Superhighness and strong jump traceability

André Nies
The University of Auckland

ICALP 2009

Preliminaries: Randomness notions

How to formalize the intuitive notion that a set $Z \subseteq \mathbb{N}$ is random?

- Polynomial randomness: the tests are polynomial time betting strategies (martingales).
- Schnorr, and computable randomness: the tests are computable betting strategies.
- Martin-Löf-randomness: the tests are uniformly computably enumerable (c.e.) sequences $(G_m)_{m\in\mathbb{N}}$ of open sets in Cantor space $2^{\mathbb{N}}$ such that the uniform measure of G_m is at most 2^{-m}.

Part 1

Lowness properties

We study lowness properties in recursion theory and computational complexity theory.

Lowness properties in recursion theory

A lowness property of a set $A \subseteq \mathbb{N}$ specifies a sense in which A is close to being computable.
Often this means that A is weak when used as an oracle. A says:

"I can't tell you much that you don't know already".

Lowness properties are closed downwards under Turing reducibility $\leq_T$. Here are some examples:

1. each function $f \leq_T A$ is dominated by a computable function;
2. the usual lowness $A' \leq_T \emptyset'$ (for a set X we let X' denote the halting problem relative to X);
3. lowness for Martin-Löf-randomness: if a set $Z \subseteq \mathbb{N}$ is ML-random, then Z is already ML-random relative to A.

Lowness properties in computational complexity theory

Let $\Sigma = \{0, 1\}$. In complexity theory, a lowness property of a language $A \subseteq \Sigma^*$ specifies a sense in which A is nearly in P.

Such a property should be closed downwards under polynomial Turing reducibility $\leq_T^p$ (or at least one of its variants such as $\leq_m^p$).

We now study subclasses of the recursive sets. So, unlike the case of recursion theory, a lowness property may be given by:

- A resource bound on deterministic Turing machines. This is the case for PSPACE.
- A variant of the machine concept. This is the case for BPP.

Weakness as an oracle

We can also define a lowness property of a language $A \subseteq \Sigma^*$ by specifying a sense in which A is weak when used as an oracle.

Examples:

1. each function $f : \mathbb{N} \to \mathbb{N}$, $f \in \mathrm{P}^A$, is dominated by a function in P (here numbers are represented in binary);
2. $\mathrm{NP}^A = \mathrm{NP}$ (this is equivalent to $A \in \mathrm{NP} \cap \mathrm{CoNP}$);
3. if the set $Z \subseteq \mathbb{N}$ is polynomially random, then Z is already polynomially random relative to A.

Part 2

Weakness as an oracle via tracing

A is computationally weak because the functions A computes have few possible values.

Computable traces

Let $h : \mathbb{N} \to \mathbb{N}$ be computable.

Definition 1

A computable trace with bound h is a sequence $(T_n)_{n\in\mathbb{N}}$ of non-empty sets such that

- $|T_n| \leq h(n)$ for each n
- from n, one can compute the finite set T_n.

$(T_n)_{n\in\mathbb{N}}$ is a trace for the function $f\colon \mathbb{N} \to \mathbb{N}$ if $f(n) \in T_n$ for each n.

We say that A is computably traceable if there is a fixed h such that each function $f \leq_T A$ has a computable trace with bound h.

Theorem 2 (Terwijn, Zambella, 2001; Kjos-Hanssen, Nies, Stephan, 2005)

A is computably traceable $\Leftrightarrow$ *A is low for Schnorr randomness.*

Complexity theory: k-traces

Definition 3

Let $k \geq 1$. A k-trace is a sequence $(T_x)_{x \in \Sigma^*}$ of subsets of Σ^* such that

- $|T_x| = k$ for each x
- The function $x \to$ (code for) T_x is in P.

$(T_x)_{n \in \mathbb{N}}$ is a trace for the function $f\colon \Sigma^* \to \Sigma^*$ if $f(x) \in T_x$ for each x.

We say that A is k-traceable if each function $f \in \mathrm{P}^A$ has a k-trace.

Supersparse sets are 2-traceable

Definition 4 (Ambos-Spies 1986)

Let $f : \mathbb{N} \mapsto \mathbb{N}$ be a strictly increasing, time constructible function. A is f-super sparse if

- $A \subseteq \{0^{f(i)} : i \in \mathbb{N}\}$
- Some machine determines $A(0^{f(i-1)})$ in time $O(f(i))$.

Let f be the iteration of the function $n \rightarrow 2^n$. Ambos-Spies constructed an f-supersparse set in EXPTIME $-$ P.

Theorem 5 (Ambos-Spies 1986)

Each f-supersparse set is 2-traceable.

Question 6

Is each k-traceable set low for polynomial randomness [polynomial Schnorr randomness]?

Part 3

Strong jump traceability

We characterize a strong lowness property in recursion theory using randomness.

Strongly jump traceable sets

- A computably enumerable trace with bound h is a uniformly computably enumerable sequence $(T_x)_{x \in \mathbb{N}}$ such that $|T_x| \leq h(x)$ for each x.
- Let $J^A(e)$ be the value at e of a universal A-partial computable function. (For instance, let $J^A(e) \simeq \Phi_e^A(e)$ where Φ_e is the e-th Turing functional.)
- The set A is called strongly jump traceable if for each order function h, there is a c.e. trace $(T_x)_{x \in \mathbb{N}}$ with bound h such that, whenever $J^A(x)$ it is defined, we have

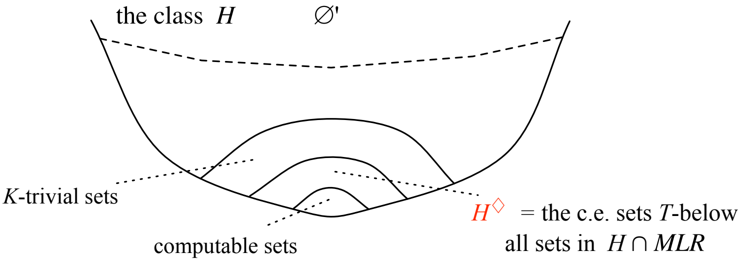
$$J^A(x) \in T_x$$

- *SJT* will denote the class of c.e. strongly jump traceable sets.
- There is an incomputable set in *SJT* by Figueira, Nies, Stephan (2004).

Diamond Classes

$2^{\mathbb{N}}$ denotes Cantor space with the uniform (coin-flip) measure.
For a null class $\mathcal{H} \subseteq 2^{\mathbb{N}}$, we define

$\mathcal{H}^\Diamond =$ the c.e. sets A Turing below each ML-random set in $\mathcal{H}$.



- The larger $\mathcal{H}$ is, the smaller is $\mathcal{H}^\Diamond$.
- $\mathcal{H}^\Diamond$ induces an ideal in the computably enumerable Turing degrees.

A lowness property and its dual highness property

- Recall that $Z \subseteq \mathbb{N}$ is low if $Z' \leq_T \emptyset'$, and Z is high if $\emptyset'' \leq_T Z'$.
- These classes are "too big": we have

  $$(\text{low})^\diamondsuit = (\text{high})^\diamondsuit = \text{computable}.$$

  (For instance, $(\text{high})^\diamondsuit$= computable because there is a minimal pair of high ML-random sets.)
- So we will try somewhat smaller classes, replacing $\leq_T$ by the stronger truth-table reducibility $\leq_{tt}$.

Definition 7 (Mohrherr 1986)

A set Z is superlow if $Z' \leq_{tt} \emptyset'$. Z is superhigh if $\emptyset'' \leq_{tt} Z'$.

A random set can be superlow (low basis theorem). It can also be superhigh but Turing incomplete (Kučera coding).

These diamond classes characterize SJT

The following theorems say that a c.e. set A is strongly jump traceable iff it is computed, in a specific sense, by many ML-random oracles.

Theorem 8 (Greenberg, Hirschfeldt and Nies (to appear))

$$\mathit{SJT} = \mathit{superlow}^{\Diamond}.$$

That is, a c.e. set A is strongly jump traceable $\Leftrightarrow$ *A is Turing below each superlow ML-random set.*

Theorem 9 (Nies, improved version in G'berg, H'feldt, N)

$$\mathit{SJT} = \mathit{superhigh}^{\Diamond}.$$

Diagram: *SJT* means computed by many oracles