Turing Unbound: Transfinite Computation

P.D. Welch

p.welch@bristol.ac.uk
School of Mathematics, Unversity of Bristol, England.

Till thine Infinity shall be
A robe of envenomed agony;
And thine Omnipotence a crown of pain,
To cling like burning gold round thy dissolving brain.

Shelley, Act 1 *Prometheus Unbound*

1 Introduction
2 To the ω'th station!
3 Scope of some other models: the extent of computation in Malament-Hogarth spacetimes. *Case Studies*: a) Etesi-Németi models in Kerr spacetimes;
b) Hogarth's models more abstractly considered.
4 Examples from the Other Side: Infinite Time Turing Machines, Punch Hole Machines.
5 Longer Tapes, Larger objects: Ordinal length Turing machines, Register machines on Ordinals, α-recursion theory; Higher type recursion: Kleene recursion on reals.

1 Introduction

The intention of this talk is to look at various models of transfinite computation and give some calculations as to their comparative power. To clarify the kind of models that we are looking at, they will all be *discrete acting*: this will mean that they essentially perform simple discrete tasks in simple steps or stages. The reader should have in mind the paradigm of the *standard Turing Machine* which of course performs such simple actions as moving one cell left or right on the tape that it is reading, altering a symbol, changing a state *etc etc.* We are thus not considering any kind of machine or notional device that computes in an analogue fashion, nor any machine, such as neural network with nodes primed by infinitely precise real numbers, nor computations performed in chemistry beakers, across cell membranes, or in buckets of slime.

Our purpose here is purely *logico-mathematical*: to determine what these models can and can not do. Just as Turing established the range and capabilities of the Turing machine we wish to do likewise for the various models considered here. We are diagnostic of the formalisms proposed, but agnostic as to the desirability, notional feasability, and so forth, of them. The disclaimer is that mention of any product or products does not imply endorsement by the author.

Various machine proposals that fall under our rubric but which seek to compute functions beyond the standard recursive functions[1] are deliberately constructed to conform to some ambient physical theory or constraint (*cf.* [2], Davies). Apart from Sect.5, we shall not be entering into any discussion of *physical* viabilities, feasabilities and so forth.

In Section 2 we consider briefly some well-known facts about the standard model when the latter is allowed to run out to ω: *i.e.* infinitely many stages are computed and we then are allowed to inspect the tape.

Section 3 looks at a recent group of papers of Hogarth, [9], [10] and Etesi & Németi [2] which consider how to answer questions beyond the recursive using arrangements of Turing machine(s) in a class of general relativistic spacetimes known as *Malament-Hogarth* spacetimes. We delineate there the possible extent of computation in such universes.

Going beyond the ω'th stage requires either stacking up machines, or recycling the tape in some way or other. The Infinite Time Turing Machines (ITTM's) of Hamkins and Kidder [5] do precisely this. One might regard such a model as a laboratory to investigate what can be done transfinitely, just as the standard TM appears to be a suitable laboratory for finite discrete computation (strong forms of the Church-Turing thesis may claim that it *is* a suitable model for such computation). Section 4 considers these. Section 5 catalogues conceptual devices that extend this, or otherwise use "more sets" in their description. One may allow the tape to also be transfinite and one arrives at the ordinal tape machines of Koepke, Dawson, Siders; Koepke and Siders have considered *ordinal register machines*: these have finitely many registers, and a push-down stack, but allow ordinals (rather than integers) as objects to populate the machines. Naturally such devices will compute more relations than an ITTM will. As their authors have shown, one essentially can define Gödel's constructible hierarchy L from these. For completeness we mention here also finite state automata working on trees, or ordinals, (Rabin, Büchi, Neeman, Thomas *inter alia*) but their capabilities are well documented elsewhere, and so we do not pay them much attention in this article. Nor de we consider the extensive field of *computation on the reals*: we consider computation on infinite sequences, as ITTM's and others deal with (and so are capable of *higher type recursion*) but such models consider reals as elements of $\mathcal{P}(\mathbb{N})$ rather than elements of the real continuum $\mathbb{R}$. Older examples of such recursion are Kleene's recursion in $\mathbb{R}$. (See for example [11]).

The unifying thread is a simple statement: transfinite processes encroach on the infinitary world of subsytems of analysis, and of low level set theory, therefore analytical techniques, or those previously studied under the rubric of *generalised recursion theory* (such as α-recursion, higher type recursion) and *descriptive set theory* have a role to play. We want to urge that arguing at a higher conceptual level brings clarity and avoids repeating arguments from earlier eras or other fields.

---

[1] We shall reserve the term *recursive functions* for those standardly computed by TM's leaving "computable function" free for use for those functions computed by the model currently under consideration.

2 To the ω'th station!

A Turing machine if left to its own devices can write a recursively enumerable, or Σ_1, set of integers to its output tape. We may thus consider as computing partial answers to whether $?n \in A?$ for any Σ_1 set A. If we suppose that there is an ω'th stage of time, then we may think of that as also having computed a *full* answer: if at time ω no affirmative answer to $?n \in A?$ has been received we may conclude that $n \notin A$. However we can do better than that. We suppose that our standard Turing machine's are writing an Accept/Reject 0/1 output to the first cell of their tape, without actually halting, but continue computing in case they wish to change their minds

Definition 1. *(Putnam [16])* $R \subset \mathbb{N}$ *is* a trial and error predicate *if there is a Turing machine* M_0 *so that*

$n \in R \Longleftrightarrow$ *the eventual value of* M_0*'s output tape on input* n *is 1*

$n \notin R \Longleftrightarrow$ *the eventual value of* M_0*'s output tape on input* n *is 0.*

The determining feature is that for such predicates the machine eventually settles down to a fixed value after some finite time. The standard interpretation is that for a trial and error predicate one can have a computational arrangement that provides the correct answer without one ever being in the position at a finite stage in time of knowing for sure that we have it. Nor can we have any recursive bound in terms of the input n on the number of time to expect a change of mind. Such predicates are known to be equivalent to Δ_2 predicates in the arithmetical hierarchy (*cf.* also Gold [3]).

One might be concerned that positing an ω'th stage in time might require a mechanism that writes to a particular cell on the tape infinitely often, when calculating such predicates, thereby raising the spectre of some Thomson Lamp like difficulties of what is written on the cell at time ω. However for Δ_2 predicates one can arrange matters so that this does not occur. (Of course this begs the question of where the read/write head will be at time ω!)

To decide Σ_2 predicates in ω steps, this will have to fail. We may *write* a Σ_2 predicate Q, or subset of $\mathbb{N}$, to a recursive slice of the tape, but if we are requiring that the first cell C_0 on the tape has the correct 0/1 answer after ω steps, it is easy to see we may have a positive answer (say given by 1) but for a negative answer in general there can be no finite stage at which "0" will be written to C_0 never to change later (else the predicate would be Δ_2).

So to *decide* Σ_2 questions requires a substantial modification, to which we shall turn after an interlude spent looking at computations in special spacetimes..

3 General Relativistic Models

As mentioned Hogarth, in [9], and [10], and Etesi & Németi [2] consider how to answer questions beyond the recursive using arrangements of Turing machine(s) in a class of general relativistic spacetimes known as *Malament-Hogarth* spacetimes. Both sets of authors query how far in the arithmetic hierarchy such computations can succeed. In [24] we consider the logico-mathematical limits of computation in their models.

Pitowsky [15] gives an account of an attempt to define spacetimes in which the effect of infinitely many tasks can be realised - essentially they allow the result of infinitely many computations by one observer O_r (he used the, as then unsolved, example of Fermat's Last Theorem) performed on their infinite (*i.e.* endless in proper time) world line γ_1, to check whether there exists a triple of integers $x^k + y^k = z^k$ for some $k > 2$ as a counterexample to the Theorem or not. If a counterexample was found a signal would be sent to another observer O_p travelling along a world line γ_2. The difference being that the proper time along γ_2 was finite, and thus O_p could know the truth or falsity of the Theorem in a (for them) finite time, depending on whether a signal was received or not. As Earman and Norton [1] mention, there are problems with this account not least that along γ_2 O_p must undergo unbounded acceleration.

Malament and Hogarth alighted upon a different spacetime example. The following definition comes from [1] (M is a pseudo-Riemannian manifold, g_{ab} a suitable metric):

Definition 2. $\mathcal{M}$=(M, g_{ab}) *is a* Malament-Hogarth (MH) spacetime *just in case there is a time-like half-curve* $\gamma_1 \subset M$ *and a point* $p \in M$ *such that* $\int_{\gamma_1} d\tau = \infty$ *and* $\gamma_1 \subset I^-(p)$.

(Here τ is proper time.[2]) This makes no reference to the word-line of an observer $\mathcal{O}_p$ travelling along their path γ_2, but point out that there will be in any case such a future-directed timelike curve γ_2 from a point $q \in I^-(p)$ to p such that $\int_{\gamma_2(q,p)} d\tau < \infty$, with q chosen to lie in the chronological future of the past endpoint of γ_2. (The important point is that the *whole* of γ_1 lies in the chronological past of $\mathcal{O}_p$. As Hogarth showed in [8] such spacetimes are not globally hyperbolic, thus ruling out many "standard" space-times (such as Minkowski space-time). Hogarth's diagram of a "toy MH space-time" is Figure 1 below. These are in general limited by assumptions they make concerning their physical set ups. Both sets of authors make

Assumption 1 *"no swamping":* no observer or part of the machinery of the system has to send or receive infinitely many signals.

Etesi & Németi then consider a particular spacetime (Kerr spacetime) and the case of a Turing machine sent along the world-line γ_1 searching for counterexamples to a Π_1 predicate (for example). A signal can be sent to $\mathcal{O}_p$ if one is found. Consequently they have a (real world (?)) procedure for deciding Π_1 queries. They observe that if the arrangment sends, for example, *two* signals to $\mathcal{O}_p$, then they can decide the difference of two Π_1 sets, and ask how far this can be taken. We show:

[2] We conform to the notation of Hawking & Ellis [7] and so $I^-(p)$ is the *chronological past* of p: the set of all points q from which a future-directed timelike curve meets p. The spacetimes, all derived from Malament and Hogarth's "toy spacetime", are differentible manifolds with a Lorentz metric g_{ab}, and are time-oriented.

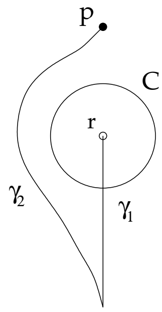

Fig. 1. A toy MH spacetime

Theorem 1. *[24] The relations $R \subseteq \mathbb{N}$ computable in the Etesi-Németi model form a subclass of the Δ_2 predicates of $\mathbb{N}$; this is a proper subclass if and only if there is a fixed finite bound on the number of signals sent to the observer $\mathcal{O}_p$.*

Clearly this is limited by Assumption 1: we know that there can be no recursive bound on the number of times a machine can change its mind (*i.e.* send a signal to $\mathcal{O}_p$) when computing a trial and error predicate, and so *a fortiori* no finite bound. If $\mathcal{O}_p$ is prepared to receive a *potentially* unbounded number of signals then the arrangement decides Δ_2. Note that $\mathcal{O}_p$ will not actually have to *receive* infinitely many signals, to decide a Δ_2 predicate, so Assumption 1 is not broken, but they had better be able in their physical arrangements not to have any boundedness restriction.

Hogarth is more ambitious, he considers preparing spacetime manifolds by arranging singularities as "MH-points" with observers or machines now $\mathcal{O}_{p_j}$, in open regions O_j of M, arranged with chains. By so doing he can decide any arithmetic predicate. He explicitly makes:

Assumption 2 *The regions O_j are all disjoint open regions of M.*

However with this formalism, the construction only scratches the surface of what is possible, one may show:

Theorem 2. *If H is any hyperarithmetic predicate on integers, then there is an MH spacetime in which any query $?n \in H?$ can be computed. Indeed there is a single spacetime which is* HYP*-Deciding, that can compute all such queries.*

However in one sense this is best possible. The last theorem is demonstrated by embedding, not just *chains* of MH spacetime components as in Fig. 1 in regions O_j, but embedding *recursive finite path trees*. Such can code the recursive *construction* of hyperarithemetic sets (a well known fact - see [17] 16.8 or [18]) [3]

[3] Of course this is *not* to say that H is itself recursive, it is just that its *construction* has a recursive description. The set of codes of hyperarithmetic sets is not recursive, or r.e., or even arithmetic, it is complete Π^1_1 ; it thus requires a universal function quantification in analysis, or second order number theory (see for example, [17] Thm.*XX*.

Theorem 3. *Assuming the (modest and standard) requirement that space-time manifolds be paracompact and Hausdorff, for any MH spacetime $\mathcal{M}$ there will be a countable ordinal upper bound, $w(\mathcal{M})$, on the complexity of predicates in the Borel hierarchy resolvable in it.*

The reason is simple: just as in first year analysis, there can be at most countably many disjoint open intervals of $\mathbb{R}$, so there can be only countably many disjoint open regions $O_j \subseteq M$ (paracompactness and Hausdorff implies separable). So there cannot be for every countable ordinal α less than ω_1, regions of M in which there is finite path tree T_α of rank α, of singularities arranged *à la* Hogarth.

Of course you may prefer to believe that *our* spacetime $\mathcal{M}_{\text{real}}$ has w number 0!

4 Beyond ω: Infinite Time Turing Machines (ITTM's)

Let us change the architecture. Suppose we allow cell values to change infinitely often. Let us arbitrarily declare that at stage ω the value of the cell C_i is its eventual value, if such exists, or otherwise, if the value has changed infinitely often let us fill the cell with B(lank) at time ω. Now our extended Turing machine (as we have gone beyond the standard Turing machine) will have a definite answer for us at time ω for answering $?n \in Q?$ for Σ_2 Q: the first cell will have a 1 for "yes" and a B for "no".[4]

Having gone this far we might as well go the whole hog and restart the machine using the current tape contents. We specify that the R/W head has magically reappeared on the first cell C_0. The cell values are as given in the last paragraph, and the machine has entered a special "Limit state" q_L. We thus enlarge the standard state set that the machine was equipped with, $q_1, \ldots, q_N$. Likewise the transition table, or program instruction list, or however we have set up our machines, will have instructions involving q_L which will tell the machine how to proceed to the $\omega + 1$, or $\omega.2 + 1$ or any $\lambda + 1$ stage (for any limit ordinal λ). Note that we still have *finite programs* and so can assume an enumeration $\langle P_e | e \in \mathbb{N} \rangle$ of all programs. At limit stages λ we regard the R/W head as reading C_0. Note also that we have essentially a machine acting on an alphabet of 3 letters $\{0, 1, B\}$. We may let this machine run indefinitely through any or all ordinal segments of time according to taste.

Noting that the machine may halt with the contents of its output tape essentially a member of $3^{\mathbb{N}}$ (identifying 3 with $\{0, 1, B\}$) or perhaps $2^{\mathbb{N}}$, we can think of the machine as outputting reals. Indeed we may think of the machine as having a sequence $y \in 3^{\mathbb{N}}$ on the tape to start with. So we could also think of the machines as performing some kind of higher type recursion computing functionals $F : 3^{\mathbb{N}} \longrightarrow 3^{\mathbb{N}}$, or we may require more conventional output and

[4] One can show that actually one only needs the first cell alone to have the ability to change value infinitely often, the calculation can be done without more than a finite number of chages for the other cells.

compute functions $F : 2^{\mathbb{N}} \longrightarrow 2^{\mathbb{N}}$. Suppose we denote by $P_e(n)$ the e'th computation on integer input of n, represented by an infinite string of n 1's followed by an infinite string of 0's. Several natural questions arise.

Q1 *What is* $\{e | P_e(0) \downarrow\}$*? (The halting problem on integers).*

Q2 *What is* $\{y \in 2^{\mathbb{N}} | \exists e \in \mathbb{N}\ P_e(0) \downarrow y\}$ *?*

Q3 *What are the halting times that arise? That is if* $P_e(0) \downarrow$*halts in* α *steps* $(P_e(0) \downarrow^{\alpha})$*how large is* α*? Is* α *a recursive ordinal?*

Q4 *What is the degree structure that arises, setting*

$$x \leq_\infty y \Longleftrightarrow \exists e \in \mathbb{N}\ P_e(y) \downarrow x?$$

The architecture we have here is essentially that of the *Infinite Time Turing Machines* of Hamkins and Kidder [5]. The difference between the presentation of a 1 tape machine here, and their 3 tape version is rather inessential: they have separate but parallel infinite tapes for input, scratch work, and output. A R/W head may scan the k'th cell from each of the tapes simultaneously, but it acts at limit stages in the same manner by returning to the leftmost triplet of cells. They work instead on an alphabet of $2 = \{0, 1\}$ and specify at limit times λ that $C_n(\lambda)$ has value the $limsup_{\alpha \to \lambda}\{C_n(\alpha)\}$. The answers to **Q1-Q4** above are unaffected by the choice between these two formalisms[5]. Likewise the class of functions $f : \mathbb{N} \longrightarrow \mathbb{N}$, or $F : 2^{\mathbb{N}} \longrightarrow 2^{\mathbb{N}}$ is the same for both types of machine. For most global questions it is immaterial which class of machines is employed.

In order to prove some kind of *Normal Form Theorem* (such as Kleene's T-predicate provides for standard Turing machines) one needs that the class of halting times of computations does not outstrip codes for ordinals that are potentially themselves the lengths of such halting computations. Fortunately this turns out to be the case [22] and we have

Theorem 4. *(Normal Form Theorem [23])* $\forall e \exists e' \forall x \in 2^{\mathbb{N}}$
$P_e(x) \downarrow \longrightarrow [P_{e'}(x) \downarrow y$ *where* $y \in 2^{\mathbb{N}}$ *codes a wellordered course-of-computation sequence for* $P_e(x) \downarrow]$.
Moreover the map $e \longrightarrow e'$ *is effective (in the usual Turing sense).*

Definition 3. *[5]*

$x \subseteq \mathbb{N}$ *is (infinite time) semi-decidable iff* $\exists e \forall n \in \omega[P_e(n) \downarrow 1 \leftrightarrow n \in x]$.

$X \subseteq 2^{\mathbb{N}}$*is (infinite time) semi-decidable iff* $\exists e \forall x \in 2^{\mathbb{N}}[P_e(x) \downarrow 1 \leftrightarrow x \in X]$.

[5] There are some tiny variations in some cases of answers to **Q3** on halting times, globally the supremum γ of halting times on integer inputs are the same, but to some halting times an addition of ω or so units of time is needed to get the sums to come out right. In fact the limit rule that [5] choose is quite robust: changing lim sup to liminf is (perhaps unsurprisingly) immaterial when considering the class of computable functions.

Definition 4. *(i)* $x \leq_\infty y \longleftrightarrow \exists e P_e(y) \downarrow x$; *we have a jump operation:*

$$x^\nabla =_{\mathrm{df}} \{e | P_e(x) \downarrow\}$$

(ii) $X \leq_\infty Y \longleftrightarrow \exists e \forall x [P_e^Y(x) \downarrow 1 \leftrightarrow x \in X \wedge P_e^Y(x) \downarrow 0 \leftrightarrow x \notin X]$; *we have again a jump operation:*

$$X^\blacktriangleright =_{\mathrm{df}} \{(e,x) | P_e(x) \downarrow\} \cup X.$$

In the second clause we have used the notation P_e^Y for the relativised oracle machine which answers queries $?y \in Y?$ for $y \in 2^{\mathbb{N}}$.

Theorem 5. *(Hamkins-Lewis) [5] Any arithmetical predicate is decidable by ITTM's in* $< \omega.\omega = \omega^2$ *steps.*

(ii) Any Π^1_1 *predicate on* $2^{\mathbb{N}}$ *is also ITTM decidable, in particular there is a machine* P_e *that decides whether an input* $y \in 2^{\mathbb{N}}$ *is in WO.*

(iii) The relations "$P_e(x) \downarrow y$*" and "*$P_e(x) \uparrow$*" are* Δ^1_2*. The ITTM semi-decidable predicates form a Spector class which is a proper subclass of the* Δ^1_2 *sets.*

Because of (ii) and (iii) above we see that the reducibility $\leq_\infty$ is, say on the sets of integers, intermediate between hyperarithmeticity and Δ^1_2. Indeed we have an ordinal assignment that satisfies a Spector Criterion:

Theorem 6. *Let* $\lambda^x =_{\mathrm{df}}$ *the least ordinal* λ *which has no code computable from* x*. Then:*

$$x \leq_\infty y \longrightarrow (x^\nabla \leq_\infty y \longleftrightarrow \lambda^x < \lambda^y).$$

4.1 Punch Tape Machines

Suppose you object to the notion of rewriting values to a cell on a tape infinitely often. The you might be happier with the concept of a *punch tape machine*. Such merely have a read/punch head now for writing to a blank cell by punching a hole. So, ignoring difficulties with "hanging chads", such cells are usable once only. Equipped with a standard-ish Turing program, one easily sees that in ω steps again Δ_2, or trial-and-error predicates are decidable. What if now we reset the head to the zero'th cell at time ω and let the machine continue? Could we calculate more? We have the following observation:

Proposition 1. *(S-D. Friedman-PDW) Precisely the arithmetical predicate are decidable by punch tape machines, and any such computation either halts by, or is in an infinite loop, by time* ω^2*.*

5 Larger machines

One may consider ITTM's a starting point for wider classes of machines. Koepke and Dawson independently came up with the idea of allowing ordinal length tape on which $\{0,1\}$ marks could be input or written. To make use of this tape we must not continually reset the head position to 0, to allow the head position at some limit ordinal time λ say, to be the *liminf* of its previous positions. They both allowed the machine at time λ to go into the state that was the again the *liminf* of the previous states. In programming terms this quite neatly puts the machine back at the head of the outermost loop it was cycling through before reaching time λ.

As sets can be coded by ordinals (assuming some form of Axiom of Choice) we have a means of working with sets. Conversely we can think of such machines as *producing* codes for sets by the marks they write. Dawson formulates an *Axiom of Computability* that asserts thet every set can appear coded on the output tape by some program. He then proves that the computable sets form a transitive class satisfying AC and moreover the Generalised Continuum Hypothesis. Clearly such machines have a very absolute nature, and so can be run inside the constructible hierarchy L. It then becomes clear that the class of computable sets in this sense, being a transitive ZF model containing all ordinals, can be none other than L itself which is the smallest such.

Koepke gave a detailed description [12], [13] of the organisation of such programs and a proof that a bounded truth function for L is ordinal computable by a halting program. Whereas Dawson was considering sets that appeared on a tape, Koepke considers *halting* computation from an input tape containing marks for finitely many ordinals.

Theorem 7. *(Koepke [12]) A set x of ordinals is ordinal computable from a finite set of of ordinal parameters if and only if it is an element of the constructible hierarchy L.*

We thus have another presentation of the constructible hierarchy to join those of Gödel, Jensen, and Silver.

5.1 Ordinal Register Machines (ORM)

It appears that the concept of a machine with finitely many registers, but with ordinals (rather than natural numbers) stored in them has arisen from time to time. Platek considered such when he formulated the axioms of KP, and the *metarecursion theory* which was an early example of generalising recursion theory from ω to ω_1^{ck} (see Part B of [18]). Siders considers [14] such a machine with a stack, and the authors show again that the recursive truth predicate for the constructible universe is decidable by such machines.

5.2 Higher Type- and α-recursion

Kleene Recursion and ITTM semi-decidability Kleene developed an equational calculus for developing the notion of *recursion on a higher type*:

$$x \in A \simeq \{e\}(x, y, B, {}^2E) \downarrow 1 \qquad A, B \subseteq \mathbb{R}\,(= 2^{\mathbb{N}})$$

(Here ${}^2E$ is the Type 2 functional evaluating equality between members of ${}^{\mathbb{N}}2$.) It has been characterised ([11]) as a model of computation in which a computational device had a

(i) *countably infinite memory,* and

(ii) *an ability to manipulate (search through, write to) that memory in finite time;* optionally

(iii) *an ability to quiz an oracle (for B) about its entire memory contents.*

We may think of this as a Turing machine with one (or more) infinite tapes on which reals (identified with infinite sequences of 0's,1's) are written and the ability to ask the oracle at any stage of the computation as to whether the current real under consideration is in some "oracle set" $B \subseteq \mathbb{R}$.

• This is not to be conceived as a computation that runs in transfinite segments of discrete time, but rather as one that makes calls for values from *sub-computations*; a computation thus has a *wellfounded finite path tree* structure.

• The course of computation may evolve its own tree structure as it progresses according to its instruction set; we may also view a "machine" as having a previously determined tree structure as part of its "instructions" or program. In short the machine may be viewed as determined by a (finite) program together with a (code, y, for) an infinite finite path-tree.

Kleene degrees: Let $A, B \subseteq \mathbb{R}$; we say that $A \leq_K B$ iff there is some computational arrangement P such as above so that

$$\forall x \in \mathbb{R}(x \frac{\in}{\notin} \Longleftrightarrow P^B(x) \downarrow \frac{1}{0})$$

iff there are Σ_1-formulae in $\mathcal{L}_{\in,\dot{X}}$ φ_1, φ_2, there is $y \in \mathbb{R}$ so that for any $x \in \mathbb{R}\,(x \in A \Longleftrightarrow L_{\omega_1^{B,y,x}}[B, y, x] \models \varphi_1[B, y, x]$
$\Longleftrightarrow L_{\omega_1^{B,y,x}}[B, y, x] \models \neg\varphi_2[B, y, x]\ )$

(here $\omega_1^{B,y,x}$ is the least (B, y, x)-admissible ordinal).

0_K contains $\varnothing, \mathbb{R}$, and in fact consists of the *Borel sets.* $0'_K$ (the K-degree of a complete Kleene semi-recursive set of reals) contains WO the set of reals coding wellorders, and so a complete Π^1_1 set of reals. In fact it consists of the co-analytic, so $\boldsymbol{\Pi^1_1}$ sets.

• (Solovay) [20] AD (Axiom of Determinacy) *implies that the K-degrees are wellordered. Indeed a K-degree forms a boldface pointclass being closed under continuous preimages.*

•(Harrington-Steel) [21], [6] Determinacy(Bool(Π^1_1)) $\Longleftrightarrow \neg\exists A(0 <_K A <_K$ WO)

(Simpson) [11] $V = L \Longrightarrow$ there are many $<_K$-incomparable sets of reals below WO.

If one formulates the Definitions 3 and 4 of decidable and semi-decidable similarly with a real parameter y occurring just as for Kleene degrees, then one obtains similarly a notion of bold-face pointclass. The degrees so obtained are then closed unde continuous pre-images, in short they form a *Wadge degree* and their study is then amenable to methods of descriptive set theory.

Contrasting with the characterisation of Kleene's degrees above we have:

Theorem 8. *[23]* $A \leq_\infty B \Longleftrightarrow_{\mathrm{df}}$ *for some* $e \in \omega$*,some* $y \in \mathbb{R}$:

$$\forall x (x \genfrac{}{}{0pt}{}{\in}{\notin} A \longleftrightarrow P_e^{y,B}(x) \downarrow \genfrac{}{}{0pt}{}{1}{0})$$

$\Longleftrightarrow$ *there are* Σ_1*-formulae in* $\mathcal{L}_{\in,\dot{X}}$ φ_1, φ_2*, and* $y \in \mathbb{R}$*, so that*

$$\forall x \in \mathbb{R}(x \in A \longleftrightarrow L_{\lambda^{B,y,x}}[B,y,x] \models \varphi_1[B,y,x]$$
$$\longleftrightarrow L_{\lambda^{B,y,x}}[B,y,x] \models \neg\varphi_2[B,y,x]).$$

Note the recurrence of the λ symbol: $\lambda^{B,y,x}$ is the supremum of all halting times of programs on input x with oracles for y, and B. The first equivalence here is The last equivalence of course has to be proven, but the reader can take it as a definition.

The $\leq_\infty$-degrees so formed are then boldface pointclasses within the Wadge hierarchy, with 0 and $0^{\blacktriangleleft}$ as the (degrees of the) ∞-recursive, and ∞-semirecursive sets of reals, respectively.

Definition 5. *Let* Γ_0 *be the class of* ∞*-semi-decidable sets of reals.*

Let F be the class of "quickly computable" reals: $x \in F \Longleftrightarrow x \in L_{\lambda^x}$ (Think of the "quickly constructible reals $Q = \{x | x \in L_{\omega_1^x}\}$.) The theorems of Simpson, and Harrington, Steel above become in this setting:

Theorem 9. *[23] (V=L)* $\quad 0 <_\infty F <_\infty 0^{\blacktriangleleft}$.

Theorem 10. *[23] (Det(Bool(Γ_0))) There is no* $A \subseteq \mathbb{R}$ *with* $0 <_\infty A <_\infty 0^{\blacktriangleleft}$.

ITTM's, ORM's and α-recursion Clearly computing on ordinals invites comparison with the theory of α-recursion (see [18] Part C). One could consider computation on tapes of admissble ordinal length, or ORM's with entries restricted to ordinals below some admissible. Given the very absolute nature (in the sense of set theoretic absoluteness) of these machine models, and given the theorems just cited concerning the decidability of the bounded truth predicate for L one sees fairly immediately that proofs of theorems such as the Sacks-Simpson Theorem [19] on a solution to Post's problem, can be carried over, more or less by straight translation, to corresponding proofs concerning incomparable "semi-decidable" sets.

We may also compare the computations on ITTM's with say integer input and α-recursion theoretic results, with $\alpha = \lambda$. Here we are in the easiest case of Sacks-Simpson, as we find, like ω_1^{ck}, that λ is Σ_1-projectible to ω and $L_\lambda \models$"Everything is countable".

Bearing this in mind the following theorem of Hamkins and Lewis (which they proved by using Friedberg-Muchnik arguments) has a soft proof from α-recursion theory: one can simply replace the ordinals coded in the λ-r.e. incomparable sets A^*, B^* by their real codes and obtain ITTM semi-decidable (according to Def. 4) incomparable sets of reals A, B.

Theorem 11. *[4] (Hamkins-Lewis) There are ITTM semi-decidable sets A, B with $0 <_\infty A, B <_\infty 0^{\blacktriangleright}$ which are $<_\infty$ incomparable.*

(If the reader wonders why this appears to be a ZF theorem, which seems to conflict with 9 above, this is because the reducibility [4] uses is that of Def. 4, and the sets A, B are countable - in fact Σ_1-definable over L_λ - as one would expect from an α-recursion theoretic perspective. Hence if one uses the "boldface" reduction of 8 these sets simply disappear into a real parameter y.)

Similar results can be deduced from α-recursion theory for ordinal register or tape machines where the ordinals, or tape lengths come from other admissible α.

References

[1] J. Earman and J.D. Norton. Forever is a day: Supertasks in Pitowsky and Malament-Hogarth spacetimes. *Philosophy of Science*, 60:22–42, 1993.

[2] G. Etesi and I. Németi. Non-Turing computations via Malament-Hogarth spacetimes. *International Journal of Theoretical Physics*, 41(2):341–370, 2002.

[3] E. Gold. Limiting recursion. *Journal of Symbolic Logic*, 30(1):28–48, Mar 1965.

[4] J. D. Hamkins and A. Lewis. Post's problem for supertasks has both positive and negative solutions. *Archive for Mathematical Logic.*

[5] J. D. Hamkins and A. Lewis. Infinite time Turing machines. *Journal of Symbolic Logic*, 65(2):567–604, 2000.

[6] L. Harrington. Analytic determinacy and $0^{\#}$. *JSL*, 43(4):684–693, 1978.

[7] S.W. Hawking and G.F.R.Ellis. *The large scale structure of space-time.* Cambridge University Press, 1973.

[8] M. Hogarth. Does general relativity allow an observer to view an eternity in a finite time? *Foundations of Physics Letters*, 5(2):173–181, 1992.

[9] M. Hogarth. Non-Turing computers and non-Turing computability. *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association Vol. 1*, 1994:126–138, 1994.

[10] M. Hogarth. Deciding arithmetic using *SAD* computers. *British Journal for the Philosophy of Science*, 55:681–691, 2004.

[11] K.Hrbacek and S.Simpson. On Kleene degrees of analytic sets. In H.J.Keisler J.Barwise and K.Kunen, editors, *Proceedings of the Kleene Symposium*, Studies in Logic, pages 347–352. North-Holland, 1980.

[12] P. Koepke. Turing computation on ordinals. *Bulletin of Symbolic Logic*, 11:377–397, 2005.

[13] P. Koepke and M. Koerwien. Ordinal computations. *Mathematical Structures in Computer Science*, 16.5:867–884, October 2006.

[14] P. Koepke and R. Siders. Computing the recursive truth predicate on ordinal register machines. In A. Beckmann *et al.*, editor, *Logical Approaches to Computational Barriers*, Computer Science Report Series, page 21. Swansea, 2006.

[15] I. Pitowsky. The physical Church-Turing thesis and physical computational complexity. *Iyyun*, 39:81–99, 1990.

[16] H. Putnam. Trial and error predicates and the solution to a problem of Mostowski. *Journal of Symbolic Logic*, 30:49–57, 1965.

[17] H. Rogers. *Recursive Function Theory.* Higher Mathematics. McGraw, 1967.

[18] G.E. Sacks. *Higher Recursion Theory.* Perspectives in Mathematical Logic. Springer Verlag, 1990.

[19] G.E. Sacks and S. Simpson. The α-finite injury method. *Annals of Mathematical Logic*, 4:343–367, 1972.

[20] R.M. Solovay. Determinacy and type 2 recursion. *Journal of Symbolic Logic*, 36:374, 1971.

[21] J. R. Steel. Analytic sets and Borel isomorphisms. *Fundamenta Mathematicae*, 108:83–88, 1980.

[22] P. D. Welch. The length of infinite time Turing machine computations. *Bulletin of the London Mathematical Society*, 32:129–136, 2000.

[23] P. D. Welch. Post's and other problems in higher type supertasks. In B. Löwe B. Piwinger T. Räsch, editor, *Classical and New Paradigms of Computation and their Complexity hierarchies, Papers of the Conference Foundations of the Formal Sciences III*, volume 23 of *Trends in logic*, pages 223–237. Kluwer, Oct 2004.

[24] P.D. Welch. Turing Unbound: The extent of computations in Malament-Hogarth spacetimes. *British Journal for the Philosophy of Science*, submitted.