complexity and tiny use

Johanna Franklin, Noam Greenberg, Frank Stephan, Guohua Wu

30th September 2009

Background: strong reducibilities and randomness

Turing reducibility is not a measure of relative randomness: it is possible for a non-random set to compute a random set. Some restrictions of Turing reducibility attempt to bridge the gap between $\leqslant_{\mathrm{T}}$ and measures of relative randomness, such as $\leqslant_K$.

One way is to limit the use of the reduction.

Use

Let $A, B \in 2^\omega$, and suppose that $A \leqslant_T B$ by an oracle computation procedure Φ. The use of the computation on input n, denoted $\varphi^B(n)$, is the least upper bound of all the numbers which occur as oracle queries during the computation $\Phi(B, n)$.

Shifting slightly, $B\restriction_{\varphi^B(n)}$ is the shortest initial segment of B which via Φ is mapped to $A\restriction_n$.

Lipschitz reductions [Downey, Hirschfeldt, LaForte]

A Turing reduction $A = \Phi(B)$ is a computable Lipschitz reduction if $\varphi^B(n) \leqslant n + c$ for some constant c. We write $A \leqslant_{\mathrm{cL}} B$.

Fact

If $A \leqslant_{\mathrm{cL}} B$, and A is random, then so is B.

WTT

Computable Lipschitz is a special case of weak truth table reductions. A Turing reduction $\Phi(B) = A$ is a weak truth table reduction if φ^B is bounded by a computable function. We write $A \leqslant_{\text{wtt}} B$.

The associated degree structure, $\mathcal{D}_{\text{wtt}}$, has been studied, but not as extensively as the Turing degrees.

ORDER FUNCTIONS [SCHNORR]

An order function is a non-decreasing, unbounded computable function.

Order functions serve as gauges for computable rates of growth, usually slow ones.

Tiny use

We say that A is reducible to B with tiny use, $A <_{\mathrm{tu}} B$, if for every order function h, there is a reduction $A = \Phi(B)$ such that φ^B is bounded by h.

Note:

1. This is not a reflexive relation. In fact, $A <_{\mathrm{tu}} A$ if and only if A is computable.
2. For some A, it is quite possible that for no B do we have $A \leqslant_{\mathrm{tu}} B$ (not even A'). If $A \leqslant_{\mathrm{tu}} B$, then B is much more compressible than A (beyond all computable compression rates). Hence if A is random, then for no B do we have $A \leqslant_{\mathrm{tu}} B$.
3. The relation $<_{\mathrm{tu}}$ is invariant in $\mathcal{D}_{\mathrm{wtt}}$.

Some motivation for tiny use

Theorem (G,Nies)

If A is strongly jump-traceable, and B is an ω-c.e. random set, then $A \leqslant_{\mathrm{tu}} B$.

Complex sets [Kjos-Hanssen, Merkle, Stephan]

Let C denote plain Kolmogorov complexity.

A set A is complex if there is some order function f such that for all n, $C(A\restriction_{f(n)}) \geq n$.

Fact

A set A is complex if and only if there is some fixed-point-free function $f \leqslant_{\mathrm{wtt}} A$.

Anti-complex sets

Theorem

The following are equivalent for a set A:

1. *For every order function f, for almost all n, $C(A\upharpoonright_{f(n)}) \leqslant n$.*
2. *For all $f \leqslant_{\mathrm{wtt}} A$, $C(f(n)) \leqslant^{+} n$.*

We call these sets anti-complex.

TRACEABILITY [TERWIJN, ZAMBELLA, RAISONNIER]

Let $f\colon \omega \to \omega$. A trace for f is a sequence of finite sets $\langle T_n \rangle$ such that for all n, $f(n) \in T_n$.

- The trace is called computable if the sequence $\langle T_x \rangle$ is computable. The trace is called c.e. if the sequence $\langle T_x \rangle$ is uniformly c.e.

We say that a trace $\langle T_x \rangle$ is bounded by a function f if for all n, $|T_x| \leqslant f(n)$.

Definition

Let h be an order function. A collection $\mathcal{F}$ of functions is *h*-computably traceable if every $f \in \mathcal{F}$ has a computable trace which is bounded by h.

Similarly define, *h*-c.e. traceable.

Fact

If $\mathcal{F}$ is closed under some computable operations, then the following are equivalent:

1. *For some order function h, $\mathcal{F}$ is h-computably traceable.*
2. *For all order functions h, $\mathcal{F}$ is h-computably traceable.*

The same holds for c.e. traceable.

We thus say that $\mathcal{F}$ is computably traceable, analogously, c.e. traceable.

THEOREM (ISHMUKHAMETOV)

Every c.e. traceable Turing degree has a strong minimal cover.

THEOREM (G, DOWNEY, AFTER KUMMER)

Let $A \in 2^\omega$. *If* $\deg_{\mathrm{wtt}}(A)$ *is c.e. traceable, then the effective packing dimension of A is 0.*

LOWNESS IN ALGORITHMIC RANDOMNESS

Let $\mathcal{R}$ be a relativisable notion of randomness. We say that A is low for $\mathcal{R}$ if $\mathcal{R} = \mathcal{R}^A$.

THEOREM (TERWIJN,ZAMBELLA; KJOS-HANSSEN, STEPHAN, NIES)

A Turing degree is low for Schnorr randomness if and only if it is computably traceable.

TRIVIALITY

Sometimes, associated with a notion of randomness is a measure of compression. For example, associated with Martin-Löf randomness is prefix-free Kolmogorov complexity K:

THEOREM (SCHNORR)

A is Martin-Löf random if and only if $K(A\restriction_n) \geq^+ n$.

We can then define a notion of triviality (being far from random):

DEFINITION (SOLOVAY)

A set $A \in 2^\omega$ is Martin-Löf trivial if $K(A\restriction_n) \leqslant^+ K(n)$.

In the case of Martin-Löf randomness, we have a remarkable convergence:

THEOREM (NIES)

A set A is Martin-Löf-trivial if and only if $\deg_T(A)$ *is low for Martin-Löf randomness.*

Schnorr triviality

Schnorr randomness is characterised by an analogue of K – prefix-free complexity, restricted to machines whose domain's measure is computable. Thus we get a notion of Schnorr triviality.

Theorem (Franklin, Stephan)

A set A is Schnorr trivial if and only if $\deg_{tt}(A)$ *is computably traceable.*

Schnorr triviality is not invariant in $\mathcal{D}_{wtt}$.

The coincidence theorem

Theorem

The following are equivalent for a set A.

1. *There is some B such that* $A <_{\mathrm{tu}} B$*.*
2. *A is anti-complex.*
3. $\deg_{\mathrm{wtt}}(A)$ *is c.e. traceable.*
4. $A \leqslant_{\mathrm{wtt}} B$ *for some Schnorr-trivial set B.*

The collection of such sets induces an ideal in $\mathcal{D}_{\mathrm{wtt}}$.

A QUESTION

What is the distribution of anti-complex sets in the Turing degrees?

- If a Turing degree **a** is c.e. traceable, then every set in **a** is anti-complex. This applies to every array computable c.e. Turing degree.

THEOREM

Every high Turing degree contains both anti-complex sets, and sets which are not anti-complex.

A conjecture

Conjecture

There is a c.e. Turing degree which does not contain any anti-complex sets.