The transfinite action of 1 tape Turing machines

P. D. Welch

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

Abstract. • We produce a classification of the pointclasses of sets of reals produced by infinite time turing machines with 1-tape. The reason for choosing this formalism is that it apparently yields a smoother classification of classes defined by algorithms that halt at limit ordinals.
• We consider some relations of such classes with other similar notions, such as arithmetical quasi-inductive definitions.
• It is noted that the action of ω many steps of such a machine can correspond to the double jump operator (in the usual Turing sense): $\boldsymbol{a} \longrightarrow \boldsymbol{a}''$.
• The ordinals beginning gaps in the "clockable" ordinals are admissible ordinals, and the length of such gaps corresponds to the degree of reflection those ordinals enjoy.

1 Introduction

This paper is concerned with exploring the actions of certain models of transfinite time Turing machines. The idea of formulating such a model is due to Hamkins and Kidder, and [4] is the standard reference here. We refer the reader to this article for basic description of these machines. There is some discussion there on relating these machines to other types of "supertasks" (thus computations involving Rapidly Accelerating Machines, Zeno machines, and computations done in Malament-Hogarth spacetimes). We take the view that the infinite time Turing machines are an idealised laboratory for discussing notions of computation involving the transfinite, much as ordinary Turing machines do for ordinary forms of algorithmic computation. The advantage of these infinite time Turing machines is that they may simulate these others, whilst coming uncluttered with any "Thompson-Lamp" like worries about what state the machine is in after a limit number of steps: we simply define a behaviour for them at limit stages of time.

There have been suggestions of other models: a 1-tape version ([4] had three tapes, one for input, one for scratch work, and one for output.) The one defining feature of such machines is that, of course, if they can take transfinite time, they can read and out put infinite strings of $\{0, 1\}$ bits. Such a string we shall identify with a subset of $\mathbb{N}$ or equivalently with a member of Cantor space ${}^{\omega}2$. Such devices work acording to an ordinary Turing program at "ordinary" successor stages of time, but of course one must specify what these machines do at limit stages of ordinal time. A *limit rule* is needed to specify what a cell on the tape contains at limit time λ if it has altered unboundedly in λ. The class of "computable" functions is surprisingly robust if one alters these limit rules within reason, but the pointclasses of what kinds of real numbers is on the tape at particular times can and does vary.

We can relate these "infinite time Turing Machines" (ITTM's) to some other notions that have appeared: the *arithmetical quasi-inductive definitions* (Burgess)[3]; distilled from the notion of sets of integers defined by *Herzberger revision sequences* ([8],[7]; the *partial fixed point semantics* of Kreutzer [10]. In one sense these definitions are all different facets of the same many sided coin: any one such notion can be replicated or simulated by another, and proofs and techniques formulated with one notion are usually translatable to another.

Pointclasses can be defined by functions delimiting the number of steps that an infinite time Turing machine was allowed to take before converging. We investigate these, so to speak, "TIME" classes a little here. The questions these pointclasses raise are really in turn unanswered questions about the action of such machines. [4] mentions the existence of "gaps" in the order types of halting times of machines on integer inputs. The issue of what those gaps were was not resolved. It was also at that time an open question as to whether a machine on an integer input could require longer time, meaning more ordinal stages, to run, than could be coded by any other nachine's output.

There is a further issue of what are these "machine" processes really? In [13] we looked at the "global" set-theoretical properties of these machines and analysed the relationships between halting times and ordinals produced by such machines, and determined exactly what were the decidable sets of integers *etc.* However the actual detailed analysis of what the machines were producing was passed over. What a machine can produce in ω many steps can be concretely given (see Theorem 4 below). One could thus view a universal such machine as a "double jump" operator, (Corollary 1), which can be iterated through the ordinals, with a specific non-monotone limit operation of "eventual value".

Instead of the 3-tape machines of [4], we shall use instead the 1-tape machine model that we proposed in [15]. We feel that the results here about classifying these classes support the use of this model.

There already has been an analysis of 1-tape machines (in [5]) where it was surprisingly shown that 1-tape machines could not replicate all the features of the standard 3-tape machines for functions $f : {}^{\omega}2 \longrightarrow {}^{\omega}2$ (although they could for $f : {}^{\omega}2 \longrightarrow \mathbb{N}$.)

The difference between the machine of [5] and that of [15] is the use of a third symbol besides $0, 1$ a blank (denoted "B") to be interpreted as "undetermined". We enumerate the cells of the tape by $\langle C_i | i < \omega \rangle$ with C_0 being the leftmost one. We let the contents of the i'th cell at time ν be denoted by $C_i(\nu)$. At successor stages of time, the cells' contents are specified according to the usual Turing machine program finitary rules.

We need to introduce a limit rule to specify cell values at limit times. We declare that a cell of the machine C_i at a limit time μ should have contents $C_i(\mu)$ determined by the contents $C_i(\alpha)$ for $\alpha < \mu$, according to a scheme where $C_i(\mu)$ is a symbol (*i.e.* a 0 or 1 or B) if $C_i(\alpha)$ was 0 (or 1 or B) for all sufficiently large ordinals $\alpha < \mu$; if there has been cofinally in μ a value change, then $C_i(\mu)$ is set to the blank symbol B.

One Tape machine limit rule: If μ is a limit ordinal, then the contents of the i'th cell on the (single) tape at time μ, $C_i(\mu)$, is given by: $(\exists \nu < \mu)(\forall \nu' < \mu)(\nu < \nu' \rightarrow C_i(\nu) = C_i(\nu')) \rightarrow C_i(\mu) = C_i(\nu)$; otherwise set $C_i(\mu)$ to be a blank.

Thus if a cell's value has varied cofinally often below μ, we set the value to the "non-determined" value of a blank. The formalism is otherwise similar to that of [4]: at limit times, by fiat, it is in a special limit state q_L viewing cell C_0. At successor steps of time, the action is just as for an ordinary Turing machine: it acts according to its finite program, reading/writing and moving one cell to the left or right.

If we identify the reals $\mathbb{R}$ with Cantor space 2^ω one then has:

Theorem 1. (cf. *[15] Theorem 1) Let $\mathcal{C}$ be the class of functions $F : \mathbb{R} \rightarrow \mathbb{R}$ computable by the Hamkins-Lewis machines of [4], and $\mathcal{C}'$ those of the one-tape machine just specified. Then $\mathcal{C} = \mathcal{C}'$.*

In general we feel that the 1-tape model has conceptual advantages, not just that it provides a smoother theory of the classes P^f as below. The model

- has a "physical" construction that of a normal Turing machine: namely one infinite tape
- it treats 0's and 1's symmetrically at limits;
- the use of a blank to assign truth values at a limit indicating "undetermined" accords with one's perceptions of a process that going through time vacillates cofinally in that limit ordinal;
- allows a "clean" halting process at limits: algorithms that produce an output only can use the "bit" in the single cell C_0 at the beginning of the tape.

The latter may seem somewhat obscure, but it is the feature of the standard model that there are 3 cells observable that creates an "odd" class of sets of reals computable in exactly certain limit times. It should be emphasised that for the vast majority of results, especially those of a more "global" nature, it makes no difference whatsoever which model one uses. It is only in the mechanics of halting, and the results pertaining thereto that can be affected. (See also a discussion in [5] as to which ordinals are "clockable"[1] on 1-tape machines: there are possibly minor variations here, but the overall picture of the machines, the functions they compute etc, is no different if one takes the [4] or the [15] model.)

We shall define these classes as follows:

Definition 1. *Let* $A \subseteq {}^{\omega}2$.
We say that A *is* semi-decidable *if there is an (infinite time) Turing machine computable functional* φ_e *so that*
(i) $x \in A$ *if and only if* $\varphi_e(x){\downarrow}1$
(ii) A *is* decidable *if both* A *and its complement are semi-decidable.*

By "$\varphi_e(x){\downarrow}1$" we mean that the machine has halted with the first cell of the tape containing a 1; similarly 0 *etc.*

We recall a definition from [4]:

Definition 2. $\lambda^x =_{df} \sup\{\alpha \mid \exists e \varphi_e(x){\downarrow}y \wedge y \in WO \wedge rk(y) = \alpha\}$.

Equivalently (and the reader may take this as a definition):

Fact 11 *([14] Theorem 1.1)* λ^x *is the supremum of halting times of any Turing computable function on input* x.

Prior to the last Fact's proof it was thought possible that halting times might have outrun the ordinals producible by such machines. Without the last Fact one could not have proven:

Theorem 2. (Normal form theorem) $\forall e \exists e' \forall x \in {}^{\omega}2$:
$[\varphi_e(x){\downarrow} \longrightarrow \varphi_{e'}(x){\downarrow}y \in {}^{\omega}2$ *where* y *is a code for a wellordered computation sequence for* $\varphi_e(x)]$.
The map $e \mapsto e'$ *can be made effective (in the usual sense).*

Implicit in Fact 11 - when taken with the definition of decidable sets of reals [4] - (see the discussion in [15]) is the following characterisation of such sets.

[1] They call an ordinal *clockable* if it is the length of a halting computation on 0 input

Fact 12 *$A \in {}^{\omega}2$ is decidable if and only if there are Σ_1 formulae in the language of set theory $\varphi_0(v_0), \varphi_1(v_0)$ so that*

$$x \in A \Longleftrightarrow L_{\lambda^x}[x] \models \varphi_0[x] \Longleftrightarrow L_{\lambda^x}[x] \models \neg\varphi_1[x]$$

We shall be concerned with classifying certain pointclasses of sets of reals that fall strictly within $\boldsymbol{\Delta}^1_2$.

Suppose we are given any function $f : \mathcal{D} \to \omega_1$ of ordinary Turing degrees to countable ordinals that is definable *via* a Σ_1 formula $\psi(v_0, v_1)$, so that for any (ordinary) Turing degree $[y]_T$ we have $f([y]_T) = \alpha$ iff $L[y] \models \psi(y, \alpha)$; then we may define a slice through Δ^1_2 defining a lightface pointclass Γ_0 as follows: $A \in \Gamma_0$ if and only if for some formula $\theta(v_0)$ we have $x \in A \longleftrightarrow L_{f(x)}[x] \models \theta(x)$. (A boldface definition would add in a real parameter here to ψ and θ.) How high a rank f has in ${}^{\mathcal{D}}\aleph_1$ modulo the Martin measure (*cf* [1] p386), determines the complexity of the pointclass.

In [6] we were initially motivated by certain questions of Schindler [12] where certain pointclasses P^f were defined that can be seen to fit into the above general scheme. The pointclasses are strictly within a proper initial segment of Δ^1_2, bounded by the function $f(x) = \lambda^x$ (recalled below.)

We shall define these classes as follows:

Definition 3. *Let $f : \mathcal{D} \longrightarrow \omega_1$ be (standard) Turing invariant. Let $A \subseteq {}^{\omega}2$. We say that $A \in P^f$ if there is a total (infinite time) Turing machine computable functional φ_e so that*
(i) A is decidable by φ_e, that is $x \in A$ if and only if $\varphi_e(x){\downarrow}1$
(ii) $\forall z \in {}^{\omega}2 \quad \varphi_e(z){\downarrow}$ in $\leq f(z)$ steps.

By "Turing invariant" we mean that the value of $f(x)$ is the same irrespective of which choice of x from a degree $\boldsymbol{d} \in \mathcal{D}$ is made. Here, in this phrase, we mean the standard notion of Turing recursion, and degree; hereafter we shall use the notions of infinite time Turing recursion only, and shall always mean these, unless otherwise specified.

If the value of $f(x)$ is some constant α then the classes P^f lie strictly within the Borel hierarchy ([12] Lemma 2.7). Recall that ω_1^x denotes the first ordinal not recursive in x and (see [2]) that ω_1^x is also the first *x-admissible ordinal*, beyond ω, where α is x-admissible, if it is the ordinal height of a transitive model of Kripke-Platek set theory containing x. (It is admissible if it is $\varnothing$-admissible.) Thus the smallest transitive set model of $KP+$ Axiom of Infinity, containing x is $L_{\omega_1^x}[x]$. The ordinals λ^x are x-admissible, and enjoy strong reflecting properties (*cf*[4]).

If we now define $f(x) = \omega_1^x$ then P^f coincides with hyperarithmetic (and so we are really still within the realms of Kleene recursion *e.g.* see [9]). When $f(x) > \omega_1^x$ for all x we then truly enter for the first time the world of sets that are essentially computed by infinite time Turing machines.

Clearly then:

Lemma 1. *If f dominates the function $x \longmapsto \lambda^x$ then P^f equals the class of decidable sets.*

Let f_k be defined as $f_k(x) = \omega_1^x + \omega + k$. As $[\omega_1^x, \omega_1^x + \omega)$ is a gap in the x-clockable ordinals (in either formulation of machine: *cf.* [4] 3.4 and [5] 3.3) thus f_0 is the first one to interest us beyond $g(x) = \omega_1^x$. (In [12] P^{f_0} is denoted P^{++}.)

We classify P^{f_0} as follows. We take Γ to be the pointclass of sets of reals A so that there are formulae $\varphi_0(v_0), \varphi_1(v_0)$, which are

Σ_4 in the language $\mathcal{L}_{\{\dot{\in},\dot{x}\}}$, with the property that

$$\forall x \quad x \in A \Longleftrightarrow L_{\omega_1^x}[x] \models \varphi_0(x) \longleftrightarrow \neg\varphi_1(x).$$

In other words A is in "$\Delta_4(L_{\omega_1^x}[x])$".

By the above comments then P^g is the pointclass of sets of reals that are "$\Delta_1(L_{\omega_1^x}[x])$". Here we have:

Theorem 3. $P^{f_0} = \Gamma$.

The theme from the above analysis is that ω many steps of the ITTM can add two levels of definability (in the arithmetical hierarchy) to the tape's contents, and hence the double Turing jump nature of this operation. As a corollary to the method of proof of $\Gamma \subseteq P^{f_0}$ in the above, we may state this as: that:

Theorem 4. *There is an infinite time program P_e so that on 0 input $(0,0,0,\ldots)$, after ω many steps the tape contains a code for $Fin =_{df} \{n | W_n$ is finite $\}$ (where $W_n = dom\,\varphi_n$ for φ_n a standard Turing computable function.)*

Fin is complete Σ_2 whence follows our remarks on the double jump of the abstract. To be more precise, as the tape works in 3^ω (rather than 2^ω), for the program P_e under discussion, $Fin = \{n | C_{2n}(\omega) = 1\}$. Fin is thus recursive in $\langle C_k(\omega) | k < \omega \rangle$.

This is result is best possible. More formally put:

Theorem 5. *Let $g = \langle C_k(\omega) | k < \omega \rangle$ code the contents of the tape after some program on 0 input has run for ω steps. Then $g \leq_T 0''$.*

Corollary 1. *If, then, g is the tape's contents after running the program P_e of Theorem 4, then $g \equiv_T Fin \equiv_T 0''$.*

It is also not hard to see from the form of Γ above that if $A \in \mathrm{Diff}(<\omega_1^{ck}, \Sigma_1^1)$ the Hausdorff difference hierarchy for levels below the first non-recursive ordinal, then $A \in P^{f_0}$. (Here ω_1^{ck} is the first non-recursive ordinal.)

Theorem 3 generalises:

Theorem 6. *Let $f(x)$ be an x-admissible ordinal which is uniformly not Π_3-reflecting. (That is, we suppose there is a Π_3 formula $\varphi(\dot{x})$ so that for all $x \in {}^{\omega}2$ $L_{f(x)}[x] \models \varphi(\dot{x})$ whilst for all $\alpha < f(x)$ $L_{f(x)}[x] \models \neg\varphi(\dot{x})$.) Then $P^f = \Delta_4(L_{f(x)}[x])$.*

Let $Bool(\Gamma_1)$ be the class of sets of reals that are Boolean combinations of Γ_1 where Γ_1 is the class of similarly defined $\Sigma_4(L_{\omega_1^x}[x])$ sets of reals.

Our methods show:

Theorem 7. $\bigcup_{k<\omega} P^{f_k} = Bool(\Gamma_1)$.

Let Γ_2 be the $\Delta_6(L_{\omega_1^x}[x])$ definable sets.

Theorem 8. $\Gamma_2 = P^{f_\omega}$

The reader can imagine further extensions. It is to be emphasised that these results on the classes P^f hold only for the one tape model described in some more detail below. For the model from [4], one has that the class P^{f_0} would turn out to be the class of sets A that are differences of two sets in $\Sigma_4(L_{\omega_1^x}[x])$, and their complements. It is *inadequate* in the sense of Moschovakis [11] p.158, not being closed under finite unions or intersections. For the comments on the double jump operation, either model will do. In [5] a comparison of the halting times of 2-valued 1-tape machines and the 2-valued 3-tape machines was made. It was left open (Question 3.3 of [5]) whether clockables that were not 1-tape clockable (in the 2-valued sense) were those of compound limit length. The following (using the methods here) provides a counterexample, and so the answer is negative (for either type of 1-tape machine).

Theorem 9. *Let γ be the least ordinal that is Π_3 reflecting. Then $\gamma+\omega$ is 3-tape clockable but not 1-tape clockable. However $\gamma+\omega+1$ is 1-tape clockable.*

In [4] it is proven that no admissible ordinal is clockable: thus every admissible ordinal either starts a gap, or lies within a gap, in the clockable ordinals. We show:

Theorem 10. *If α is an ordinal starting a gap in the clockables, then α is admissible.*

Again, this is for any machine formulation.

References

[1] A.Kanamori. *The Higher Infinite.* Omega Series in Logic. Springer Verlag, New York, 1994.

[2] K.J. Barwise. *Admissible Sets and Structures.* Perspectives in Mathematical Logic. Springer Verlag, 1975.

[3] J.P. Burgess. The truth is never simple. *Journal for Symbolic Logic*, 51(3):663–681, 1986.

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

[5] J. D. Hamkins and D. Seabold. Infinite time Turing machines with only one tape. *Mathematical Logic Quarterly*, 47(2):271–287, 2001.

[6] J. D. Hamkins and P. D. Welch. $P^f \neq NP^f$ almost everywhere. *Mathematical Logic Quarterly*, 49(5):536–540, 2003.

[7] H.G. Herzberger. Naive semantics and the Liar paradox. *Journal of Philosophy*, 79:479–497, 1982.

[8] H.G. Herzberger. Notes on naive semantics. *Journal of Philosophical Logic*, 11:61–102, 1982.

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

[10] S. Kreutzer. Partial fixed point logic on infinite structures. In *Annual Conference of the European Association for Computer Science Logic (CSL)*, volume 2471 of *Lecture Notes in Computer Science.* Springer, 2002.

[11] Y. Moschovakis. *Descriptive Set theory.* Studies in Logic series. North-Holland, Amsterdam, 1980.

[12] R.-D. Schindler. $P \neq NP$ for infinite time Turing machines. *Monatsheft für Mathematik*, 139(4):335–340, Aug 2003.

[13] P.D. Welch. Eventually infinite time Turing degrees: infinite time decidable reals. *Journal for Symbolic Logic*, 65(3):1193–1203, 2000.

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

[15] 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.* Kluwer, Oct 2004.