf randomness.

Theorem 3 [12] *If A is a real, then the following are equivalent.*

1. A is low for Martin-Löf randomness.
2. A is low for Martin-Löf tests.
3. A is K -trivial.

Chaitin was the first to show that all K -trivial reals are Δ_2^0 [2], and Downey, Hirschfeldt, Nies, and Stephan showed that such reals cannot be high [6]. Later, Nies proved the following theorem.

Theorem 4 [12] *The K -trivials form a nonprincipal Σ_3^0 ideal in the ω -r.e. Turing degrees. This ideal is generated by its r.e. members. Furthermore, each K -trivial A is such that $A' \equiv_{tt} \mathbf{0}'$ (A is superlow).*

The three notions described above have been generalized to randomness notions other than Martin-Löf randomness. We define low for \mathcal{R} , low for \mathcal{R} -tests, and \mathcal{R} -trivial for an arbitrary notion of randomness \mathcal{R} and then discuss their connections to each other.

Definition 8 A real A is *low for \mathcal{R}* if $\mathcal{R} = \mathcal{R}^A$.

Definition 9 A real A is *low for \mathcal{R} -tests* if for every \mathcal{R} -test relative to $A \langle U_i^A \rangle_{i \in \omega}$, there is a \mathcal{R} -test $\langle V_i \rangle_{i \in \omega}$ such that $\bigcap_{i \in \omega} U_i^A \subseteq \bigcap_{i \in \omega} V_i$.

Definition 10 A real A is \mathcal{R} -trivial if $A \leq_{\mathcal{R}} 0^\omega$, where $\leq_{\mathcal{R}}$ is \mathcal{R} 's notion of relative initial-segment complexity.

Clearly, if A is low for \mathcal{R} -tests, it is low for \mathcal{R} . However, it is not obvious that either lowness notion is related to triviality.

3 Schnorr randomness, lowness, and triviality — Previous results

Schnorr randomness is a more effective version of Martin-Löf randomness. A Schnorr test is simply an effectively given Martin-Löf test.

Definition 11 A Martin-Löf test is a *Schnorr test* if $\mu(V_i) = \frac{1}{2^i}$ for all i . A real A is *Schnorr random* if for all Schnorr tests $\langle V_i \rangle_{i \in \omega}$, $A \notin \bigcap_{i \in \omega} V_i$.

There is no universal Schnorr test. Therefore, we cannot use a universal Turing machine to define Schnorr randomness in terms of computational complexity. Instead, we must quantify over all computable Turing machines.

Theorem 5 [5] *A real A is Schnorr random if and only if $(\forall M \text{ comp.})(\exists c \in \omega)(\forall n \in \omega)[K_M(A \upharpoonright n) \geq n - c]$.*

It is clear that all Martin-Löf-random reals are also Schnorr random, but this implication is not reversible. In fact, Nies, Stephan, and Terwijn showed that these concepts are separable in the high degrees [13].

Downey, Griffiths, and Laforte developed the following characterization of Schnorr triviality in [3]. They began by defining a notion of Schnorr reducibility.

Definition 12 [3] We say that $A \leq_{Sch} B$ if for every computable Turing machine M , there is a computable Turing machine M' and a constant $c \in \omega$ such that $(\forall n \in \omega)[K_{M'}(A \upharpoonright n) \leq K_M(B \upharpoonright n) + c]$. Therefore, a real A is *Schnorr trivial* ($A \leq_{Sch} 0^\omega$) if the following statement holds.

$$(\forall M \text{ comp.})(\exists M' \text{ comp.})(\exists c \in \omega)(\forall n \in \omega)[K_{M'}(A \upharpoonright n) \leq K_M(0^n) + c]$$

Downey, Griffiths, and Laforte have proved that there is a Turing complete Schnorr trivial real, but that there is an r.e. degree that contains no Schnorr trivial reals [3]. While this shows that the Schnorr trivial Turing degrees are not downward closed, they also proved that the Schnorr trivial *tt*-degrees are downward closed [3].

The work done with reals that are low for Schnorr to date has produced an entirely degree-theoretic characterization. Here, we let D_n denote the n th canonical finite set.

Definition 13 A set A is *recursively traceable* if there is a recursive, increasing, unbounded function $p : \omega \rightarrow \omega$ as follows.

$$(\forall f \leq_T A)(\exists r : \omega \rightarrow \omega \text{ rec.})(\forall n \in \omega)[f(n) \in D_{r(n)} \text{ and } |D_{r(n)}| \leq p(n)]$$

In short, a real is recursively traceable if it is uniformly hyperimmune free. We say that r is a *recursive trace* and that p is a *bound for a recursive trace for every* $f \leq_T A$.

The first result on reals that are low for Schnorr tests comes from Terwijn and Zambella [15]. Later, Kjos-Hanssen, Nies, and Stephan used a similar technique to demonstrate that, given Terwijn and Zambella's result, the reals that are low for Schnorr tests are precisely the reals that are low for Schnorr [8].

Theorem 6 [15] *A set is recursively traceable if and only if it is low for Schnorr tests.*

Theorem 7 [8] *A set is recursively traceable if and only if it is low for Schnorr randomness.*

Although the reals that are low for K are precisely those that are K -trivial, being low for Schnorr is not equivalent to being Schnorr trivial. All reals that are low for Schnorr are hyperimmune free, and there is a Turing complete Schnorr trivial. This Schnorr trivial is clearly not hyperimmune free and is thus not Schnorr low. The best that can be hoped for is that all Schnorr lows are Schnorr trivial. In another publication, we will show that this is, in fact, the case [7].

In [15], Terwijn and Zambella also demonstrated that there is a perfect set of Schnorr lows by showing that Miller and Martin's construction in [11] produces only recursively traceable sets.

4 Schnorr trivial reals: The basic construction

Theorem 8 *Let \mathbf{h} be a Turing degree such that $\mathbf{h}' \geq_T \mathbf{0}'$. Then \mathbf{h} contains a Schnorr trivial.*

The proof of this theorem requires, for each such degree \mathbf{h} , the construction of a perfect binary tree T recursive in \mathbf{h} such that every branch of T is Schnorr trivial and every $A \in 2^{<\omega}$ is encoded by some branch of T .

We note that Terwijn and Zambella's perfect set of Schnorr lows is also a perfect set of Schnorr trivials. However, since it consists entirely of hyperimmune-free degrees, a new construction is necessary to prove Theorem 8.

Theorem 9 *There is a perfect set of Schnorr trivials.*

The following theorem will be necessary for the proof.

Theorem 10 (Kraft-Chaitin Theorem [1]) *Let $\langle d_i, \sigma_i \rangle_{i \in \omega}$ be a recursive sequence with $d_i \in \omega$ and $\sigma_i \in 2^{<\omega}$ for all i such that $\sum_i \frac{1}{2^{d_i}} \leq 1$. (Such a sequence is called a Kraft-Chaitin set, and each element of the sequence is called a Kraft-Chaitin axiom.) Then there are strings τ_i and a prefix-free machine M such that $\text{dom}(M) = \{\tau_i \mid i \in \omega\}$ and for all i and j in ω ,*

1. *if $i \neq j$, then $\tau_i \neq \tau_j$,*
2. *$|\tau_i| = d_i$,*
3. *and $M(\tau_i) = \sigma_i$.*

The Kraft-Chaitin Theorem allows us to construct a prefix-free machine by specifying only the lengths of the strings in the domain rather than the strings themselves. We will therefore identify $\langle \tau, \sigma \rangle$ with $\langle d, \sigma \rangle$ where $d = |\tau|$ throughout.

Within the proof, we will consider an arbitrary computable machine. We introduce the following terminology to simplify the construction.

Definition 14 Let M be a computable machine. We say that a real A is *Schnorr trivial with respect to M* if the following condition holds.

$$(\exists M' \text{ comp.})(\exists c \in \omega)(\forall n \in \omega)[K_{M'}(A \upharpoonright n) \leq K_M(0^n) + c]$$

As demonstrated by Downey and Griffiths [5], we may consider only computable machines with domain 1. In fact, it should be noted that for every computable machine, there are infinitely many equivalent computable machines with domain 1.

Proof (Theorem 9) We build a perfect tree of Schnorr trivials by constructing a sequence of recursive trees $T_i \supseteq T_{i+1}$ such that each branch of T_i is Schnorr trivial with respect to the computable machine M_i for a given listing $\langle M_i \rangle_{i \in \omega}$ of computable machines with domain 1.

We first illustrate the construction of the tree by describing the construction for a single component. Let M be a computable machine with domain 1. We wish to build a perfect tree T such that every branch B of T is Schnorr trivial with respect to M .

Remark 1 Throughout the construction, note that, while we are building our T within $2^{<\omega}$, we could easily build it within any recursive tree.

We begin by dividing M 's domain into pieces of measure approximately $\frac{1}{2^k}$ for $k \geq 1$. At stage k , we increase the height of the components of our tree up to the height of the longest string in M appearing by the first stage at which $\mu(M) \geq 1 - \frac{1}{2^k}$. We then assign appropriate complexities to all the strings in our tree of the appropriate heights to build another computable machine M' which demonstrates the Schnorr triviality of all branches of T with respect to M thus far. Then we insert a branching point in the tree in such a way as to ensure that the tree is perfect.

We will distinguish between the stages in the enumeration of the machine M we are using and the stages in our own construction of the tree T and the machine M' . Each stage k in our construction will correspond to the stages $s_{k-1} < s \leq s_k$ in the enumeration of M . We will refer to the latter as M -stages and to the former simply as stages.

We define two sequences of natural numbers that are recursive in M and thus simply recursive. The first is $\langle s_k \rangle_{k \in \omega}$, where s_k is the least M -stage s such that $\mu(M_s) \geq 1 - \frac{1}{2^k}$. The second is $\langle n_k \rangle_{k \in \omega}$, where $n_k = \max\{|\sigma| \mid \sigma \in M_{s_k}\}$.

$k = 0$: Clearly, $s_0 = 0$ and $n_0 = 0$. We define $M'_0 = \emptyset$ and $T_0 = 2^{<\omega}$.

$k = 1$: We consider each s for $s_0 < s \leq s_1$. If an axiom $\langle d, \tau \rangle$ enters M at M -stage s , we add $\langle d + 1, 1^{|\tau|} \rangle$ to M' . Note that in particular, whenever an axiom of the form $\langle d, 0^n \rangle$ enters M , the pair $\langle d + 1, 1^n \rangle$ enters M' . After the M -stage s_1 , we

define M'_1 to be the set of Kraft-Chaitin axioms enumerated into M' thus far. Now we let σ be the string 1^{n_1} and define $T_1 = [\sigma \hat{\ } 0] \cup [\sigma \hat{\ } 1]$.

At this point, we have completely determined the tree up to height $n_1 + 1$, and we have two branches at this height.

$k > 1$: We consider each s for $s_{k-1} < s \leq s_k$. We let $\Sigma = \{\sigma_1, \dots, \sigma_k\}$ be the set of the longest elements of T_{k-1} that have been completely determined. We will call their length r . There are precisely k , as we create one new branching point at each stage. Suppose $\langle d, \tau \rangle$ enters M at M -stage s and define $n = |\tau|$.

Case 1: $n \leq r$. For all $\sigma \in T_{k-1}$ such that $|\sigma| = n$, we add $\langle d+1, \sigma \rangle$ to M' at M -stage s . There will be l many such σ for some $l \leq k$, so we add $\langle d+1, \sigma' \rangle$ to M' an additional $k-l$ times, where σ' is the leftmost $\sigma \in T_{k-1}$ such that $|\sigma| = n$. Note that we have added $\frac{l}{2^{d+1}} + \frac{k-l}{2^{d+1}} = \frac{k}{2^{d+1}}$ to $\mu(M')$.

Case 2: $n > r$. For all $\sigma_i \in \Sigma$, we add $\langle d+1, \sigma_i \hat{\ } 1^{n-r} \rangle$ to M' at M -stage s . This time there will be k many such σ_i , so we have added $\frac{k}{2^{d+1}}$ to $\mu(M')$.

This gives us M'_k . For each successive interval of $\frac{1}{2^k}$ in $\mu(M)$, we add no more than $\frac{k}{2^{k+1}}$ to $\mu(M')$. We may add less because our last step may have increased the measure of M by slightly more than $\frac{1}{2^{k-1}}$, leaving less for us to use at this step.

Now we define T_k as follows. Let $r' = \max\{r, n_k\}$.

$$T_k = [\sigma_\varphi \hat{\ } 0] \cup \cup_{\sigma \in M'_k \text{ and } |\sigma| = r'} [\sigma \hat{\ } 1]$$

where σ_φ is the unique σ such that $|\sigma| = r'$, $\sigma \in M'_k$, and $\varphi(\sigma)$ holds, where $\varphi(\sigma)$ is true if and only if σ extends the branch immediately to the left of the branch extended with a 0 at the last stage, or, if there is no such branch, the rightmost branch. Note that since σ_φ is unique, the number of branches increases by one in every stage. This condition also ensures that the tree is perfect.

We can see that for all $k \in \omega$, $T_{k+1} \subseteq T_k$ and $M'_k \subseteq M'_{k+1}$, so we can define $T = \bigcap_{k \in \omega} T_k$ and $M' = \bigcup_{k \in \omega} M'_k$. T is clearly a recursive perfect tree.

Lemma 1 M' is a Kraft-Chaitin set.

Proof We first note that, since our construction is recursive, the elements of M' form an r.e. set.

Now we must show that $\mu(M') \leq 1$. We know that $\mu(M) = 1$. During stage k , if $\langle d, \sigma \rangle$ enters M , $\langle d+1, \tau \rangle$ enters M' for k many τ . Then, when $\frac{1}{2^k}$ enters $\mu(M)$, no more than $\frac{k}{2^{k+1}}$ enters $\mu(M')$. It may be slightly less than $\frac{k}{2^{k+1}}$ if $\mu(M_{s_{k-1}}) > 1 - \frac{1}{2^{k-1}}$. In this case, $\mu(M_{s_{k-1}}) - \frac{1}{2^{k-1}}$, and we can see that $\frac{k-1}{2}(\mu(M_{s_{k-1}}) - \frac{1}{2^{k-1}})$ will enter $\mu(M')$ instead of $\frac{k}{2}(\mu(M_{s_{k-1}}) - \frac{1}{2^{k-1}})$. Therefore, $\mu(M') \leq \sum_{k \in \omega} \frac{k}{2^{k+1}} = 1$.

Lemma 2 $\mu(M')$ is a recursive real.

Proof During each stage k , approximately $\frac{1}{2^k}$ enters $\mu(M)$. As mentioned in the proof of Lemma 1, we may add less if $\mu(M_{s_{k-1}})$ is strictly greater than $1 - \frac{1}{2^{k-1}}$. However, we can always calculate the amount of measure that has entered M by

the end of stage k recursively. Since we add $\frac{k}{2^{d+1}}$ to M' every time an axiom of the form $\langle d, \tau \rangle$ enters M , we can make a similar calculation for M' .

After stage k , no more than $\frac{1}{2^k}$ will enter M , so no more than $\sum_{j>k} \frac{j}{2^{j+1}}$ can be added to $\text{dom}(M')$. We can therefore use the inequality

$$|\mu(M') - \mu(M'_k)| \leq \sum_{j>k} \frac{j}{2^{j+1}}$$

to see that $\mu(M')$ is a recursive real.

Therefore, by Lemmas 1 and 2 and the Kraft-Chaitin Theorem, we may take M' to be a computable Turing machine.

Lemma 3 M' witnesses the Schnorr triviality of each branch of T with respect to M .

Proof Let $B \in [T]$, and let $n \in \omega$. If $K_M(0^n) = \infty$, we automatically have $K_{M'}(B \upharpoonright n) \leq K_M(0^n) + 1$. Otherwise, suppose that d is the least integer such that $\langle d, 0^n \rangle \in M$, so $d = K_M(0^n)$. Our construction guarantees that $\langle d+1, \sigma \rangle \in M'$ for each σ of length n in our tree T , including $B \upharpoonright n$. Therefore, $K_{M'}(B \upharpoonright n) \leq d+1 = K_M(0^n) + 1$. This is enough to show that the machine M' and the constant 1 witness the Schnorr triviality of B with respect to M .

Now that we have produced a perfect tree with all of its branches Schnorr trivial with respect to a single computable machine M , we produce a perfect tree with all of its branches Schnorr trivial (with respect to all computable machines) using our listing $\langle M_i \rangle_{i \in \omega}$ of computable machines with domain 1. For each $i \in \omega$, we build a tree T_i and a machine M'_i such that $T_i \supseteq T_{i+1}$ and $(\forall A \in [T_i])(\forall n \in \omega)[K_{M'_i}(A \upharpoonright n) \leq K_{M_i}(0^n) + (i+1)]$. Once again, we define two sequences recursive in M_i for each i . The first is $\langle s_{i,k} \rangle_{k \in \omega}$, where $s_{i,k}$ is the least stage s such that $\mu(M_{i,s}) \geq 1 - \frac{1}{2^k}$. The second is $\langle n_{i,k} \rangle_{k \in \omega}$, where $n_{i,k} = \max\{|\sigma| \mid \sigma \in M_{s_{i,k}}\}$. Once again, both of these sequences are recursive.

We build T_0 and M'_0 as previously. For $i \geq 1$, we build the tree T_i within T_{i-1} . As mentioned in Remark 1, this is permissible since T_{i-1} is recursive. This, however, will require some slight modifications to the constructions of T_i and M'_i .

Since we are working within T_{i-1} instead of $2^{<\omega}$, there are two factors that we must take into account. First, there may not be a branching point in T_{i-1} at precisely the height required by an $n_{i,k}$. To compensate for this, we generate a new recursive sequence $\langle n'_{i,k} \rangle_{k \in \omega}$, where $n'_{i,k}$ is the height of the lowest branching point in T_{i-1} above $n_{i,k}$, and we use this sequence rather than $\langle n_{i,k} \rangle_{k \in \omega}$.

Second, we fix the tree through the first i branching points of T_{i-1} to ensure that the resulting tree is perfect. Until $n'_{i,k}$ is greater than the height of the i th branching point in T_{i-1} plus 1, we treat it as in Case 1 of the $k > 1$ case to ensure that the branching points do not disappear when we take the intersection of the T_i s in the end. This means that we may have $k+i$ branching points at the end of stage $s_{i,k}$ instead of k , so instead of adding $\langle d+1, \sigma \rangle$ to $M'_{s_{i,k}}$ for each appropriate σ , we add $\langle d+i+1, \sigma \rangle$. This ensures that the measure of the machine M'_i will still be less than or equal to 1, since $\sum_{k \in \omega} \frac{k+i}{2^{k+i+1}} \leq 1$ for any i .

This requires other changes in the construction of the T_i s as well. Whereas before we could assume there were k branching points in our tree after stage k , this time we cannot assume that there will be exactly $k+i$ after stage k . This depends entirely on T_{i-1} and the sequence $\langle n_{i,k} \rangle_{k \in \omega}$. Therefore, the set of the longest elements already determined, Σ , may contain $m \leq k+i$ elements. To ensure that $\mu(M'_i)$ is still a recursive real, we add $(k+i) - m$ additional copies of $\langle d+i+1, \sigma' \rangle$ to M'_i , where σ' is the leftmost $\sigma \in T_{k-1}$ such that $|\sigma| = r'$ if Case 2 holds, just as we did in Case 1 before.

Let $T = \bigcap_{i \in \omega} T_i$.

Lemma 4 *T is a perfect tree, and each branch of it is Schnorr trivial with respect to each machine M_i .*

Proof For all $k > i$, T_k has the same i th branching point, so T is infinitely branching. Since we have used φ as a condition for determining the branching points, we ensure that for all $\sigma \in T$, there are incomparable τ_1 and τ_2 extending σ . This is sufficient to see that T is a perfect tree.

Let $B \in [T]$, and let $i \in \omega$ and $n \in \omega$. As before, if $K_{M_i}(0^n) = \infty$, we automatically have $K_{M'_i}(B \upharpoonright n) \leq K_{M_i}(0^n) + (i+1)$. Otherwise, there is a least number d such that $\langle d, 0^n \rangle \in M_i$. We will have put $\langle d+i+1, \sigma \rangle$ into M'_i for all σ of length n in T , including $B \upharpoonright n$. Therefore, $K_{M'_i}(B \upharpoonright n) \leq d + (i+1) = K_{M_i}(0^n) + (i+1)$ for each i , and each branch B is Schnorr trivial with respect to each M_i . It follows that each branch B of T is Schnorr trivial.

Corollary 1 *There are 2^{\aleph_0} Schnorr trivials.*

Remark 2 The tree T is recursive in $\mathbf{0}''$, since the only nonrecursive component of the construction is the list of computable machines with domain 1. This list is recursive in $\mathbf{0}''$.

Theorem 11 *Every degree $\mathbf{a} \geq_T \mathbf{0}''$ contains a Schnorr trivial.*

Proof We build a tree T of Schnorr trivials recursively in $\mathbf{0}''$ almost as before. However, now we must build our trees T_i such that every $A \in 2^\omega$ is coded by a branch in $T = \bigcap_{i \in \omega} T_i$.

The first step is to modify the construction to produce a uniform tree. Rather than branch once at each stage $k \geq 1$ so there are k determined branching points at the end of each stage k , we instead extend each string of the appropriate length by both a 1 and a 0, and only at stages of the form $2^k - 1$, giving us up to 2^k branches at the end of these particular stages. This will produce no more branching points at the end of any given stage than permitted in the original argument, so the Kraft-Chaitin Theorem will still apply.

However, we will need more than the simple uniformity of T to code every real $A \in 2^\omega$ within T . We begin by observing that in our previous construction, a 0 would only appear in T at height n if there had been a branching point at height $n-1$ in some T_i . We will adapt this construction so that in our final tree T , the 0s indicate the branching heights in T itself rather than simply in some T_i . This will allow us to code each element of 2^ω into a branch of T via the elements immediately following these 0s.

Now we branch only at stages of the form $3^k - 1$, permitting us to have 3^k branches at the ends of these stages. At these stages, we extend each string of the appropriate length by 11, 01, and 00 instead of simply 1 and 0 as before. Our goal is to produce a final uniform tree $T = \bigcap_{i \in \omega} T_i$ in which 0s appear only at branching points. When we fix the i th branching point in T_k for $k > i$, we prune the 11 branch, leaving only the 01 and 00 branches. Later in the construction, if we must remove a branching point, we prune the 01 and 00 branches to avoid introducing 0s into the final tree at nonbranching points, leaving only the 11 branch. Any branching point generated in this construction will eventually be preserved or removed, so T will be a uniform tree with 2^n branches at the n th branching level.

We can now see that every set in 2^ω is coded by a branch in our tree T . Consider a set A . We code $A(n)$ by the direction we go at the $(n+1)$ st branching point. If $A(n) = 0$, we follow the left, or 00, branch. If $A(n) = 1$, we follow the right, or 01, branch. Let the branch of T obtained in this way from A be called $T(A)$.

Lemma 5 *If $A \geq_T 0''$, $A \equiv_T T(A)$.*

Proof Clearly, $A \geq_T T(A)$: by Remark 2, A computes the tree T since $A \geq_T 0''$, and A can identify the proper path through T as described above.

Similarly, given $T(A)$, we can find $A(n)$ by looking for the $(n+1)$ st disjoint pair beginning with a 0 and looking at the second bit in the pair. If it is a 1, then $A(n) = 1$, and if it is a 0, then $A(n) = 0$. Then $T(A) \geq_T A$.

Therefore, if $\mathbf{a} \geq_T 0''$, there is $B \in [T]$ such that $B \in \mathbf{a}$. We can see that all of the M'_s produced are computable Turing machines and that all $B \in [T]$ are Schnorr trivial as in the original construction, so the theorem is proved.

Now that we have constructed a perfect tree of Schnorr trivial reals in $0''$ coding all $A \in 2^\omega$, we can adapt this construction to be recursive in any degree whose jump is above $0''$. This will give us Theorem 8.

Proof (Proof of Theorem 8)

Let \mathbf{h} be a Turing degree such that $\mathbf{h}' \geq_T 0''$, let $H \in \mathbf{h}$, and let $\langle M_e \rangle_{e \in \omega}$ be a list of all Turing machines. Since $H' \geq_T 0''$ and the statement “the Turing machine M_e is computable, and $\mu(M_e) = 1$ ” is recursive in $0''$, we may define the following functional Ψ .

$$\lim_{t \rightarrow \infty} \Psi(e, t, H) = \begin{cases} 1 & \text{if } M_e \text{ is computable and } \mu(M_e) = 1 \\ 0 & \text{else} \end{cases}$$

We will use this functional to build a perfect tree of Schnorr trivials through a sequence of approximations so that

$$(\forall M_e \text{ comp.})(\exists M'_{e,c} \text{ comp.})(\forall n \in \omega)[K_{M'_{e,c}}(A \upharpoonright n) \leq K_{M_e}(0^n) + (c+1)]$$

and every $B \in 2^\omega$ is coded by a branch of T .

The need for these approximations produces the main differences between this construction and the previous ones. We cannot use a list of the computable Turing machines with domain 1, so we must consider all Turing machines. For each Turing machine with domain < 1 , there will be a c such that the measure of its domain is $< 1 - \frac{1}{2^c}$, so the construction of the corresponding tree as described above would

halt prematurely. Therefore, we will not use M -stages in this construction. Since we cannot identify the computable Turing machines with domain 1 with certainty at any stage of the construction, we will build the trees simultaneously for all M_e rather than building a tree for each M_e within the tree for M_{e-1} after the latter has been entirely determined. This means that we may have parts of the tree that are irrelevant since we will have built them based on a machine that turns out not to be computable with domain 1. Furthermore, we might not begin to construct our tree for M_e before we begin to construct it for M_i for some $i > e$. This may create extra branches in the tree for M_e , but we will not know how many there will be at any given point.

To compensate for this, we will build an $M'_{e,c}$ for each M_e and every $c \geq e$. The constant c will account for c “extra” branches, just as the constant i did for M'_i in the previous construction. Infinitely many such $M'_{e,c}$ will work for each M_e , but we cannot identify them in advance.

We will activate a machine M_e at a stage t when $\Psi(e, t, H) = 1$ and $\mu(M_e) \geq 1 - \frac{1}{2^{3^{k-1}}}$ for a larger k than was possible the last time the machine was activated. The first requirement allows us to only consider machines that appear to be computable with domain 1, and the second allows us to build a uniform tree coding every $B \in 2^\omega$ as before.

$s = 0$: We let $T_0 = 2^{<\omega}$ and $M'_{e,c} = \emptyset$ for all e and c . We then activate all M_e for stage 0.

$s > 0$: We compute $\Psi(e, s, H)$ for all $e \leq s$. If $\Psi(e, s, H) = 0$ for all such e , we let $T_s = T_{s-1}$ and go on to stage $s + 1$. Otherwise, we list all $e \leq s$ such that $\Psi(e, s, H) = 1$ in increasing order: $e_1 < e_2 < \dots < e_n$. For each e_i , let $k_{e_i, s}$ be the largest integer k for which $\mu(M_{e_i, s}) \geq 1 - \frac{1}{2^{3^{k-1}}}$.

First, we consider e_1 . If $k_{e_1, s} = k_{e_1, t}$, where t is the most recent stage at which M_{e_1} was activated, we end stage s . Otherwise, we activate M_{e_1} and list all k such that $3^{k_{e_1, t}} - 1 < k \leq 3^{k_{e_1, s}} - 1$.

For each such k , we will construct our tree very much as before. Since we know such stages will exist now, we can define s_k to be the least stage s such that $\mu(M_s) \geq 1 - \frac{1}{2^k}$ for these k . For each such k , we let $\Sigma = \{\sigma_1, \dots, \sigma_m\}$ be the set of the longest elements of T_{s-1} that have already been determined. We will call their length r .

We now construct the $M'_{e,c}$ s and trees via the following procedure for each such k .

If $\langle d, \tau \rangle$ enters M_{e_1} between states s_{k-1} and s_k , we have the following two cases. Let $n = |\tau|$.

Case 1: $n \leq r$. For all $\sigma \in T_{k-1}$ such that $|\sigma| = n$, we add $\langle d + c + 1, \sigma \rangle$ to $M'_{e_1, c}$ for all $c \geq e_1$. There will be l many such σ for some $l \leq k$, so we add $\langle d + c + 1, \sigma' \rangle$ to $M'_{e_1, c}$ an additional $(k + c) - l$ times, where σ' is the leftmost σ in our tree such that $|\sigma| = n$.

Case 2: $n > r$. For all $\sigma_i \in \Sigma$, we add $\langle d + c + 1, \sigma_i \hat{\ } 1^{n-r} \rangle$ to $M'_{e_1, c}$ for each $c \geq e_1$. This time there will be m such σ_i , so we add $\langle d + c + 1, \sigma_1 \rangle$ to $M'_{e_1, c}$ an additional $(k + c) - m$ times.

Note that in both Case 1 and Case 2, we added $\frac{k+c}{2^{d+c+1}}$ to $\mu(M'_{e_1, c})$ each time we added $\frac{1}{2^d}$ to $\mu(M_{e_1})$.

If k is of the form $3^j - 1$ for some j , we let our tree be

$$T'_{s,1} = \cup_{\sigma \in M'_{e_1, s_k} \text{ and } |\sigma| = r'} ([\sigma \hat{\ } 11] \cup [\sigma \hat{\ } 01] \cup [\sigma \hat{\ } 00])$$

where $r' = \max\{\{r\} \cup \{|\sigma| \mid \sigma \in M'_{e_1, s_k}\}\}$. Otherwise, we do nothing to the tree.

After we have finished the above procedure for all $k \leq 3^{k_{e_1, s}} - 1$, we continue this process for each e_i for $i > 1$ until we reach one for which $k_{e_i, s} = k_{e_i, t}$, where t is the most recent stage at which M_{e_i} was activated, or we reach an e_i whose $k_{e_i, s}$ requires us to branch higher in $T_{s, i-1}$ than the highest previously determined branching point. In either of these cases, we end stage s . If neither case holds, we activate M_{e_i} , augment the $M'_{e_i, c}$ s as before, and build the tree $T_{s, i}$ within $T_{s, i-1}$.

The construction of $T_{s, i}$ must do two things. The first is to preserve certain branching points to ensure that our final tree is perfect. To do this, when we build our tree according to M_e , we will require that the first e branching points appearing in the construction of the $M'_{e, c}$ s will be preserved. Therefore, for these branching points, we will prune the 11 branches of the existing tree. The second is to ensure that 0s only appear in the final tree at its branching points. Therefore, if we must remove a branching point, we will prune the 01 and 00 branches at that height and leave only the 11 branch as before.

Let the last tree produced before the end of the stage be T_s .

Let $T = \cap_{s \in \omega} T_s$.

Lemma 6 *For every $e \in \omega$, there is a $c' \in \omega$ such that $M'_{e, c}$ is a Kraft-Chaitin set for all $c \geq c'$.*

Proof Let $e \in \omega$ be given. By the definition of Ψ , there is an s such that for all $t \geq s$ and $i \leq e$, $\Psi(i, t, H) = 1$ if M_i is computable with domain 1 and $\Psi(i, t, H) = 0$ otherwise. After this stage s , the construction of M'_e will be identical to that in the previous construction. There may be more than $k + e$ branches after $\mu(M_e) \geq 1 - \frac{1}{2^k}$, but this number will now increase at the same rate as it did before. Therefore, some $c' \geq e$ will reflect this number of branches. This is enough to see that $\mu(M'_{e, c}) \leq 1$ for all $c \geq c'$, and since $\mu(M'_{e, c})$ is a recursive real for all c , $M'_{e, c}$ is a Kraft-Chaitin set for all $c \geq c'$.

Lemma 7 *Suppose M_e is computable. Then if $M'_{e, c}$ is a Kraft-Chaitin set, $M'_{e, c}$ is computable as well.*

Proof This proof proceeds in the same manner as the proof of Claim 2.

Lemma 8 *Every branch A of T is Schnorr trivial.*

Proof Let $A \in [T]$, and let M_e be an arbitrary computable machine with domain 1. We will show that $(\exists M'_{e, c} \text{ comp.})(\forall n \in \omega)[K_{M'_{e, c}}(A \upharpoonright n) \leq K_{M_e}(0^n) + (c + 1)]$. Let c' be as in Claim 6, and let $n \in \omega$ be given.

If $K_{M_e}(0^n) = \infty$, it is clear that $K_{M'_{e, c'}}(A \upharpoonright n) \leq K_{M_e}(0^n) + (c' + 1)$. Otherwise, let d be the least integer such that $\langle d, 0^n \rangle$ enters M_e . Then $d = K_{M_e}(0^n)$. When $\langle d, 0^n \rangle$

entered $M_e, \langle d + c' + 1, \sigma \rangle$ entered $M'_{e,c'}$ for all σ of length n in T , including $A \upharpoonright n$. Therefore, $K_{M'_{e,c'}}(A \upharpoonright n) \leq d + (c' + 1) = K_{M_e}(0^n) + (c' + 1)$, and $M'_{e,c'}$ and $c' + 1$ will witness the Schnorr triviality of A with respect to M_e .

Therefore, we will have $K_{M'_{e,c'}}(A \upharpoonright n) \leq K_{M_e}(0^n) + (c' + 1)$ for all n , and $M'_{e,c'}$ and the constant $c' + 1$ will witness the Schnorr triviality of A with respect to M_e .

Now we can code every $B \in 2^\omega$ by a branch in T . We code $B(n)$ by the branch we choose at the $(n + 1)$ st branching point as before. If $B(n) = 0$, we choose the left, or 00, branch. If $B(n) = 1$, we choose the right, or 01, branch. We call the branch chosen in this manner $T(B)$.

Claim If $A \geq_T H$, $A \equiv_T T(A)$.

Proof This proof is identical to that of Claim 5.

Since T is a perfect tree of Schnorr trivial reals, T is recursive in \mathbf{h} , and every $B \in 2^\omega$ is coded by a branch of T , every degree $\geq_T \mathbf{h}$ contains a Schnorr trivial real. Since \mathbf{h} is an arbitrary degree whose jump is $\geq_T \mathbf{0}''$, we have proved Theorem 8.

It should be mentioned that since T is a strongly uniform tree, $T \leq_T B$ for any $B \in [T]$. Therefore, we have actually produced a recursively pointed tree.

As previously noted, the Schnorr trivial Turing degrees are not closed downwards [3]. Theorem 8 provides additional evidence that the Turing degrees are not the appropriate framework in which to consider Schnorr triviality, for it suggests that an arbitrarily computationally strong real may have minimal initial-segment complexity.

The truth table degrees seem to be a much more appropriate framework for Schnorr triviality. Downey, Griffiths, and Laforte have shown that the Schnorr trivial reals are closed downwards in this structure [3]. Furthermore, since the reals that are low for Martin-Löf randomness are precisely those that are K -trivial, one might naturally expect the same relationship to exist for the reals which are low for Schnorr and Schnorr trivial. It can be seen from results of Kjos-Hanssen, Nies, and Stephan and Downey, Griffiths, and Laforte that this is not so [8, 3]. However, I have shown that if we restrict our attention to the hyperimmune-free Turing degrees, the Schnorr trivial reals are precisely those that are low for Schnorr [7]. In the hyperimmune-free degrees, the Turing degrees and truth table degrees coincide. This result, when considered with Downey, Griffiths, and Laforte's, suggests that it is more natural to consider Schnorr triviality in the context of the truth table degrees.

Acknowledgements I would like to thank Theodore A. Slaman, my advisor, and Leo Harrington for many useful conversations and suggestions.

References

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

2. Chaitin, G.J.: Algorithmic information theory. *IBM J. Res. Develop.* **21**(4), 350–359 (1977)
3. Downey, R., Griffiths, E., Laforte, G.: On Schnorr and computable randomness, martin-gales, and machines. *Math. Log. Q.* **50**(6), 613–627 (2004)
4. Downey, R., Hirschfeldt, D.R., Nies, A., Terwijn, S.A.: Calibrating randomness. *Bull. Symbolic Logic* **12**(3), 411–491 (2006)
5. Downey, R.G., Griffiths, E.J.: Schnorr randomness. *J. Symbolic Logic* **69**(2), 533–554 (2004)
6. Downey, R.G., Hirschfeldt, D.R., Nies, A., Stephan, F.: Trivial reals. In: *Proceedings of the 7th and 8th Asian Logic Conferences*, pp. 103–131. Singapore Univ. Press, Singapore (2003)
7. Franklin, J.N.Y.: Hyperimmune-free degrees and Schnorr triviality. In progress
8. Kjos-Hanssen, B., Nies, A., Stephan, F.: Lowness for the class of Schnorr random reals. *SIAM J. Comput.* **35**(3), 647–657 (electronic) (2005)
9. Kučera, A.: Measure, Π_1^0 -classes and complete extensions of PA. In: *Recursion theory week (Oberwolfach, 1984)*, *Lecture Notes in Math.*, vol. 1141, pp. 245–259. Springer, Berlin (1985)
10. Martin-Löf, P.: The definition of random sequences. *Information and Control* **9**, 602–619 (1966)
11. Miller, W., Martin, D.A.: The degrees of hyperimmune sets. *Z. Math. Logik Grundlagen Math.* **14**, 159–166 (1968)
12. Nies, A.: Lowness properties and randomness. *Adv. Math.* **197**(1), 274–305 (2005)
13. Nies, A., Stephan, F., Terwijn, S.A.: Randomness, relativization and Turing degrees. *J. Symbolic Logic* **70**(2), 515–535 (2005)
14. Schnorr, C.P.: A unified approach to the definition of random sequences. *Math. Systems Theory* **5**, 246–258 (1971)
15. Terwijn, S.A., Zambella, D.: Computational randomness and lowness. *J. Symbolic Logic* **66**(3), 1199–1205 (2001)