MARTIN-LÖF REDUCIBILITY AND COST FUNCTIONS

NOAM GREENBERG, JOSEPH S. MILLER, ANDRÉ NIES, AND DAN TURETSKY

Abstract. Martin-Löf (ML)-reducibility compares K-trivial sets by examining the Martin-Löf random sequences that compute them. We show that every K-trivial set is computable from a c.e. set of the same ML-degree. We investigate the interplay between ML-reducibility and cost functions, which are used to both measure the number of changes in a computable approximation, and the type of null sets used to capture ML-random sequences. We show that for every cost function there is a c.e. set ML-above the sets obeying it (called a "smart" set for the cost function). We characterise the K-trivial sets computable from a fragment of the left-c.e. random real Ω. This leads to a new characterisation of strong jump-traceability.

Contents

1. Introduction

K-trivial sets sets are very close to being computable. Far from being an obstacle, this can be advantageous for the study of their relative computational complexity: tools can be applied that would not work for more complex Δ^0_2 sets. We use this idea to study computational complexity inside the class of K-trivials, which for a long time had looked quite amorphous.

The approach we take is to study the complexity of K-trivial sets via their interaction with the Martin-Löf (ML-)random sequences. K-triviality can be characterised by such an interaction: a set A is K-trivial if and only if it is computable from an A-random sequence [15]; a c.e. set is K-trivial if and only if it is computable from an incomplete random sequence [6, 2]; a set is K-trivial if and only if it is computable from every ML-random sequence that is not Oberwolfach random [3, 6, 2].

Date: June 24, 2017.

2010 *Mathematics Subject Classification.* Primary 03D32; Secondary 03D30, 68Q30.

Greenberg and Nies were supported by the Marsden Fund of New Zealand. Greenberg was also supported by a Rutherford Discovery Fellowship from the Royal Society of New Zealand. This research was started during a retreat at the Research Centre Coromandel, February 2015.

We start differentiating between K-trivials when we ask what kind of incomplete randoms compute them. For example, some but not all K-trivials are computable from both halves of a random sequence [3]. In [11], three of the authors of the present paper characterised this subclass of K-trivial sets, and placed it within a dense linear hierarchy of sub-ideals of the K-trivials. These ideals have a dual characterisation, using either ML-reducibility or cost functions. In this paper, we give a systematic study of the interaction of these concepts in the K-trivials.

Definition 1.1 ([3])**.** For sets A and B, we write $B \leqslant_{\mathrm{ML}} A$ if $A \leqslant_{\mathrm{T}} Y$ implies $B \leqslant_{\mathrm{T}} Y$ for every ML-random sequence Y.

ML-reducibility is a weakening of Turing reducibility. The least degree consists of the computable sets. The usual join operation induces a least upper bound in the ML-degrees.

Our first result, Theorem 2.1, strengthens the intuition that K-trivial sets are inherently enumerable. The ideal of Turing degrees of K-trivial sets is c.e. generated: every K-trivial is computable from a c.e. K-trivial. Our result says that every K-trivial is, in fact, computable from a c.e. K-trivial of the same ML-degree. This is a powerful tool. It is usually easier to prove results for the c.e. K-trivials; extra work is needed to lift results to the general case. Theorem 2.1 simplifies this process in many cases. Indeed, we use it in both Section 4 and Section 6 for this purpose.

The non-K-trivial c.e. sets are computable only from complete randoms [15] and so consist of a single ML-degree. We are mainly interested in the ML-degrees of K-trivial sets, a setting in which an interesting degree structure emerges. In [3], it is shown that there is a greatest K-trivial ML-degree; on the other hand, it is well-known that there is no greatest Turing degree of K-trivial sets [18, 5.3.22]; also see the end of our Section 3. We show in Theorem 5.14 that every countable partial ordering is embeddable into the ML-degrees of K-trivial sets. Also, there is a pair of incomparable degrees below each non-zero K-trivial ML-degree.

Cost functions were introduced in [18, Section 5.3] and developed further in [12, 20].

Definition 1.2. A *cost function* is a computable function

$$\mathbf{c}\colon \mathbb{N} \times \mathbb{N} \to \{r \in \mathbb{R} : r \geqslant 0\}.$$

We only consider *monotonic* cost functions (satisfying $\mathbf{c}(x,s) \leqslant \mathbf{c}(x,s+1)$ and $\mathbf{c}(x,s) \geqslant \mathbf{c}(x+1,s)$) that have the *limit condition*: for all x, $\underline{\mathbf{c}}(x) = \lim_s \mathbf{c}(x,s)$ exists, and $\lim_x \underline{\mathbf{c}}(x) = 0$. Further, we assume that $\mathbf{c}(x,s) = 0$ when $x \geqslant s$.

The original purpose of cost functions was to quantify the number of changes required in a computable approximation of a Δ^0_2 set A: $\mathbf{c}(x,s)$ is the cost of changing at stage s our guess about the value of $A(x)$. Monotonicity means that the cost of a change increases with time, and that changing the value at a smaller number is more costly. Formally:

Definition 1.3 ([18])**.** Let $\langle A_s \rangle$ be a computable approximation of a Δ^0_2 set A, and let $\mathbf{c}$ be a cost function. The *total* $\mathbf{c}$*-cost* of the approximation is

$$\mathbf{c}\langle A_s \rangle = \sum_{s \in \omega} \{\mathbf{c}(x,s) : x \text{ is least such that } A_{s-1}(x) \neq A_s(x)\}.$$

We say that a Δ^0_2 set A *obeys* $\mathbf{c}$ if the total $\mathbf{c}$-cost of *some* computable approximation of A is finite. We write $A \models \mathbf{c}$. For cost functions $\mathbf{c}$ and $\mathbf{d}$, we write $\mathbf{c} \to \mathbf{d}$ if $A \models \mathbf{c}$ implies $A \models \mathbf{d}$ for each Δ^0_2 set A. By [20, Thm. 3.4], this is equivalent to $\underline{\mathbf{d}} \leqslant^\times \underline{\mathbf{c}}$.

The basic existence theorem for cost functions, e.g., described in [20, Thm. 2.7(i)], says that if a cost function $\mathbf{c}$ has the limit condition, then some non-computable c.e. set obeys A. An important example of a cost function is $\mathbf{c}_\Omega(x, s) = \Omega_s - \Omega_x$, where $\langle \Omega_s \rangle$ is an increasing sequence of rational numbers converging to a left-c.e., ML-random real Ω. A set obeys this cost function if and only if it is K-trivial ([20, Thm. 4.3], which modified a result for a related cost function in [17]).

Cost functions can also be used to introduce randomness notions between weak 2-randomness and ML-randomness.

Definition 1.4 ([3])**.** Let $\mathbf{c}$ be a cost function. A nested sequence $\langle V_n \rangle$ of uniformly c.e. open sets is a $\mathbf{c}$-*bounded test* (or $\mathbf{c}$-test for short) if $\mu(V_n) \leqslant^\times \underline{\mathbf{c}}(n)$ for all n.

The test *captures* the elements of $\bigcap_n V_n$. We also say that these elements *fail* the test. A sequence is $\mathbf{c}$-random if it fails no $\mathbf{c}$-test. The motivating result connecting the two uses of cost functions is the following:

Proposition 1.5. *If $A \models \mathbf{c}$ and Y is a ML-random sequence captured by a $\mathbf{c}$-bounded test, then $A \leqslant_{\mathrm{T}} Y$.*

Kučera showed that every Δ^0_2 ML-random sequence is Turing above a non-computable c.e. set. Hirschfeldt and Miller in unpublished work dating from 2006 strengthened this: below any Σ^0_3 null class of randoms there is a non-computable c.e. set. Relying on Proposition 1.5, these proofs can be framed in the language of cost functions; see [12] and [18, 5.3.15], respectively.

A main goal of the present paper is to understand the ML-degrees of K-trivial sets by examining the cost functions that they obey. A special case was discovered in [3], where it is shown that (a) a ML-random is captured by a $\mathbf{c}_\Omega$-test if and only if it computes all K-trivial sets; and (b) there is a K-trivial set A computable only from randoms that are captured by a $\mathbf{c}_\Omega$-test. Thus, this set has greatest ML-degree among the K-trivials. A set of this type was called "smart". We extend this notion to designate a converse to Proposition 1.5.

Definition 1.6. Let $\mathbf{c}$ be a cost function and A be a K-trivial set. We say that A is *smart for* $\mathbf{c}$ if A obeys $\mathbf{c}$ and for each ML-random set Y,

$$Y \text{ is captured by a } \mathbf{c}\text{-bounded test} \Leftrightarrow A \leqslant_{\mathrm{T}} Y.$$

Informally, A is as complex as possible for obeying $\mathbf{c}$, in the sense that the only random sets Y above A are the ones that have to be there because of Proposition 1.5. In Theorem 3.3, we show that there is a smart set for any cost function $\mathbf{c}$ such that obedience to $\mathbf{c}$ implies K-triviality. Dually, in Proposition 4.1, we prove that any K-trivial set is smart for some cost function $\mathbf{c}_A$; when A is c.e., this will be the strongest cost function obeyed by A.

By an n-column of a set Z we mean the bits of Z in a location of the form $kn + r$, for a fixed $r < n$ and all $k \in \mathbb{N}$. Our results, together with the work in [11], show that there is a natural dense chain of ML-degrees: the degree corresponding to the rational number k/n is the greatest ML-degree possible for a K-trivial set that is computable from every join of k of the n-columns of a left-c.e. random sequence Ω (equivalently, any random sequence). Such K-trivials are characterised as those obeying the cost function $\mathbf{c}_{\Omega,k/n}(x, s) = (\Omega_s - \Omega_x)^{k/n}$. We show in Section 5 that, in fact, for c.e. sets and K-trivial sets, being computable from *one* of the joins of k of the n-columns of Ω is sufficient. Thus, for example, a c.e. set is computable from one 2-column of Ω if and only if it is computable from the other 2-column.

The work in Section 5 is very general; it applies to any fragment of Ω given by the bits with location in an infinite computable set. We are able to classify, in terms of an appropriate cost function, which K-trivials are computable from a given fragment, and thus compare the computational power of different fragments with respect to the dual of ML-reducibility, which is coarser than Turing reducibility. We use this analysis in Section 7 to show that the strongly jump-traceable sets are exactly the sets computable from every such fragment of Ω.

We remark that the correspondence between ML-degrees and cost functions is incomplete. In one direction it is well-behaved: a cost function $\mathbf{c}$ determines a ML-degree, that of the sets which are smart for $\mathbf{c}$. But not every set in that degree obeys $\mathbf{c}$. In the other direction, every K-trivial set A is smart for some cost function $\mathbf{c}_A$, but this cost function is not determined by the ML-degree of A; in fact, in Theorem 4.3 we construct an example of a set A such that even the *shift* of A (i.e., the set that results from removing the first bit of A) does not obey $\mathbf{c}_A$. Ideally, we could characterise ML-reducibility on K-trivials in terms of which cost functions they obey. This would give a satisfying positive answer to the following question, which remains open:

Question 1.7. Is the relation $\leqslant_{\mathrm{ML}}$ on the K-trivial sets arithmetical?

In fact, the weaker question remains open: whether ML-completeness among the K-trivals is arithmetical.

2. Inherent enumerability of K-trivials

In this section, we prove a powerful generalisation of the fact that every K-trivial is computable from a c.e. K-trivial: we show that the c.e. K-trivial can be taken to have the same ML-degree.

Theorem 2.1. *For every K-trivial set A, there is a (K-trivial) c.e. set $D \geqslant_{\mathrm{T}} A$ such that $D \leqslant_{\mathrm{ML}} A$.*

This goes a long way to formalising the intuition that K-triviality is essentially a c.e. notion. In particular, every K-trivial ML-degree contains a c.e. set.

Note that the K-triviality of D is free: every K-trivial is below an incomplete ML-random sequence [6, 2], and any c.e. set below an incomplete random is K-trivial. So by virtue of being ML-equivalent to A, the c.e. set D must be K-trivial.

Theorem 2.1 follows from a fact of independent interest:

Proposition 2.2. *For every K-trivial set A, there is a computable approximation $\langle A_s \rangle$ of A such that for every random X and Turing functional Φ with $A = \Phi^X$, for all but finitely many n, if $A \upharpoonright n \preccurlyeq \Phi_s^X$, then for every $t \geqslant s$, $A_t \upharpoonright n = A \upharpoonright n$.*

Intuitively, the approximation to A converges faster than any computation of A from a random, and so a random computing A must also compute a modulus for A; this modulus will have a c.e. degree.

For the rest of the paper, we fix a Turing functional Υ that is universal in the sense that $\Upsilon^{0^e1\hat{\ }X} = \Phi_e^X$ for each X and e. We assume that for every e, for all but finitely many n, for all s and X, $\Phi_{e,s}(X;n)\downarrow \Rightarrow \Upsilon_s(0^e1\hat{\ }X;n)\downarrow$. Thus it will suffice to prove Proposition 2.2 for the functional Υ.

Proof of Proposition 2.2. Let us first give a brief explanation of the proof. We will use the "main lemma" derived from the golden run construction [18, 5.5.1]. This lemma says that if we design a left-c.e. oracle discrete measure on ω, (equivalently, an adaptive additive cost function, or a prefix-free oracle machine), then there is a sufficiently speedy computable approximation $\langle A_s\rangle$ of A such that the total of all weights that are believed at some stage of the construction and later are shown to be false is finite. Roughly, we would like, at stage s, to put the weight $\mu(\Upsilon_s^{-1}[A_s \upharpoonright n])$ on the string $A_s \upharpoonright n$, where μ is Lebesgue measure on Cantor space, and

$$\Upsilon_s^{-1}[\sigma] = \left\{X \in 2^\omega \;:\; \Upsilon_s^X \succcurlyeq \sigma\right\}.$$

A speedy approximation for A will be as required: we can put a Solovay test on the reals that compute A too early, and thus random oracles will only converge and agree with $A_s \upharpoonright n$ after it has settled.

The problem is that this definition does not give a discrete measure: there is no reason to believe that $\sum_n \mu(\Upsilon^{-1}[A \upharpoonright n])$ is finite. What we notice, though, is that if an oracle X gives us a correct version of A too early, then this version $A_s \upharpoonright n = A \upharpoonright n$ will later change to $A_t \upharpoonright n \neq A \upharpoonright n$, but after that will need to change back. We can thus put the weight not on $A \upharpoonright n$ but on an incorrect version $A_t \upharpoonright n$. And this is guaranteed to give a measure: the collection of strings of the form $(A \upharpoonright n)\hat{\ }(1-A(n))$ that disagree with A only on the last bit is pairwise incomparable, and so the pull-backs by Υ of these strings are pairwise disjoint.

Now to the formal details. For $\sigma \in 2^{<\omega}$ with $\sigma \neq \langle\rangle$, define $\hat{\sigma}$ to be the binary string of the same length which disagrees with σ on the final bit, but agrees on all other bits. For example, if $\sigma = 001011$, then $\hat{\sigma} = 001010$. For $\sigma \in 2^{<\omega}$ with $\sigma \neq \langle\rangle$, for brevity, define

$$\mathcal{U}_\sigma = \left\{X \;:\; \Upsilon^X \succcurlyeq \hat{\sigma}\right\}.$$

To avoid needing to repeatedly deal with $\langle\rangle$ separately, define $\mathcal{U}_{\langle\rangle} = \varnothing$. Note that $(\mathcal{U}_\sigma)_{\sigma\in 2^{<\omega}}$ are uniformly Σ^0_1-classes. Also, for $\sigma \precneq \rho$, $\mathcal{U}_\sigma$ and $\mathcal{U}_\rho$ are disjoint.

For our argument, we will require a computable approximation to A that obeys an adaptive cost function—a cost function where the cost at a given stage depends on the approximation up to that stage. For $\langle A_t\rangle$, a computable approximation of A, define

$$\mathbf{c}^{\langle A_t\rangle}(n,s) = \mu(\mathcal{U}_{A\upharpoonright n+1}[s]).$$

Claim 2.2.1. There is a computable approximation $\langle A_t\rangle$ to A that obeys $\mathbf{c}$. That is, if n_s is least with $A_s(n_s) \neq A_{s+1}(n_s)$, then $\sum_s \mathbf{c}^{\langle A_t\rangle}(n_s,s) < \infty$.

Proof. Uniformly in σ and s, let $C_{\sigma,s} \subset 2^{<\omega}$ be a finite anti-chain that generates $\mathcal{U}_{\sigma,s}$, with $C_{\sigma,s} \subseteq C_{\sigma,s+1}$. Define an oracle machine M with $M_s^\sigma(\pi)\downarrow$ for $\pi \in C_{\sigma,s}$. Since $\mathcal{U}_\sigma$ and $\mathcal{U}_\rho$ are disjoint for $\sigma \precneq \rho$, M is prefix-free.

Note that for $s > n$ and any computable approximation $\langle A_t\rangle$,

$$\sum_\pi 2^{-|\pi|}\left[\!\left[M^A(\pi)[s]\downarrow \;\&\; \text{use}\, M^A(\pi)[s] = n+1\right]\!\right] \geqslant \mathbf{c}^{\langle A_t\rangle}(n,s).$$

Fix a computable approximation $\langle \dot{A}_q\rangle$ to A. By the Main Lemma of the golden run, there is a computable sequence $q(0) < q(1) < \cdots$ with $q(0) \geqslant 1$, such that if we define m_s to be least with $\dot{A}_{q(s)} \neq \dot{A}_{q(s+1)}$, then

$$\sum_s\sum_\pi 2^{-|\pi|}\left[\!\!\left[M^{\dot{A}}(\pi)[q(s)]\downarrow \;\&\; m_s < \text{use}\, M^{\dot{A}}(\pi)[q(s)] \leqslant q(s-1)\right]\!\!\right] < \infty.$$

Now let $A_t = \dot{A}_{q(t)}$, so $n_s = m_s$. Since $s \leqslant q(s-1)$, if $n_s < s$ then the inner summation above is at least $\mathbf{c}^{\langle A_t \rangle}(n_s, s)$. So $\sum_s \mathbf{c}^{\langle A_t \rangle}(n_s, s) < \infty$, as desired. $\square_{2.2.1}$

Now, let $\langle B_s \rangle$ be a uniformly computable sequence of finite anti-chains with

$$\bigcup_{\tau \in B_s} [\tau] = \mathcal{U}_{A \upharpoonright n_s + 1}[s],$$

and define $S = \bigcup_s B_s$. Note that

$$\begin{aligned} \sum_{\tau \in S} 2^{-|\tau|} &\leqslant \sum_s \sum_{\tau \in B_s} 2^{-|\tau|} \\ &= \sum_s \mu(\mathcal{U}_{A \upharpoonright n_s + 1}[s]) \\ &= \sum_s \mathbf{c}^{\langle A_t \rangle}(n_s, s) < \infty. \end{aligned}$$

Thus S is a Solovay test.

Let X be random and suppose that $A = \Phi_e^X$. Then $Y = 0^e 1 \hat{} X$ is random and $A = \Upsilon^Y$. Since Y is not captured by S, we can fix an s_0 such that no B_s with $s \geqslant s_0$ contains an initial segment of Y. Fix N such that for all $n \geqslant N$, if $A \upharpoonright n \preccurlyeq \Upsilon_s^Y$, then $s \geqslant s_0$.

Claim 2.2.2. For all $n \geqslant N$, if $A \upharpoonright n \preccurlyeq \Upsilon_s^Y$, then for every $t \geqslant s$, $A_t \upharpoonright n = A \upharpoonright n$.

Proof. Suppose $n \geqslant N$ were a counterexample. Let s be such that $A \upharpoonright n \preccurlyeq \Upsilon_s^Y$ and $t \geqslant s$ be such that $A_t \upharpoonright n \neq A \upharpoonright n$ and $A_{t+1} \upharpoonright n = A \upharpoonright n$. Note that definitionally, $n_t < n$. Since $A_t \upharpoonright n_t = A_{t+1} \upharpoonright n_t$, we know that $A_t \upharpoonright n_t = A \upharpoonright n_t \prec \Upsilon_s^Y$ and $A_t \upharpoonright n_t + 1 \neq A \upharpoonright n_t + 1$. So $(\widehat{A_t \upharpoonright n_t + 1}) = A_{t+1} \upharpoonright n_t + 1$, and $Y \in \mathcal{U}_{A_t \upharpoonright n_t + 1}$. By assumption, Y has already entered this Σ_1^0-class by stage s. Since $t \geqslant s$, B_t contains an initial segment of Y, contrary to our choice of N and s_0. $\square_{2.2.2}$

Since convergence of $\Phi_{e,s}^X$ up to n implies convergence of Υ_s^Y up to n for all but finitely many n, the proposition follows. $\square_{2.2}$

Proof of Theorem 2.1. Let $\langle A_s \rangle$ be the approximation from the previous lemma. Let D be the change-set for this approximation: $(n, k) \in D$ if and only if there is a sequence of stages $s_0 < \cdots < s_k$ with $A_{s_i}(n) \neq A_{s_{i+1}}(n)$ for all $i < k$. D is clearly c.e., and it can compute $A(n)$ by searching for the least k with $(n, k) \notin D$ and considering the parity of k and the value of $A_0(n)$.

Suppose that $A = \Phi^X$ for some random X. By the lemma, for all but finitely many n, the approximation converges to $A \upharpoonright n$ faster than Φ^X does. Thus X can compute $D(n, k)$ by waiting until a stage t with $A \upharpoonright n \preccurlyeq \Phi_t^X$ and then only searching for sequences of stages below t. For the finitely many n on which Φ^X converges faster than the approximation, we can arrange that our computation knows $D(n, k)$ non-uniformly. $\square$

3. The most powerful set obeying a given cost function

In this section, we prove the existence of smart sets for cost functions that imply K-triviality. Recall that for cost functions $\mathbf{c}$ and $\mathbf{d}$, we write $\mathbf{c} \to \mathbf{d}$ if $A \models \mathbf{c}$ implies $A \models \mathbf{d}$ for every Δ_2^0 set A. By [20, Thm. 3.4], this is equivalent to $\underline{\mathbf{c}} \geqslant^{\times} \underline{\mathbf{d}}$, that is, $\underline{\mathbf{c}}$ multiplicatively dominates $\underline{\mathbf{d}}$ (we may assume $\underline{\mathbf{c}}(x) > 0$ for every x). Since obedience to $\mathbf{c}_\Omega$ characterises K-triviality, obedience to a cost function $\mathbf{c}$ implies K-triviality if and only if $\mathbf{c} \to \mathbf{c}_\Omega$.

We start with a couple of very simple lemmas.

Lemma 3.1. *Suppose that aY fails a $\mathbf{c}$-bounded test $\bigcap_n V_n$, where $a \in \{0,1\}$. Then Y fails a $\mathbf{c}$-bounded test.*

Proof. We may suppose $a = 0$ and $X \in V_n$ implies $X(0) = 0$. Then $\mu(T[V_n]) = 2\mu(V_n)$, where T is the usual shift operator on Cantor space, and so $\langle T[V_n]\rangle$ is also a $\mathbf{c}$-bounded test. Clearly Y fails it. □

We recall that an *additive* cost function is a cost function of the form $\mathbf{c}_\alpha(n,s) = \alpha_s - \alpha_n$, where $\langle \alpha_s \rangle$ is an increasing approximation of a left-c.e. real α. So $\underline{\mathbf{c}}_\alpha(n) = \alpha - \alpha_n$. By the universality of Ω, every time we see an increase in α, we can cause a proportional and later increase in Ω. Thus:

Lemma 3.2. *If $\mathbf{c}_\alpha$ is an additive cost function, then $\mathbf{c}_\Omega \to \mathbf{c}_\alpha$.*

In particular, $2^{-n} \leqslant^\times \underline{\mathbf{c}}_\Omega(n)$.

Theorem 3.3. *If $\mathbf{c}$ is a cost function such that $\mathbf{c} \to \mathbf{c}_\Omega$, then some c.e. set A is smart for $\mathbf{c}$.*

Proof. Recall that Υ is a "universal" Turing functional in the sense that $\Upsilon^{0^e 1\hat{\ } X} = \Phi_e^X$ for all X and e. We build A and a $\mathbf{c}$-test $\langle \mathcal{U}_k \rangle$ capturing any ML-random Y such that $A = \Upsilon^Y$. This suffices for the theorem by Lemma 3.1. The tension in this construction is between trying to capture all reals computing A, and keeping the measure of $\mathcal{U}_n$ bounded by (a multiple of) $\underline{\mathbf{c}}(n)$. The idea is for us to move A in case we see that too many oracles compute it. This needs to be done judiciously; we must ensure that A obeys $\mathbf{c}$. The basic idea, as in [3], is to charge the cost of changing A to the increase in the measure of the *error set*, the set of oracles that have already been proven to be incorrect about A. Since $\mathbf{c} \to \mathbf{c}_\Omega$, the increase in the error set is bounded by $\mathbf{c}$, and so we can catch our tail.

To the details. During the construction of A we build a global "error set":

$$\mathcal{E}_s = \left\{ Y \,:\, \Upsilon_s^Y \text{ lies to the left of } A_s \right\}.$$

An enumeration of a number into A causes A to move to the right, and so potentially increases $\mathcal{E}$. The basic idea, again, is that we enumerate a number x into A only when the cost $\mathbf{c}(x,s)$ is smaller than the amount by which $\mathcal{E}$ will be increased.

We will ensure that at every stage s,

$$(\diamond) \qquad \mu(\mathcal{U}_{k,s}) \leqslant \mathbf{c}(k,s) + \mu(\mathcal{E}_{s+1} - \mathcal{E}_k).$$

By Lemma 3.2, $\mu(\mathcal{E} - \mathcal{E}_k) = \underline{\mathbf{c}}_{\mu(\mathcal{E})}(k) \leqslant^\times \underline{\mathbf{c}}_\Omega(k)$, so as $\underline{\mathbf{c}}_\Omega(k) \leqslant^\times \underline{\mathbf{c}}(k)$, the test $\langle \mathcal{U}_k \rangle$ is indeed a $\mathbf{c}$-test.

We reserve the interval $I_k = [2^k, 2^{k+1})$ for ensuring $(\diamond)$. The construction of $\mathcal{U}_k$ (and A) is as follows. At stage $s > k$ we let

$$\mathcal{V}_{k,s} = \left\{ Y \,:\, \Upsilon_s^Y \prec A_s \;\&\; |\Upsilon_s^Y| \geqslant 2^{k+1} \right\};$$

and

$$\mathcal{U}_{k,s} = \bigcup_{t \in [k,s]} \mathcal{V}_{k,t}.$$

As $\mathcal{V}_{k,s} \supseteq \mathcal{V}_{k+1,s}$, the test $\langle \mathcal{U}_k \rangle$ is nested. Note that $\mathcal{V}_{k,s}$ is disjoint from $\mathcal{E}_k$, for every s, hence $\mathcal{U}_{k,s}$ is disjoint from $\mathcal{E}_k$.

Let $s > k$. We let $x_s = x_s(k) = \min(I_k - A_s)$. If $(\diamond)$ threatens to fail at s, namely $\mu(\mathcal{U}_{k,s}) > \mathbf{c}(k,s) + \mu(\mathcal{E}_s - \mathcal{E}_k)$, we enumerate $x_s(k)$ into A_{s+1}. This causes $\mathcal{U}_{k,s}$ to go into $\mathcal{E}_{s+1}$. Since $\mathcal{U}_{k,s}$ is disjoint from $\mathcal{E}_k$, it follows in this case that $\mu(\mathcal{U}_{k,s}) \leqslant \mu(\mathcal{E}_{s+1} - \mathcal{E}_k)$, and so $(\diamond)$ holds at stage s.

First we verify that x_s always exists, that is, we enumerate at most 2^k times for $\mathcal{U}_k$. By Lemma 3.2, we may assume that $\mathbf{c}(x,s) \geqslant 2^{-x}$ for $x < s$. (To be clear, here we are using the fact that if $\underline{\mathbf{c}} =^\times \underline{\mathbf{d}}$, then the same sets obey $\mathbf{c}$ and $\mathbf{d}$.) If we enumerate $x_s(k)$ into A_{s+1}, then $\mu(\mathcal{U}_{k,s}) > 2^{-k} + \mu(\mathcal{E}_s - \mathcal{E}_k)$. Since $\mathcal{U}_{k,s} \cap \mathcal{E}_k = \varnothing$, and $\mathcal{U}_{k,s} \subseteq \mathcal{E}_{s+1}$, it follows that $\mu(\mathcal{E}_{s+1} - \mathcal{E}_s) > 2^{-k}$. Since $\mu(\mathcal{E}) \leqslant 1$, this can happen at most 2^k times.

By delaying computations from appearing in Υ, we may assume that for all Y and s, Υ^Y_s does not lie to the right of A_s. Hence, if $A = \Upsilon^Z$ then $Z \in \bigcap_k \mathcal{U}_k$. It remains to verify that $A \models \mathbf{c}$. If we enumerate $x_s(k)$ into A_{s+1}, then

$$\mu(\mathcal{U}_{k,s} - \mathcal{E}_s) = \mu(\mathcal{U}_{k,s} - (\mathcal{E}_s - \mathcal{E}_k)) \geqslant \mu(\mathcal{U}_{k,s}) - \mu(\mathcal{E}_s - \mathcal{E}_k) > \mathbf{c}(k,s) \geqslant \mathbf{c}(x,s).$$

Since $\mathcal{U}_{k,s} - \mathcal{E}_s \subseteq \mathcal{E}_{s+1} - \mathcal{E}_s$, we see that $\mathbf{c}(x,s) < \mu(\mathcal{E}_{s+1} - \mathcal{E}_s)$. This implies that the total cost of the enumeration of A is at most $\mu(\mathcal{E}) \leqslant 1$. □

Definition 3.4. Let $\mathbf{c}$ be a cost function and let A be a Δ^0_2 set. We say that A is *ML-complete for* $\mathbf{c}$ if $A \models \mathbf{c}$, and $\forall B\,[B \models \mathbf{c} \Rightarrow B \leqslant_{\mathrm{ML}} A]$.

Corollary 3.5. *A is smart for $\mathbf{c}$ $\Leftrightarrow$ A is ML-complete for $\mathbf{c}$.*

Proof. ($\Rightarrow$) Suppose that $A \leqslant_{\mathrm{T}} Y$ for ML-random Y. Then some $\mathbf{c}$-bounded test captures Y. If $B \models \mathbf{c}$, then $B \leqslant_{\mathrm{T}} Y$ by the basic fact, Proposition 1.5.
($\Leftarrow$) Let $\widetilde{A}$ be smart for $\mathbf{c}$. If $A \leqslant_{\mathrm{T}} Y$ for ML-random Y, then $\widetilde{A} \leqslant_{\mathrm{T}} Y$, so Y is captured by a $\mathbf{c}$-bounded test. □

In particular, the ML-degree of a smart set A for $\mathbf{c}$ is uniquely determined by $\mathbf{c}$. On the other hand, for each low c.e. set A, there is a c.e. set $B \not\leqslant_T A$ such that $B \models \mathbf{c}$ [18, 5.3.22]. If A is smart for $\mathbf{c}$, then $A \oplus B$ is also smart for $\mathbf{c}$. As every K-trivial is low, the Turing degree of a set A that is smart for $\mathbf{c}$ is not uniquely determined by $\mathbf{c}$.

4. The strongest cost function obeyed by a c.e. K-trivial set

Given a K-trivial set A, we will define a cost function $\mathbf{c}_A$ with $A \models \mathbf{c}_A$ such that every random computing A is captured by a $\mathbf{c}_A$ test. In other words, we build $\mathbf{c}_A$ in such a way that A is smart for $\mathbf{c}_A$. Furthermore, in case that A is c.e., $\mathbf{c}_A$ is the strongest cost function that A obeys, in the sense that if $A \models \mathbf{c}$, then $\mathbf{c}_A \to \mathbf{c}$.

We note that $\mathbf{c}_A$ may not behave in an overly nice way. We build a c.e. K-trivial A such that the class of sets obeying $\mathbf{c}_A$ is not closed downward under $\leqslant_{\mathrm{T}}$. In fact, in our example $T(A) \not\models \mathbf{c}_A$, where $T(A)$ is the shift of A, obtained by deleting the first bit.

As before, Υ denotes a universal Turing functional. Let A be K-trivial. The idea for defining $\mathbf{c}_A$ is as follows. Suppose first that A is c.e., and let $\langle A_s \rangle$ be an effective enumeration of A that obeys $\mathbf{c}_\Omega$ [20]. We want to define $\mathbf{c}_A$ so that we can capture by a $\mathbf{c}_A$-bounded test all the reals Z such that $\Upsilon^Z = A$. The natural test is $\mathcal{U}_k = \bigcup_{s \geqslant k} \Upsilon^{-1}_s[A_s \upharpoonright k]$. So we define $\underline{\mathbf{c}}_A(k) = \mu(\mathcal{U}_k)$. Why does A obey this cost function? Since the approximation is left-c.e., A does not have to pay for the measure of those oracles that compute $A \upharpoonright k$ correctly: these only appear after $A \upharpoonright k$

has settled, and after that, all changes to A are beyond k. So we only need to pay for the measure of those reals Z that compute an incorrect version $A_s \upharpoonright k$. This price is bounded by the increase of the measure of the error set: those oracles that compute some string to the left of A. Thus the total A-cost is the same as the total A-cost of an additive cost function, and hence bounded by the total $\mathbf{c}_\Omega$-cost of this enumeration; but this was chosen to be finite.

When A is not c.e., we use a c.e. intermediary. Let us give the details of the definition. By Theorem 2.1, fix a c.e. set $C \equiv_{\mathrm{ML}} A$ that computes A; let Ψ be a Turing functional such that $A = \Psi^C$. Fix an enumeration $\langle C_s \rangle$ of C and an approximation $\langle A_s \rangle$ of A that witnesses $A \models \mathbf{c}_\Omega$. By speeding up both Ψ and our approximations, we may assume that $A_s \upharpoonright s \preccurlyeq \Psi_s^{C_s}$ for every s. To unify our construction with the earlier discussion, we assume that if A is c.e., then $C = A$ and Ψ is the natural reduction with identity bounded use.

Similarly to what we did above, we let

$$\mathcal{E}_s = \left\{ Y \,:\, \Upsilon_s^Y \text{ lies to the left of } C_s \right\}$$

and

$$\mathcal{V}_{x,s} = \left\{ Y \,:\, \Upsilon_s^Y < C_s \;\&\; A_s \upharpoonright x+1 \preccurlyeq \Psi_s^{\Upsilon_s^Y} \right\};$$

we then let

$$\mathbf{c}_A(x,s) = \mu \left(\bigcup_{x<t\leqslant s} \mathcal{V}_{x,t} \right).$$

Note that $\mathbf{c}_A$ is monotonic, as $\mathcal{V}_{x,t} \supseteq \mathcal{V}_{x+1,t}$. It satisfies the limit condition if A is non-computable: certainly for all x, $\underline{\mathbf{c}}_A(x) \leqslant 1$. If $A \upharpoonright k$ has stabilised by stage s, then $\underline{\mathbf{c}}_A(s) \leqslant \mu \left\{ Y \,:\, A \upharpoonright k \preccurlyeq \Psi^{\Upsilon^Y} \right\}$. Hence $\lim_x \underline{\mathbf{c}}_A(x) \leqslant \mu \left\{ Y \,:\, A = \Psi^{\Upsilon^Y} \right\}$; if A is non-computable, this is 0.

Proposition 4.1. *A is smart for $\mathbf{c}_A$.*

Proof. First we show that A obeys $\mathbf{c}_A$. In fact, the fixed approximation $\langle A_s \rangle$ witnesses this. Define an increasing approximation of the left-c.e. "error real" by

$$\varepsilon_s = \mu \left(\mathcal{E}_{s+1} \right).$$

Suppose that $A_s(x) \neq A_{s+1}(x)$. For each $t \in (x,s]$ and every $Y \in \mathcal{V}_{x,t}$, Υ_t^Y lies to the left of C_{s+1}, and so $Y \in \mathcal{E}_{s+1}$; on the other hand $Y \notin \mathcal{E}_t$ and so $Y \notin \mathcal{E}_{x+1}$. It follows that

$$\mathbf{c}_A(x,s) \leqslant \mu \left(\mathcal{E}_{s+1} \setminus \mathcal{E}_{x+1} \right) = \mathbf{c}_\varepsilon(x,s).$$

By Lemma 3.2, $\mathbf{c}_\Omega \to \mathbf{c}_\varepsilon$, and so

$$\mathbf{c}_A \langle A_s \rangle \leqslant \mathbf{c}_\varepsilon \langle A_s \rangle \leqslant^\times \mathbf{c}_\Omega \langle A_s \rangle,$$

and we assumed that the latter is finite.

Next we show that every random real that computes A is captured by some $\mathbf{c}_A$-bounded test. Since $C \leqslant_{\mathrm{ML}} A$, every such real computes C. By Lemma 3.1, it suffices to build a $\mathbf{c}_A$-test capturing any random Y such that $C = \Upsilon^Y$. The desired test is the test $\mathcal{U}_k = \bigcup_{s>k} \mathcal{V}_{k,s}$ defined above. (Again, we assume that we delay computations, so for all Y and s, Υ_s^Y does not lie to the right of C_s.) □

As promised, in the case that A is c.e., $\mathbf{c}_A$ is the strongest cost function that A obeys. In particular, $\mathbf{c}_A \to \mathbf{c}_\Omega$.

Proposition 4.2. *Suppose that A is c.e. For any cost function $\mathbf{c}$ such that $A \models \mathbf{c}$, we have $\mathbf{c}_A \to \mathbf{c}$.*

Proof. After multiplying $\mathbf{c}$ by a constant, we may assume that $\underline{\mathbf{c}}(0) < 1/2$. Fix a computable speed-up f such that $\mathbf{c}\langle A_{f(s)} \rangle < 1/2$ (again, see [20]). Define a Turing functional Γ such that at every stage t,

$$\mu\left(\{Y : A_{f(t)} \upharpoonright x+1 \prec \Gamma_t^Y\} - \mathcal{E}_{\Gamma,t}\right) = \mathbf{c}(x,t),$$

where $\mathcal{E}_{\Gamma,t} = \{Y : \Gamma_t^Y$ lies to the left of $A_{f(t)}\}$. By a simple argument $\mu(\mathcal{E}_{\Gamma,t}) \leqslant \mathbf{c}\langle A_{f(s)} \rangle < 1/2$ for every t, so this construction may proceed.

Fix e with $\Phi_e = \Gamma$. Then

$$\begin{aligned}
\underline{\mathbf{c}}_A(x) &= \mu\left(\bigcup_{x<t} \mathcal{V}_{x,t}\right) \\
&= \mu\left(\bigcup_{x<t} \left\{Y : A_t \upharpoonright x+1 \preccurlyeq \Upsilon_t^Y\right\}\right) \\
&\geqslant \mu\left\{Y : A \upharpoonright x+1 \preccurlyeq \Upsilon^Y\right\} \\
&\geqslant 2^{-(e+1)} \cdot \mu\left\{Y : A \upharpoonright x+1 \preccurlyeq \Upsilon^{0^e 1 \hat{\ } Y}\right\} \\
&\geqslant 2^{-(e+1)} \underline{\mathbf{c}}(x).
\end{aligned}$$

□

Recall that $T(A)$ is the shift of A.

Theorem 4.3. *For every cost function $\mathbf{d}$ there is a cost function $\mathbf{c} \geqslant \mathbf{d}$ and a c.e. set A such that $A \models \mathbf{c}$ and $T(A) \not\models \mathbf{c}$.*

Since $\mathbf{c}_A \to \mathbf{c}$, this shows that $T(A) \not\models \mathbf{c}_A$.

Proof. The main idea is to enumerate the set A and the cost function $\mathbf{c}$ so that it has "sudden drops": numbers x with $\underline{\mathbf{c}}(x)$ much smaller than $\underline{\mathbf{c}}(x-1)$.

Let $\langle B_t^0 \rangle, \langle B_t^1 \rangle, \ldots$ be a listing of all (possibly partial) computable enumerations. In particular, let $\langle D_n \rangle$ be an effective listing of the finite sets, and let $B_{t+1}^e = B_t^e \cup D_{\varphi_e(t+1)}$, where defined.

At a stage s, we may declare $\mathbf{c}(s-1,s) \geqslant \alpha$ for some dyadic rational α, which by monotonicity entails that $\mathbf{c}(y,t) \geqslant \alpha$ for each $y < s$ and $t \geqslant s$. At the end of stage s, we will define $\mathbf{c}(x,s)$ for every $x < s$ to be the least value consistent with all of our declarations and also with $\mathbf{c}(x,s) \geqslant \mathbf{d}(x,s)$.

We must meet the global requirements that $\mathbf{c}$ has the limit condition and that $A \models \mathbf{c}$. We must also meet the requirements

$$R_e\colon\ T(A) = \bigcup_t B_t^e \Rightarrow \mathbf{c}\langle B_t^e \rangle \geqslant 1.$$

The strategy for R_e seeks to find an x and an s where $\mathbf{c}(x-1,s)$ is large and $x-1 \notin B_s^e$. Then it enumerates x into A and waits until it sees a $t > s$ with $x-1 \in B_t^e$. This will increase $\mathbf{c}\langle B_t^e \rangle$ by at least $\mathbf{c}(x-1,s)$. Then the strategy seeks to repeat the process with a new x, continuing until $\mathbf{c}\langle B_t^e \rangle \geqslant 1$.

To ensure that $\mathbf{c}$ has the limit condition, we will give R_e a bound α_e beyond which it is not allowed to increase $\mathbf{c}$. This bound will also ensure that R_e does not interfere with $R_{e'}$ for $e' < e$. To ensure that $A \models \mathbf{c}$, we will not allow R_e to cause enumerations with total cost exceeding 2^{-e}. Other than a discussion of α_e, our full strategy for R_e is:

(1) Let s be the current stage. Declare $\mathbf{c}(s-1,s) \geqslant \alpha_e$.
(2) At stage $s+1$, declare $\mathbf{c}(s,s+1) \geqslant 2^{-e} \cdot \alpha_e$.
(3) Wait for a stage $u > s$ when one of the following happens:
   (a) If $\mathbf{c}(s,u) > 2^{-e} \cdot \alpha_e$, return to Step (1).
   (b) If s is enumerated into A, return to Step (1).
   (c) If B^e_s converges with $s-1 \notin B^e_s$, enumerate s into A and proceed to Step (4).
(4) Wait until B^e_r converges for some $r > s$ with $s-1 \in B^e_r$.
(5) If $\mathbf{c}\langle B^e_t \rangle^r_{t=0} \geqslant 1$, terminate the strategy. Otherwise, return to Step (1).

Note that case (3a) might occur because of the actions of some other strategy, or might instead occur because of $\mathbf{c}(s,u) \geqslant \mathbf{d}(s,u)$. The latter can occur only finitely many times, because $\mathbf{d}$ satisfies the limit condition.

Note also that if we reach Step (5), then $s-1 \notin B^e_s$, $s-1 \in B^e_r$, and $\mathbf{c}(s-1,s) \geqslant \alpha_e$, so $\mathbf{c}\langle B^e_t \rangle^r_{t=0} - \mathbf{c}\langle B^e_t \rangle^s_{t=0} \geqslant \alpha_e$. Thus we will reach Step (5) at most $1/\alpha_e$ times before meeting the requirement and terminating the strategy. Each enumeration has a cost of $2^{-e} \cdot \alpha_e$ by construction, and so the total cost of enumerations by this strategy is at most 2^{-e}.

If the strategy waits forever at Step (3), then either $\langle B^e_t \rangle$ is partial, or $s \notin A$ but $s-1 \in \bigcup_t B^e_t$, meaning we satisfy R_e by negating the hypothesis. It thus remains only to show that we do not return to Step (1) via case (3a) or (3b) infinitely many times.

We wish to ensure that no $R_{e'}$-strategy for $e' > e$ can increase $\mathbf{c}(s,u)$ beyond $2^{-e} \cdot \alpha_e$. So we define $\alpha_0 = 1$, $\alpha_{e+1} = 2^{-e} \cdot \alpha_e$. Now case (3a) cannot be caused by the action of any $R_{e'}$-strategy for $e' > e$. Nor can case (3b), because of our action at Step (2). It is then a simple induction that no strategy returns to Step (1) more than finitely many times. □

5. The computational strength of fragments of Ω

In this section, we introduce ML*-reducibility (Definition 5.5), a natural dual of ML-reducibility that compares Martin-Löf random sequences according to the K-trivial sets that they compute. We use this notion to compare the relative strength of fragments of Ω, by which we mean the restriction of the bit-sequence Ω to a computable set R of locations. Theorem 5.6 gives a characterisation of the ML*-strength of such a fragment based on the growth of the function $m \mapsto |R \cap m|$. Theorem 5.11 characterises the K-trivial sets computable from a fragment of Ω using an appropriate cost function.

A motivating application. Before diving in, we discuss in some detail one of the applications that motivated these results. A set A is a *k/n-base* if it is computable from the join of any k of the n-columns of some random sequence X, in all possible ways. For a computable real p such that $0 < p \leqslant 1$, let $\mathbf{c}_{\Omega,p}(x,s) = (\Omega_s - \Omega_x)^p$. As mentioned in the introduction, three of the authors of the present paper proved:

Theorem 5.1 ([11])**.** *The following are equivalent for a set A and $1 \leqslant k < n$:*

(1) *A is a k/n-base.*
(2) *A is a k/n-base witnessed by Ω, i.e., it is computable from the join of any k of the n-columns of Ω.*
(3) *A obeys $\mathbf{c}_{\Omega,k/n}$.*

Hence the p-bases are characterised by cost functions. Theorem 3.3 implies that, for every rational $p \in (0,1)$, there is a smart p-base: a greatest ML-degree of p-bases. If $p < q$, then every p-base is also a q-base, as $\mathbf{c}_{\Omega,p} \geqslant^\times \mathbf{c}_{\Omega,q}$. However there also is a q-base that is not a p-base. Thus, the smart p-bases form a dense chain of ML-degrees.

One application of the results in this section will be to add another equivalent condition to Theorem 5.1:

(4) *A is K-trivial and is computable from the join of some choice of k of the n-columns of Ω.*

In other words: if a K-trivial is computable from *some* k/n-fragment of Ω, then it is computable from *any* k/n-fragment of Ω. Hirschfeldt, Nies and Stephan [15] proved that any c.e. set computable from a Turing incomplete random set is K-trivial. Since every k/n-fragment of Ω is incomplete, we obtain:

Corollary 5.2. *If X and Y are both k/n-fragments of Ω for $k < n$, then X and Y compute the same c.e. sets.*

In particular, if a c.e. set is computable from one half of Ω, it is also computable from the other half.

The generalised cost function. We now turn to the general analysis of the computational power of fragments of Ω. Recall that for infinite R, we denote by Ω_R the real obtained by erasing the bits of Ω in locations outside of R. For example, if $1 \leqslant j \leqslant n$, then the j^{th} n-column of Ω is $\Omega_{(j-1+n\mathbb{N})}$. For $n \geqslant 1$ and $T \subseteq \{1, 2, \dots, n\}$, let $R(T,n) = \bigcup_{j\in T} j - 1 + n\mathbb{N}$. So $\Omega_{R(T,n)}$ is the join of the n-columns of Ω indexed by T (up to a simple computable permutation, depending on how we take the join).

Let R be an infinite computable set. The first question is how to generalise the cost function $\mathbf{c}_{\Omega,k/n}$ to a cost function $\mathbf{c}_{\Omega,R}$. A basic step in the analysis of p-bases was the observation that if $T \subseteq \{1, 2, \dots, n\}$ has size k, then $\Omega_{R(T,n)}$ is captured by a $\mathbf{c}_{\Omega,k/n}$-test; this gave the implication (3)→(2) of Theorem 5.1. We want to generalise this; in particular, we would like to capture the bits of Ω_R that are given by $\Omega \upharpoonright n$ by the n^{th} component of a $\underline{\mathbf{c}}_{\Omega,R}$-test. So perhaps the first guess would be to define $\underline{\mathbf{c}}_{\Omega,R}(n) = (\Omega - \Omega_n)^{|R\cap n|/n}$. It turns out that this is not quite right; it works if $R = R(T,n)$, but that is misleading because in that case the density of initial segments is more or less constant k/n. What would work is $\underline{\mathbf{c}}_{\Omega,R}(n) = (\Omega - \Omega_n)^{|R\cap k(n)|/k(n)}$, where $\Omega - \Omega_n \in (2^{-k(n)-1}, 2^{-k(n)}]$. However, it is not clear that this cost function will be monotonic if the density of R varies. We get around this annoyance by using a "discrete" version.

For $n < \omega$, let

$$k(n) = \lfloor -\log_2(\Omega - \Omega_n) \rfloor,$$

so $2^{-k(n)-1} < \Omega - \Omega_n \leqslant 2^{-k(n)}$. Define $k_s(n)$ similarly, replacing Ω by Ω_s. Note that $k(n) \leqslant n$ for all but finitely many n (otherwise, Ω would not be random).

Now for an infinite computable $R \subseteq \omega$, define

$$\mathbf{c}_{\Omega,R}(n,s) = 2^{-|R\cap k_s(n)|}.$$

The cost function $\mathbf{c}_{\Omega,R}$ is monotonic: $k_{s+1}(n) \leqslant k_s(n)$ and $k_s(n+1) \geqslant k_s(n)$. It also satisfies the limit condition: $\underline{\mathbf{c}}_{\Omega,R}(n) = 2^{-|R\cap k(n)|}$ is finite and, since $\lim_n k(n) = \infty$ and R is infinite, $\lim_n \underline{\mathbf{c}}_{\Omega,R}(n) = 0$. Finally, we note that $\mathbf{c}_{\Omega,R}(n,s) \cdot \mathbf{c}_{\Omega,R^\complement}(n,s) = 2^{-k_s(n)} =^\times \Omega_s - \Omega_n$, where $R^\complement$ is the complement of R.

Remark 5.3. For any infinite R, $\underline{\mathbf{c}}_{\Omega,R}(n) =^\times (\Omega - \Omega_n)^{|R\cap k(n)|/k(n)}$. This can be checked by taking logarithms. Hence, if $T \subseteq \{1,2,\ldots,n\}$ has size k, then $\underline{\mathbf{c}}_{\Omega,R(T,n)} =^\times \underline{\mathbf{c}}_{\Omega,k/n}$.

Proposition 5.4. *Ω_R is captured by a $\mathbf{c}_{\Omega,R}$-test.*

Proof. We use the idea from the proof in [11] that p-OW tests are covered by $\mathbf{c}_{\Omega,p}$-tests. For each $\sigma \in 2^{<\omega}$, let

$$G_\sigma = \{X \in 2^\omega : (\forall m < |R \cap |\sigma||)\ X(m) = \sigma(p_R(m))\},$$

where $p_R(m)$ is the m^{th} element of R. Note that $\mu(G_\sigma) = 2^{-|R\cap|\sigma||}$.

Let $U_n = \bigcup_{s\geqslant n} G_{\Omega_s\upharpoonright n}$. Then $\Omega_R \in \bigcap_n U_n$. Note also that $U_{n+1} \subseteq U_n$. Since $\Omega - \Omega_n \leqslant 2^{-k(n)}$, the set $\{\Omega_s \upharpoonright k(n) : s \geqslant n\}$ contains at most two strings. But $U_n \subseteq \bigcup_{s\geqslant n} G_{\Omega_s\upharpoonright k(n)}$, so

$$\mu(U_n) \leqslant 2\cdot 2^{-|R\cap k(n)|} = 2\cdot \underline{\mathbf{c}}_{\Omega,R}(n).$$

□

The dual of ML-reducibility. As already mentioned, to analyse the interplay between fragments of Ω, we use the dual of ML-reducibility.

Definition 5.5. For random sequences Y and Z, we write $Y \leqslant_{\mathrm{ML}*} Z$ if for every K-trivial set A, if $A \leqslant_{\mathrm{T}} Y$ then $A \leqslant_{\mathrm{T}} Z$.

Again Turing reducibility implies ML*-reducibility. The top degree consists of those randoms that compute all K-trivial sets; these are the randoms that fail some $\mathbf{c}_\Omega$-test (i.e., the non-*Oberwolfach randoms* [3]). Of course, these include all the Turing complete randoms. The bottom degree consists of the weakly 2-random sequences, the randoms that compute no K-trivial sets. As we discussed above, it will be the case that, for $1 \leqslant k \leqslant n$, any two k/n-fragments of Ω have the same ML*-degree.

What we prove is far more general. Our equivalence (i)↔(iii) below provides a complete characterisation of ML*-reducibility between fragments of Ω by a simple combinatorial condition on the underlying computable sets: $\Omega_S \leqslant_{\mathrm{ML}*} \Omega_R$ iff for each number m, the size of S below m exceeds the size of R below m by at most a constant. The intuition is that as R gets thinner, Ω_R gets computationally weaker (in the coarse sense of ML*). By the randomness enhancement principle [19], this means that Ω_R also gets more random.

Theorem 5.6. *The following are equivalent for infinite computable sets R and S:*

(i) $\Omega_S \leqslant_{\mathrm{ML}*} \Omega_R$.
(ii) *Ω_R is captured by a $\mathbf{c}_{\Omega,S}$-test.*
(iii) $|S \cap m| \leqslant^+ |R \cap m|$.
(iv) $\mathbf{c}_{\Omega,S} \to \mathbf{c}_{\Omega,R}$.

Note that the application from the beginning of the section follows easily: if $T, T' \subseteq \{1,2,\ldots,n\}$ have size k, then $|R(T,n) \cap m| =^+ |R(T',n) \cap m|$. Therefore, any k-trivial computable from $\Omega_{R(T,n)}$ is computable from every k/n-fragment of Ω, hence is a k/n-base.

To prove (iv)→(i), we rely on a lemma of interest on its own. The proof will be given in the next section.

Lemma 5.7. *Let $R \subseteq \omega$ be computable and co-infinite. Suppose that $X \oplus Y$ is ML-random, and that X fails a $\mathbf{c}_{\Omega,R^{\complement}}$-test. Suppose that A is K-trivial, and that $A \leqslant_{\mathrm{T}} Y$. Then $A \models \mathbf{c}_{\Omega,R}$.*

Remark 5.8. While $\mathbf{c}_{\Omega,R}$ was only defined for infinite R, the definition can be interpreted for finite R, in which case the cost function does not satisfy the limit condition, and the sets obeying it will be the computable ones. Lemma 5.7 holds for R finite or co-finite as well. The case $|R^{\complement}| < \infty$ tells us nothing: the hypothesis that X fails a $\mathbf{c}_{\Omega,R^{\complement}}$-test is trivial, since there is such a test that captures the entire interval; meanwhile, $\mathbf{c}_{\Omega,R} =^{\times} \mathbf{c}_{\Omega}$, so the conclusion $A \models \mathbf{c}_{\Omega,R}$ is simply a restatement of the fact that A is K-trivial.

The case $|R| < \infty$ is a weaker version of a known result: the assumption that X fails a $\mathbf{c}_{\Omega,R^{\complement}}$-test tells us that X is not $\mathbf{c}_{\Omega}$-random, and thus that X is LR-hard [3]. Since Y is X-random, Y is 2-random, and so the only K-trivials that Y computes are the computable sets.

Due to the relative length of the proof, we state and prove the implication (ii)→(iii) of Theorem 5.6 separately.

Proposition 5.9. *Let R and S be infinite computable sets such that $|S \cap m| \leqslant^{+} |R \cap m|$. Then Ω_R is $\mathbf{c}_{\Omega,S}$-random.*

Proof. Suppose for a contradiction that Ω_R can be captured by a $\mathbf{c}_{\Omega,S}$-test. Then there is a nested test $\langle \mathcal{U}_n \rangle$ capturing the sequence $Z = \Omega_R \oplus \Omega_{R^{\complement}}$ with $\mu(\mathcal{U}_n) \leqslant (\underline{\mathbf{c}}_{\Omega,S}(n)) \cdot (\underline{\mathbf{c}}_{\Omega,R^{\complement}}(n))$. We show that this implies that Z is not ML-random. To do this, we show how to uniformly enumerate an open set $\mathcal{V}$ of small measure that contains Z.

Note that $|S \cap m| - |R \cap m| = |S \cap m| + |R^{\complement} \cap m| - m$. Given a rational $\varepsilon > 0$, we can thus effectively find a k with $|S \cap k| + |R^{\complement} \cap k| - k > 1 - \log \varepsilon$.

Define a location $n_s > k$ recursively at stages $s \geqslant k$. Recall that $k_s(n) = \lfloor -\log_2(\Omega_s - \Omega_n) \rfloor$. For $s = k$, or if $k_{s+1}(n_s) < k$, we let n_{s+1} be the least $n \geqslant k$ such that $k_{s+1}(n) > k$. Otherwise, we let $n_{s+1} = n_s$.

There are at most 2^{k+1} stages s at which $n_s \neq n_{s-1}$. For let $s < t$ be two such stages, then $\Omega_s - \Omega_{n_s} \leqslant 2^{-(k+1)}$. But $\Omega_t - \Omega_{n_s} > 2^{-k}$, so $\Omega_t - \Omega_s > 2^{-(k+1)}$.

Let $\mathcal{V} = \bigcup_{s>k} \mathcal{U}_{n_s,s}$. For each stage $s \geqslant k$, we have $k_s(n_s) \geqslant k$, so

$$\mu(\mathcal{U}_{n_s,s}) \leqslant (\mathbf{c}_{\Omega,S}(n_s,s)) \cdot (\mathbf{c}_{\Omega,R^{\complement}}(n_s,s)) = 2^{-|S \cap k_s(n_s)|} \cdot 2^{-|R^{\complement} \cap k_s(n_s)|} \leqslant 2^{-|S \cap k|} \cdot 2^{-|R^{\complement} \cap k|} = 2^{-(|S \cap k| + |R^{\complement} \cap k|)} < 2^{\log \varepsilon - k - 1} = 2^{-k-1} \cdot \varepsilon.$$

Therefore,

$$\mu(\mathcal{V}) \leqslant 2^{k+1} \cdot 2^{-k-1} \cdot \varepsilon = \varepsilon. \qquad \square$$

Proof of Theorem 5.6. (i)→(ii) Let A be smart for $\mathbf{c}_{\Omega,S}$ by Theorem 3.3. Then $A \models \mathbf{c}_{\Omega,S}$. Note that Ω_S is captured by a $\mathbf{c}_{\Omega,S}$-test by Proposition 5.4. By Proposition 1.5, $A \leqslant_T \Omega_S$, and so $A \leqslant_T \Omega_R$. By the definition of smartness for cost functions, Ω_R is captured by a $\mathbf{c}_{\Omega,S}$-test.

(ii)→(iii) This is the contrapositive of Proposition 5.9.

(iii)→(iv) Fix b such that $|S \cap m| \leqslant |R \cap m| + b$ for all m. Then $|S \cap k(n)| \leqslant |R \cap k(n)| + b$ for all n, meaning that $\underline{\mathbf{c}}_{\Omega,R} \leqslant 2^{b} \underline{\mathbf{c}}_{\Omega,S}$, and so $\mathbf{c}_{\Omega,S} \to \mathbf{c}_{\Omega,R}$.

(iv)→(i) Suppose $\mathbf{c}_{\Omega,S} \to \mathbf{c}_{\Omega,R}$, or equivalently $\underline{\mathbf{c}}_{\Omega,R} \leqslant^{\times} \underline{\mathbf{c}}_{\Omega,S}$. Since

$$\underline{\mathbf{c}}_{\Omega,R} \cdot \underline{\mathbf{c}}_{\Omega,R^{\complement}} =^{\times} \underline{\mathbf{c}}_{\Omega} =^{\times} \underline{\mathbf{c}}_{\Omega,S} \cdot \underline{\mathbf{c}}_{\Omega,S^{\complement}},$$

it follows that $\underline{\mathbf{c}}_{\Omega,S^{\complement}} \leqslant^{\times} \underline{\mathbf{c}}_{\Omega,R^{\complement}}$. By Proposition 5.4, $\Omega_{S^{\complement}}$ is captured by a $\mathbf{c}_{\Omega,S^{\complement}}$-test, and so it is captured by a $\mathbf{c}_{\Omega,R^{\complement}}$-test. By Lemma 5.7, for any K-trivial $A \leqslant_{\mathrm{T}} \Omega_S$, $A \models \mathbf{c}_{\Omega,R}$. Thus $A \leqslant_T \Omega_R$ by Proposition 1.5, and so $\Omega_R \geqslant_{\mathrm{ML}*} \Omega_S$. $\square_{5.6}$

The dual of smartness. As a consequence of Lemma 5.7, we obtain a natural characterisation of the K-trivial sets computable from Ω_R for an infinite computable set R. We express it using a notion dual to smartness:

Definition 5.10. A ML-random sequence Y is *dumb* for a cost function $\mathbf{c}$ if Y fails a $\mathbf{c}$-test, and for any K-trivial set A,

$$A \models \mathbf{c} \Leftrightarrow A \leqslant_{\mathrm{T}} Y.$$

That is, the only K-trivial sets that Y computes are the ones that it has to compute because they obey $\mathbf{c}$.

The following can be seen as a general form of (3)↔(4) from Theorem 5.1.

Theorem 5.11. *If R is an infinite computable set, then Ω_R is dumb for $\mathbf{c}_{\Omega,R}$. (So the K-trivials computable from Ω_R are exactly those that obey $\mathbf{c}_{\Omega,R}$.)*

Proof. By Proposition 5.4, Ω_R fails a $\mathbf{c}_{\Omega,R}$-test. So by Proposition 1.5, if $A \models \mathbf{c}_{\Omega,R}$, then it is computable from Ω_R. The other direction is Lemma 5.7: $\Omega_R \oplus \Omega_{R^{\complement}}$ is ML-random and $\Omega_{R^{\complement}}$ fails a $\mathbf{c}_{\Omega,R^{\complement}}$-test, so if $A \leqslant_T \Omega_R$ is K-trivial, then $A \models \mathbf{c}_{\Omega,R}$. $\square$

The next two lemmas are immediate from the definitions of dumb and smart (and Proposition 1.5). They tell us that cost functions that admit dumb sequences are special. The first lemma implies that if $\mathbf{c}$ has a dumb random sequence, then the collection of sets that obey $\mathbf{c}$ is a principal ideal of ML-degrees.

Lemma 5.12. *Let $\mathbf{c}$ be a cost function such that $\mathbf{c} \to \mathbf{c}_{\Omega}$. Suppose that Z is dumb for $\mathbf{c}$ and that B is smart for $\mathbf{c}$. Then the following are equivalent for a K-trivial set A:*

(1) $A \models \mathbf{c}$;
(2) $A \leqslant_{\mathrm{T}} Z$;
(3) $A \leqslant_{\mathrm{ML}} B$.

Thus, for example, no random can be dumb for the cost function $\mathbf{c}_A$ for A constructed for Theorem 4.3.

Lemma 5.13. *Let $\mathbf{c}$ and $\mathbf{d}$ be cost functions. Suppose that $\mathbf{c} \to \mathbf{c}_{\Omega}$ and $\mathbf{d} \to \mathbf{c}_{\Omega}$. Suppose that $Z_{\mathbf{c}}$ and $Z_{\mathbf{d}}$ are dumb for $\mathbf{c}$ and $\mathbf{d}$ (respectively), and that $B_{\mathbf{c}}$ and $B_{\mathbf{d}}$ are smart for $\mathbf{c}$ and $\mathbf{d}$ (respectively). The following are equivalent:*

(1) $\mathbf{c} \to \mathbf{d}$;
(2) $Z_{\mathbf{c}} \geqslant_{\mathrm{ML}*} Z_{\mathbf{d}}$;
(3) $B_{\mathbf{c}} \leqslant_{\mathrm{ML}} B_{\mathbf{d}}$.

Incomparable ML-degrees. As a consequence of the foregoing facts and results, we obtain a structural result for the ML-degrees.

Theorem 5.14. *There is an infinite antichain of ML-degrees of K-trivial sets. In fact, every countable partial ordering is embeddable in the ML-degrees of K-trivial sets.*

Proof. We fix a uniformly computable partition of $\mathbb{N}$ into countably many sets R_n such that the upper density of each R_n is 1 (greater than 1/2 would do). For a computable set $F \subseteq \mathbb{N}$, let $R(F) = \bigcup_{n \in F} R_n$, and let B_F be a K-trivial set that is smart for $\mathbf{c}_{\Omega,R(F)}$. The map $F \mapsto B_F$ is an embedding of the partial ordering of computable sets under inclusion into the ML-degrees. This suffices, as it is easy to embed the countable atomless Boolean algebra into the algebra of computable sets under inclusion.

To see that the map is a partial order embedding, first suppose that $F \subseteq G$. Then $R(F) \subseteq R(G)$, and so $\mathbf{c}_{\Omega,R(F)} \geqslant \mathbf{c}_{\Omega,R(G)}$; by Lemma 5.13, $B_F \leqslant_{\mathrm{ML}} B_G$.

On the other hand, if $F \nsubseteq G$, take some $n \in F \smallsetminus G$; so $R_n \subseteq R(F)$ but $R_n \cap R(G) = \varnothing$. The fact that the upper density of R_n is 1 implies that $|R(F) \cap m| \nleqslant^+ |R(G) \cap m|$. By Theorem 5.6 and Theorem 5.11, and Lemma 5.13, $B_F \nleqslant_{\mathrm{ML}} B_G$. □

We finish the section with another structural result about the ML-degrees, although one that does not use the machinery developed above. We prove downward density, and give an alternative construction of incomparable ML-degrees.

Theorem 5.15. *For every non-computable c.e. set D, there are c.e. sets $A, B \leqslant_{\mathrm{T}} D$ such that $A \mid_{\mathrm{ML}} B$.*

Proof. We extend Kucera's injury-free proof, as published in [18, p. 154], of the Friedberg–Muchnik theorem, which states that there are Turing incomparable c.e. sets A, B. Two versions of the proof are given; the version relying on [18, Cor. 4.2.5] actually shows that there are ML-random Δ^0_2 sets Y, Z such that $A \leqslant_{\mathrm{T}} Y$, $B \leqslant_{\mathrm{T}} Z$, $A \nleqslant_{\mathrm{T}} Z$, and $B \nleqslant_{\mathrm{T}} Y$. Therefore $A \mid_{\mathrm{ML}} B$ as witnessed by Y, Z.

To ensure that $A, B \leqslant_{\mathrm{T}} D$, all we need to do is modify [18, Cor. 4.2.5]:

Claim 5.15.1. There is a computable function r such that for each e, if $Y = \Phi_e^{\varnothing'}$ is total and ML-random, then $A = W_{r(e)} \leqslant_{\mathrm{wtt}} Y$, $A \leqslant_{\mathrm{T}} D$, and A is non-computable.

To see this, we use the cost function version of Kucera's result as presented in [12] and [18, 5.3.13]. Given a ML-random Δ^0_2 set Y, one defines a cost function c_Y such that if $A \models c_Y$, then $A \leqslant_{\mathrm{wtt}} Y$. The cost function c_Y emulates a given computable approximation of Y, and is therefore obtained uniformly from an e such that $Y = \Phi_e^{\varnothing'}$. The construction of a non-computable c.e. set A obeying a given cost function with the limit condition [20, Thm. 2.7(i)] is compatible with simple permitting, so we can ensure that $A \leqslant_{\mathrm{T}} D$. It is also uniform in the cost function. So we obtain the c.e. set A uniformly in e, as required. □

6. Proof of Lemma 5.7

Recall that for $\mathcal{A} \subseteq 2^\omega$ and $Z \in 2^\omega$, the Lebesgue (binary) lower density $\varrho(\mathcal{A}|Z)$ of $\mathcal{A}$ at Z is $\liminf_n \mu(\mathcal{A}|Z \upharpoonright n)$, where $\mu(\mathcal{A}|\sigma) = \mu(\mathcal{A} \cap [\sigma])/\mu([\sigma])$ is the conditional probability of $\mathcal{A}$ given $[\sigma]$.

We require two facts from [4]. The first implies that *difference random* sequences are *positive density* points. A difference test is one of the form $\langle \mathcal{U}_n \cap \mathcal{P} \rangle$, where the open sets $\mathcal{U}_n$ are uniformly Σ^0_1 and nested, $\mathcal{P}$ is Π^0_2, and $\mu(\mathcal{U}_n \cap \mathcal{P}) \leqslant 2^{-n}$. A sequence Z is a positive density point if the lower density $\underline{\varrho_2}(\mathcal{P}|Z)$ is positive for any Π^0_1 class $\mathcal{P}$ that contains Z. The first fact says that the aforementioned implication (which is actually an equivalence for ML-random sequences) is witnessed on the same Π^0_1 class.

Fact 6.1 ([4], Lemma 3.3)**.** *Suppose that $\mathcal{Q}$ is a Π^0_1-class and $\langle \mathcal{V}_n \cap \mathcal{Q} \rangle$ is a difference test that captures a ML-random sequence Z. Then $\mathcal{Q}$ has lower density 0 at Z.*

Recall that a set Z is *LR-hard* if Z-randomness implies $\varnothing'$-randomness, that is, 2-randomness. A sequence Z is a *density 1 point* if $\varrho(\mathcal{P}|Z) = 1$ for every Π^0_1 class containing Z.

Fact 6.2 ([4], Thm. 3.6)**.** *A ML-random sequence that is not LR-hard is a density 1 point.*

We first give a proof of Lemma 5.7 in the case that A is c.e.

Lemma 5.7. *Let $R \subseteq \omega$ be computable and co-infinite. Suppose that $X \oplus Y$ is ML-random, and that X fails a $\mathbf{c}_{\Omega,R^\complement}$-test. Suppose that A is K-trivial, and that $A \leqslant_{\mathrm{T}} Y$. Then $A \models \mathbf{c}_{\Omega,R}$.*

Proof when A is c.e. Fix a computable enumeration $\langle A_s \rangle$ of A. Fix a $\mathbf{c}_{\Omega,R^\complement}$-test $\langle \mathcal{U}_n \rangle$ that X fails, and fix a functional Φ with $A = \Phi^Y$. Let $\mathcal{E}$ be the error set for Φ with respect to A: $\mathcal{E}_s$ is the set of oracles Z such that Φ^Z_s lies to the left of A_s. Let $\mathcal{Q} = 2^\omega \times (2^\omega - \mathcal{E})$ (and $\mathcal{Q}_s = 2^\omega \times (2^\omega - \mathcal{E}_s)$).

We carry out a "ravenous sets" construction on $\mathcal{Q}$, similar to [11, Section 3]. Uniformly in $k, n \in \omega$, we enumerate Σ^0_1 open sets $\mathcal{V}^k_n \subset 2^\omega \times 2^\omega$. The *goal* for $\mathcal{V}^k_n \cap \mathcal{Q}$ is $2^{-k}(\Omega_{n+1} - \Omega_n)$; we will ensure that no set ever exceeds its goal. (The sets $\mathcal{V}^k_n$ are called "ravenous", rather than just "hungry" as in [15], because we may feed them with oracle strings that later leave $\mathcal{Q}$, in which case they get hungry again.) We will also ensure that $\mathcal{V}^k_n$ is disjoint from $\mathcal{V}^k_m$ for $n \neq m$. The parameter k determines the goal for these ravenous sets; otherwise, the constructions for distinct k are independent. The other property that we ensure is that

$$\mathcal{V}^k_n \cap \mathcal{Q} \subseteq \mathcal{U}_n \times \Psi^{-1}[A \upharpoonright n+1].$$

At stage 0, we begin with $\mathcal{V}^k_n$ empty for every $k, n \in \omega$. Fixing k, at every stage s, one of the sets $\mathcal{V}^k_n$ will be "awake" (and the others "asleep"). We start with $\mathcal{V}^k_0$ awake. At stage s, if $\mathcal{V}^k_n$ is awake at this stage and it has not reached its goal, i.e., $\mu(\mathcal{V}^k_{n,s} \cap \mathcal{Q}_s) < 2^{-k}(\Omega_{n+1} - \Omega_n)$, then we try to feed it: we add to $\mathcal{V}^k_{n,s+1}$ clopen sets of the form $[\sigma] \times [\tau]$, where $[\sigma] \subseteq \mathcal{U}_{n,s}$ and $A_s \upharpoonright n+1 \leqslant \Phi^\tau_s$, but we ensure that the goal is not exceeded, and that the cylinders added are disjoint from $\mathcal{V}^k_{m,s}$ for all $m \neq n$. If $\mu(\mathcal{V}^k_{n,s} \cap \mathcal{Q}_s) = 2^{-k}(\Omega_{n+1} - \Omega_n)$ then we put $\mathcal{V}^k_n$ to sleep[1] and declare $\mathcal{V}^k_m$ to be awake at stage $s+1$, where m is least such that $\mathcal{V}^k_{m,s}$ has not reached *half* its goal: $\mu(\mathcal{V}^k_{m,s} \cap \mathcal{Q}_s) < 2^{-(k+1)}(\Omega_{n+1} - \Omega_n)$. Such m will always exist, of course, because all but finitely many $\mathcal{V}^k_{m,s}$ will be empty. It is important to note, though, that as we enumerate measure into $\mathcal{V}^k_n$, measure leaves $\mathcal{Q}$, and so a ravenous set $\mathcal{V}^k_n$ could be put to sleep but re-awakened later.

Let $\mathcal{V}^k = \bigcup_n \mathcal{V}^k_n$. Since $\sum_n (\Omega_{n+1} - \Omega_n)$ is a telescopic series, with sum Ω, $\left\langle \mathcal{V}^k \cap \mathcal{Q} \right\rangle$ is a difference test. Since X fails a $\mathbf{c}_{\Omega,R^\complement}$ test, it is not 2-random, and so Y is not LR-hard. By Fact 6.2, Y is a density 1 point. Noticing that $Y \notin \mathcal{E}$, in particular, $2^\omega - \mathcal{E}$ has density 1 at Y; so $\mathcal{Q}$ has density 1 at $X \oplus Y$, and so certainly not lower-density 0. By Fact 6.1, $X \oplus Y \notin \bigcap_k \mathcal{V}^k \cap \mathcal{Q}$. Since $X \oplus Y \in \mathcal{Q}$, we can fix some k with $X \oplus Y \notin \mathcal{V}^k_n$ for any $n \in \omega$. In the remainder of the proof, we omit the superscript k.

Claim 5.7.1. For every n, there is a stage t such that for all $s \geqslant t$, $\mu(\mathcal{V}_{n,s} \cap \mathcal{Q}_s) \geqslant 2^{-(k+1)}(\Omega_{n+1} - \Omega_n)$.

[1]Magically, the sets go to sleep when they are told to.

Proof. For every n, there is a $\sigma \prec X$ with $[\sigma] \subseteq \mathcal{U}_n$, and there is a $\tau \prec Y$ with $A \upharpoonright n+1 \leqslant \Phi^\tau$. We never enumerate $[\sigma] \times [\tau]$ into $\mathcal{V}_n$ for any such pair (σ, τ). But if $\mathcal{V}_n$ is at some point awake and is never put to sleep, then it would eventually enumerate such a pair.

If s_0 is a stage when $\mathcal{V}_n$ goes to sleep and $s_1 > s_0$ is a stage at which $\mathcal{V}_n$ wakes back up, then $\mu(\mathcal{Q}_{s_0} - \mathcal{Q}_{s_1}) > 2^{-(k+1)}(\Omega_{n+1} - \Omega_n)$. Thus $\mathcal{V}_n$ can go to sleep only finitely often. It follows that for every n, there are only finitely many stages at which $\mathcal{V}_n$ is awake. Let t be the last stage at which any $\mathcal{V}_m$ for $m \leqslant n$ went to sleep. Then $\mu(\mathcal{V}_{n,s} \cap \mathcal{Q}_s) \geqslant 2^{-(k+1)}(\Omega_{n+1} - \Omega_n)$ for every $s \geqslant t$. For otherwise, when the current $\mathcal{V}_j$ goes to sleep, either $\mathcal{V}_n$ or $\mathcal{V}_m$ for $m < n$ would wake, contrary to the choice of t. $\square_{5.7.1}$

We now define a pair of computable functions f and g by simultaneous recursion. We begin by setting $f(-1) = -1$. Given $f(s-1)$, we define $f(s) > f(s-1)$ and $g(s)$ to be sufficiently large so that for every $n < s$,

$$\Omega_{f(s)} - \Omega_n \leqslant 2(\Omega_{g(s)} - \Omega_n),$$

and for every $n < g(s)$,

$$\mu(\mathcal{V}_{n,f(s)} \cap \mathcal{Q}_{f(s)}) \geqslant 2^{-(k+1)}(\Omega_{n+1} - \Omega_n).$$

Note such values always exist: if $g(s)$ is such that $\Omega - \Omega_s \leqslant 2(\Omega_{g(s)} - \Omega_s)$, then the first requirement is satisfied for every $f(s)$; then given any $g(s)$, a sufficiently large choice of $f(s)$ will satisfy the second requirement. Thus we can find such a pair of values by exhaustive search, and f and g are total.

Recall the following notation from Section 5:

$$k_s(n) = \lfloor -\log_2(\Omega_s - \Omega_n) \rfloor.$$

Observe that $k_{f(s)}(n) \geqslant k_{g(s)}(n) - 1$ for all $n < s$, and so

$$\mathbf{c}_{\Omega,R^\complement}(n, f(s)) \leqslant 2 \cdot \mathbf{c}_{\Omega,R^\complement}(n, g(s)).$$

The following claim will complete the proof that A obeys $\mathbf{c}_{\Omega,R}$.

Claim 5.7.2. The total cost $\mathbf{c}_{\Omega,R}\langle A_{f(s+1)} \rangle$ is bounded by 2^{k+3}.

Proof. Let n be least such that $n \in A_{f(s+1)} - A_{f(s)}$. We may assume $n < s$. Then for all $m \geqslant n$, $\pi_2[\mathcal{V}_{m,f(s)}] \subseteq \mathcal{E}_{f(s+1)}$, where π_2 is the projection onto the second coordinate. Let

$$\mathcal{S} = \bigcup_{m \geqslant n} \mathcal{V}_{m,f(s)} \cap \mathcal{Q}_{f(s)}.$$

Then

$$\mu\left(\mathcal{E}_{f(s+1)} - \mathcal{E}_{f(s)}\right) \geqslant \mu(\pi_2[\mathcal{S}]). \tag{5.1}$$

The sets $\mathcal{V}_m$ are disjoint by construction, and

$$\mu(\mathcal{V}_{m,f(s)} \cap \mathcal{Q}_{f(s)}) \geqslant 2^{-(k+1)}(\Omega_{m+1} - \Omega_m)$$

for $m < g(s)$ by choice of $f(s)$, so

$$\mu(\mathcal{S}) \geqslant 2^{-(k+1)}(\Omega_{g(s)} - \Omega_n) > 2^{-(k+1)}2^{-(k_{g(s)}(n)+1)} \geqslant 2^{-(k+2)}\mathbf{c}_\Omega(n, g(s)).$$

On the other hand, $\pi_1[\mathcal{S}] \subseteq \mathcal{U}_{n,f(s)}$, where π_1 is projection onto the first coordinate, and $\mu(\mathcal{U}_{n,f(s)}) \leqslant \mathbf{c}_{\Omega,R^\complement}(n, f(s)) \leqslant 2 \cdot \mathbf{c}_{\Omega,R^\complement}(n, g(s))$. Since $\mathcal{S} \subseteq \pi_1[\mathcal{S}] \times \pi_2[\mathcal{S}]$,

$$\mu(\mathcal{S}) \leqslant \mu(\pi_1[\mathcal{S}]) \cdot \mu(\pi_2[\mathcal{S}]) \leqslant 2 \cdot \mathbf{c}_{\Omega,R^\complement}(n, g(s)) \cdot \mu(\pi_2[\mathcal{S}]),$$

whence, as $\mathbf{c}_\Omega \leqslant \mathbf{c}_{\Omega,R} \cdot \mathbf{c}_{\Omega,R^c}$,

$$2^{-(k+3)}\mathbf{c}_{\Omega,R}(n, g(s)) \leqslant \mu(\pi_2[\mathcal{S}]).$$

Therefore, by (5.1), $\mu(\mathcal{E}_{f(s+1)} - \mathcal{E}_{f(s)}) \geqslant 2^{-k-3} \cdot \mathbf{c}_{\Omega,R}(n, g(s)) \geqslant 2^{-k-3} \cdot \mathbf{c}_{\Omega,R}(n, s)$, and so $\mathbf{c}_{\Omega,R}\langle A_{f(s+1)}\rangle \leqslant 2^{k+3}\mu(\mathcal{E})$. $\square_{5.7.2,\text{ Lem. 5.7 for } A \text{ c.e.}}$

We lift the c.e. case to the general case. Thanks to Theorem 2.1, this is relatively easy, say compared to the approach taken in [11].

Lemma 6.3. *For any infinite computable set R, obedience to $\mathbf{c}_{\Omega,R}$ is downward closed under Turing reducibility.*

Proof. This is similar to [11, Prop. 2.3]. For brevity, let $f(k) = |R \cap k|$. We use the facts that f is non-decreasing, and that for any b, $f(k) \geqslant^+ f(k + b)$.

Let B be a Δ^0_2 set that obeys $\mathbf{c}_{\Omega,R}$. Let $A \leqslant_T B$, say $A = \Psi^B$ for some functional Ψ. Let $\langle B_t\rangle$ be a computable approximation of B witnessing that $B \models \mathbf{c}_{\Omega,R}$. Since $\mathbf{c}_{\Omega,R} \to \mathbf{c}_\Omega$, B is K-trivial. Let φ be the use function for the computation $\Psi^B = A$. By [1, Lem. 2.5], as B is K-trivial and φ is B-computable, $k(n) \geqslant^+ k(\varphi(n))$. Say b is a constant witnessing this. We define an increasing sequence of stages $s(i)$, starting with $s(0) = 0$; $s(i)$ is the least stage $s > s(i-1)$ such that $|\Psi^{B_s}_s| > i$ and for all $n \leqslant i$, $k_{i+1}(n) \geqslant b + k_s(\varphi_s(n))$, where $\varphi_s(n)$ is the use of the computation $\Psi^{B_s}_s(n)$. We then let $A_i = \Psi^{B_{s(i)}}_{s(i)}$. We claim that the approximation $\langle A_i\rangle$ witnesses that A obeys $\mathbf{c}_{\Omega,R}$. The reason is that if $A_i(n) \neq A_{i+1}(n)$ and $n \leqslant i$, then the A-cost paid is $2^{-f(k_{i+1}(n))}$, whereas at some stage $t \in (s(i), s(i+1)]$ we see a change in B below $v = \varphi_{s(i)}(n)$, showing that the total cost paid by B along this interval of stages is at least $2^{-f(k_t(v))} \geqslant 2^{-f(k_{s(i)}(v))}$. This allows us to bound the A-cost, as $k_{i+1}(n) > b + k_{s(i)}(v)$. $\square$

Note that in fact obedience to $\mathbf{c}_{\Omega,R}$ is downward closed under ML-reducibility (by Theorem 5.11), but this uses Lemma 5.7.

Proof of Lemma 5.7 in the general case. Let A be K-trivial and suppose that the hypotheses of the proposition hold. By Theorem 2.1, let $C \geqslant_T A$ be c.e. such that $C \equiv_{\mathrm{ML}} A$. The ML-equivalence implies that $C \leqslant_T Y$; the c.e. case shows that $C \models \mathbf{c}_{\Omega,R}$. By Lemma 6.3, A obeys $\mathbf{c}_{\Omega,R}$ as well. $\square$

7. Fragments of Ω and strong jump-traceability

A cost function $\mathbf{c}$ is *benign* [12] if from a rational $\varepsilon > 0$, we can compute a bound on the length of any sequence $n_1 < s_1 \leqslant n_2 < s_2 \leqslant \cdots \leqslant n_\ell < s_\ell$ such that $\mathbf{c}(n_i, s_i) \geqslant \varepsilon$ for all $i \leqslant \ell$. For example, $\mathbf{c}_\Omega$ is benign, with the bound being $1/\varepsilon$.

A set A is *strongly jump-traceable* [9] if for every order function h (i.e., every computable, non-decreasing, and unbounded function), for every ψ partial computable in A, there is an h-bounded c.e. trace for ψ, that is, a sequence $\langle T(n)\rangle$ such that $|T(n)| \leqslant h(n)$, $T(n)$ is uniformly c.e., and $\psi(n) \in T(n)$ for all $n \in \operatorname{dom}\psi$.

The strongly jump-traceable sets form an ideal in the Turing degrees [5, 7] which is a proper sub-ideal of the K-trivials [8]. For more on strong jump-traceability, see the survey [13]. A characterisation that will concern us here is that a set is strongly jump-traceable if and only if it obeys all benign cost functions [12, 7].

Proposition 7.1. *The strongly jump-traceable sets form an ideal in the ML-degrees.*

Proof. We can characterise strong jump-traceability using computability from random sequences. There is more than one such characterisation. For example, a set is strongly jump-traceable if and only if it is computable from all superlow ML-random sequences [10], also if and only if it is computable from all superhigh random sequences [10, 13]. Alternatively, a c.e. set is strongly jump-traceable if and only if it is computable from a Demuth random sequence by combining [14] and [16]; this extends to all K-trivials by Theorem 2.1. □

In [11], it is observed that every strongly jump-traceable set is a p-base for all $p > 0$. However, this is not a characterisation. The sets that are p-bases for all $p > 0$ are the $1/\omega$-bases, those which are computable from each column from an infinite partition of some random sequence. Equivalently, they are computable from Ω_R for all computable sets R of positive lower density. Some such sets are not strongly jump-traceable. Here we see that we obtain a characterisation of strong jump-traceability if we drop the density condition.

Proposition 7.2. *For any infinite computable set R, $\mathbf{c}_{\Omega,R}$ is benign.*

Proof. Given a rational $\varepsilon > 0$, first, we can effectively find an m with $2^{-|R\cap m|} < \varepsilon$. Let $n_1 < s_1 \leqslant n_2 < s_2 < \cdots \leqslant n_\ell < s_\ell$ be a sequence such that for all $i \leqslant \ell$, $\mathbf{c}_{\Omega,R}(n_i, s_i) > \varepsilon$. This means that $k_{s_i}(n_i) < m$, and so $\Omega_{s_i} - \Omega_{n_i} \geqslant 2^{-m}$. So

$$1 > \Omega > \sum_{i\leqslant\ell} \Omega_{s_i} - \Omega_{n_i} \geqslant \ell \cdot 2^{-m},$$

and thus $\ell < 2^m$. □

Proposition 7.3. *For any benign cost function $\mathbf{c}$, there is an infinite computable set R with $\mathbf{c}_{\Omega,R} \to \mathbf{c}$.*

Proof. Suppose $g(\varepsilon)$ is a computable bound witnessing that $\mathbf{c}$ is benign. We will construct a left-c.e. real $\beta < 1$. By the recursion theorem, we may assume that we already know a constant $\delta > 0$ and a computable approximation to Ω with $\delta(\beta_s - \beta_n) < \Omega_s - \Omega_n$ for all n and s. Choose a computable sequence $m_0 < m_1 < \cdots$ such that

$$\sum_i \frac{2^{-m_i} \cdot g\left(2^{-(i+1)}\right)}{\delta} < 1.$$

Let $R = \{m_0 < m_1 < \cdots\}$.

Define $\beta_0 = 0$. At stage $s+1$, if $\mathbf{c}(n, s+1) \leqslant \mathbf{c}_{\Omega,R}(n,s)$ for all n, then let $\beta_{s+1} = \beta_s$. Otherwise, let n be least with $\mathbf{c}(n, s+1) > \mathbf{c}_{\Omega,R}(n,s)$. Let $i = \lfloor -\log \mathbf{c}(n, s+1) \rfloor$. Define $\beta_{s+1} = \beta_s + 2^{-m_i}/\delta$. The point of this is to increase Ω: in this case, we have

$$\Omega_{s+1} - \Omega_s > \delta \cdot (\beta_{s+1} - \beta_s) = 2^{-m_i},$$

and so $k_{s+1}(s) \leqslant m_i$. In turn, this implies that $\mathbf{c}_{\Omega,R}(s, s+1) \geqslant 2^{-i}$.

Claim 7.3.1. For all n, $\underline{\mathbf{c}}_{\Omega,R}(n) \geqslant \underline{\mathbf{c}}(n)$.

Proof. We show that for all n and s, $\mathbf{c}(n,s) \leqslant \mathbf{c}_{\Omega,R}(n,s)$. Suppose this holds for s; we verify it for $s+1$. Let $\hat{n}$ be the least n such that $\mathbf{c}(n, s+1) > \mathbf{c}_{\Omega,R}(n,s)$; $\hat{n} = \infty$ if there is no such n. For all $n < \hat{n}$,

$$\mathbf{c}(n, s+1) \leqslant \mathbf{c}_{\Omega,R}(n,s) \leqslant \mathbf{c}_{\Omega,R}(n, s+1).$$

If $\hat{n} < \infty$, let $i = \lfloor -\log \mathbf{c}(\hat{n}, s+1) \rfloor$. For all $n \geqslant \hat{n}$,

$$\mathbf{c}(n, s+1) \leqslant \mathbf{c}(\hat{n}, s+1) \leqslant 2^{-i} \leqslant \mathbf{c}_{\Omega,R}(s, s+1) \leqslant \mathbf{c}_{\Omega,R}(n, s+1),$$

as required. $\square_{7.3.1}$

The proof of the proposition will be complete once we show:

Claim 7.3.2. $\beta < 1$.

Proof. Fix i and let $s_0 < s_1 < s_2 < \cdots$ be the stages s with $\beta_{s+1} - \beta_s = 2^{-m_i}/\delta$. By construction, for every $n \leqslant s_j$, $\mathbf{c}_{\Omega,R}(n, s_j+1) \geqslant 2^{-i}$. Also by construction, there is some $n_j \leqslant s_j$ with $2^{-(i+1)} < \mathbf{c}(n_j, s_j+1) \leqslant 2^{-i}$ and $\mathbf{c}_{\Omega,R}(n_j, s_j) < \mathbf{c}(n_j, s_j+1)$. Thus $n_0 < s_0 + 1 \leqslant n_1 < s_1 + 1 \leqslant \cdots$. It follows that there are at most $g\left(2^{-(i+1)}\right)$ such stages. So

$$\beta = \sum_s \beta_{s+1} - \beta_s \leqslant \sum_i g\left(2^{-(i+1)}\right) \frac{2^{-m_i}}{\delta} < 1,$$

by the choice of m_i. $\square_{7.3.2,\ 7.3}$

It follows that the sets which obey $\mathbf{c}_{\Omega,R}$ for all computable R are precisely the strongly jump-traceable sets. Theorem 5.11 implies the following, which extends the result from [10] that a set is strongly jump-traceable if and only if it is computable from all ω-computably approximable random sequences.

Corollary 7.4. *A (K-trivial) set A is strongly jump-traceable if and only if $A \leqslant_{\mathrm{T}} \Omega_R$ for every infinite computable set R.*

Note that K-triviality is for free here, as such a set is a 1/2-base.

Similarly, we see that a K-trivial set A is strongly jump-traceable if and only if for every infinite computable R, $A \leqslant_{\mathrm{ML}} B_R$, where B_R is smart for $\mathbf{c}_{\Omega,R}$. That is, the ML-ideal of strongly jump-traceable sets is the intersection of the infinitely many principal ideals given by the sets B_R. We conjecture that this ideal is not principal.

References

[1] George Barmpalias and Rod G. Downey. Exact pairs for the ideal of the K-trivial sequences in the Turing degrees. *J. Symb. Log.*, 79(3):676–692, 2014.

[2] Laurent Bienvenu, Adam R. Day, Noam Greenberg, Antonín Kučera, Joseph S. Miller, André Nies, and Dan Turetsky. Computing K-trivial sets by incomplete random sets. *Bull. Symb. Log.*, 20(1):80–90, 2014.

[3] Laurent Bienvenu, Noam Greenberg, Antonín Kučera, André Nies, and Dan Turetsky. Coherent randomness tests and computing the K-trivial sets. *J. Eur. Math. Soc. (JEMS)*, 18(4):773–812, 2016.

[4] Laurent Bienvenu, Rupert Hölzl, Joseph S. Miller, and André Nies. Denjoy, Demuth and density. *J. Math. Log.*, 14(1):1450004, 35, 2014.

[5] Peter Cholak, Rodney G. Downey, and Noam Greenberg. Strong jump-traceabilty I: The computably enumerable case. *Adv. Math.*, 217(5):2045–2074, 2008.

[6] Adam R. Day and Joseph S. Miller. Density, forcing, and the covering problem. *Math. Res. Lett.*, 22(3):719–727, 2015.

[7] David Diamondstone, Noam Greenberg, and Daniel D. Turetsky. Inherent enumerability of strong jump-traceability. *Trans. Amer. Math. Soc.*, 367(3):1771–1796, 2015.

[8] Rod Downey and Noam Greenberg. Strong jump-traceability II: K-triviality. *Israel J. Math.*, 191(2):647–665, 2012.

[9] Santiago Figueira, André Nies, and Frank Stephan. Lowness properties and approximations of the jump. *Ann. Pure Appl. Logic*, 152(1-3):51–66, 2008.

[10] Noam Greenberg, Denis R. Hirschfeldt, and André Nies. Characterizing the strongly jump-traceable sets via randomness. *Adv. Math.*, 231(3-4):2252–2293, 2012.

[11] Noam Greenberg, Joseph S. Miller, and André Nies. Computing from projections of random points: a dense hierarchy of subideals of the K-trivial degrees. Submitted.

[12] Noam Greenberg and André Nies. Benign cost functions and lowness properties. *J. Symbolic Logic*, 76(1):289–312, 2011.

[13] Noam Greenberg and Dan Turetsky. Strong jump-traceability. Submitted.

[14] Noam Greenberg and Daniel D. Turetsky. Strong jump-traceability and Demuth randomness. *Proc. Lond. Math. Soc. (3)*, 108(3):738–779, 2014.

[15] Denis R. Hirschfeldt, André Nies, and Frank Stephan. Using random sets as oracles. *J. Lond. Math. Soc. (2)*, 75(3):610–622, 2007.

[16] Antonin Kučera and André Nies. Demuth randomness and computational complexity. *Annals of Pure and Applied Logic*, 162:504–513, 2011.

[17] André Nies. Lowness properties and randomness. *Adv. Math.*, 197(1):274–305, 2005.

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

[19] André Nies. Interactions of computability and randomness. In *Proceedings of the International Congress of Mathematicians*, pages 30–57. World Scientific, 2010.

[20] André Nies. Calculus of cost functions. In S. Barry Cooper and Mariya I Soskova, editors, *The Incomputable: Journeys Beyond the Turing Barrier*, Theory and Applications of Computability, pages 183–216. Springer, 2017.

School of Mathematics, Statistics and Operations Research, Victoria University of Wellington, Wellington, New Zealand

E-mail address: greenberg@msor.vuw.az.nz

Department of Mathematics, University of Wisconsin, Madison, WI 53706, USA

E-mail address: jmiller@math.wisc.edu

Department of Computer Science, University of Auckland, Private Bag 92019, Auckland, New Zealand

E-mail address: andre@cs.auckland.ac.nz

Kurt Gödel Research Center, University of Vienna, Vienna, Austria

E-mail address: turetsd4@univie.ac.at