Bounding lemmata for non-deterministic halting times of transfinite Turing machines

P.D. Welch

School of Mathematics, University of Bristol, BS8 1TW, England

Abstract

We use the methods of descriptive set theory and generalized recursion theory to prove various Bounding Lemmata that contribute to a body of results on halting times of non-deterministic infinite time Turing machine computations. In particular we observe that there is a Uniform Bounding Lemma which states that if any total algorithm halts before the first ordinal admissible in the input x, then there is a recursive ordinal γ by which the algorithm halts on all inputs.

Key words: models of computation, complexity classes, descriptive set theory
1991 MSC: 68Q05, 68Q15, 03D60, 03D65, 03E15

We consider some queries arising from the paper [2]. These concerned various complexity pointclasses defined using halting times of computations on the Infinite Time Turing machines of Hamkins and Kidder [3], with or without existential 'non-determinacy' witnesses. The main theorems of this note are the *Bounding* and *Uniform Bounding* Lemmata below. In particular it is the latter that answers a query from [2] and which explains various phenomena of their paper. Briefly put the Uniform Bounding Lemma states that if any *total* algorithm halts before the first ordinal admissible in the input x, then there is a recursive ordinal γ by which the algorithm halts *irrespective* of x.

We shall first recall the main definitions here of the machine architecture. Later we shall use some results and notions from admissibility theory (for which the reader may consult [1]) and from generalised recursion theory (see [7]) and descriptive set theory(see [4]).

⋆ The author acknowledges support of an EPSRC Springboard Fellowship whilst preparing this paper, and would like to thank the referee for their careful scrutiny and suggestions.

Email address: p.welch@bris.ac.uk (P.D. Welch).

An *infinite time Turing machine* is essentially a standard Turing machine that is allowed to run for transfinite lengths of time. It comes with a standard finite program with an additional *limit state*, q_L, which by fiat it enters at limit stages of time λ; a read/write head returns to the leftmost cell(s) of an infinite tape or tapes also at such times. If we enumerate the cells of the tape(s) as C_i for $i \in \mathbb{N}$, and if C_i has content $C_i(\gamma) \in \{0, 1\}$ at time γ, then, again by fiat, for any limit time λ, for any $i < \omega$ $C_i(\lambda) = \limsup_{\gamma \longrightarrow \lambda} \langle C_i(\gamma) | \gamma < \lambda \rangle$. We have written cell(s) or tape(s) because although the model of [3] had three infinite tapes, for input, scratch work, and output respectively, and an alphabet consisting of just 0's and 1's, other, single tape, models are possible see [10]. We shall maintain however throughout this paper the formalism of [3]. In that case the read/write head is considered as reading simultaneously one cell from each of the three tapes. The state of the machine and its program then determine its next action depending on which triple of 0's and 1's it is reading. At successor stages of time it acts simply like an ordinary Turing machine.

As programs are finite we can consider them as enumerated $\langle P_e | e \in \omega \rangle$ with P_e regarded as computing the e'th ITTM function $\varphi_e : \mathbb{R} \longrightarrow \mathbb{R}$ where we identify $\mathbb{R}$ with $2^{\mathbb{N}}$. P_e thus acts on input strings $x \in 2^{\mathbb{N}}$; integer input is obtained by identifying n with the string consisting of n 1's followed by all 0's. Like any Turing machine either $P_e(x)$ halts, or runs for ever - we write $P_e(x) \downarrow$ or $P_e(x) \uparrow$. We shall be concerned here mostly, but not entirely with halting times of such computations.

Definition 1 *$P_e(x) \downarrow^{\alpha}$ will denote that program $P_e(x) \downarrow$ in exactly α steps. $P_e(x) \downarrow^{\leq \alpha}, P_e(x) \downarrow^{< \alpha}$ are defined analogously.*

To clarify the above: $P_e(x) \downarrow^{\alpha}$ means that at ordinal time α the read/write head is in particular state q_s and is reading a triple of cells (one from each of the three tapes) so that it's program determines that it go into a halting state q_h. Thus a machine may halt exactly at some limit stage of time α where then $q_s = q_L$.

Suppose x is simple: perhaps it is an integer (*i.e.* it is a binary code for $n \in \mathbb{N}$ followed by an infinite string of 0's), perhaps it is 0 (in the above sense) itself. What possible halting times as e varies are there for $P_e(x)$? [3] calls an ordinal *clockable* if it is the halting time of a computation with input 0.

Further, let us define:

Definition 2 *"$P_e(x) \downarrow y$" will denote that $P_e(x) \downarrow$ and that $y \in 2^{\mathbb{N}}$ is the contents of the output tape on halting. (Again $P_e(x) \downarrow^{\alpha} y$ etc. are defined analogously).*

Then we say that y is *writable* if it is the output of some program: $P_e(0) \downarrow y$. An *ordinal β is writable* if some $y \in \mathrm{WO}$ is writable, and y codes a wellordering of rank β. What possible ordinals are writable? It is easy to readjust a program that demonstrates that β is writable to one that shows that any $\beta' < \beta$ is writable. Thus the writable ordinals are an initial segment, λ, of all ordinals. Hamkins and Lewis [3] showed that there are gaps in the clockable ordinals and the following:

Theorem 1 *Hamkins and Lewis [3] If β is admissible then it is not clockable.*

Welch [9] shows that λ, the suprema of the writable ordinals, is also the supremum of the clockable ordinals.

One may generalise these questions to those involving arbitrary input x. The following is Definition 5.18 of Deolalikar, Hamkins & Schindler:

Definition 3 *An ordinal α is* nondeterministically clockable *if there is an algorithm P_e which halts in time at most α for all input and in time exactly α for some input. More generally, α is* nondeterministically clockable before *β if there is an algorithm that halts before β on all input and in time exactly α for some input.*

Symbolically: α is nondeterministically clockable iff

$$\exists e \in \mathbb{N}[\forall x \in 2^{\mathbb{N}} P_e(x)\downarrow^{\leq\alpha} \wedge \exists x \in 2^{\mathbb{N}} P_e(x)\downarrow^{\alpha}].$$

This notion arises in the paper [2], which was concerned with various complexity pointclasses defined using halting times of computations on these machines, with or without existential 'non-determinacy' witnesses. The nomenclature comes from Schindler's paper [8]. The attempt there was made to generalise the concepts of the deterministic polynomial time class P and the non-deterministic class NP from ordinary recursion theory to the infinite time context. As 'non-determinism' in the ordinary recursion theoretic setting can be construed as an algorithm acting on a 'guess' one can use the same idea and define classes via machines that use 'accept/reject' programs to ascertain whether a number or real x is in a class A; non-determinism then here allows some extra side information from a guess to be used.

Definition 4 *(see [2]) (i) $A \in P^{\alpha}$ if there is $\beta < \alpha$ and there is an infinite time Turing machine deciding each $x \in A$ in fewer than β many steps.*

(ii) $A \in NP^{\alpha}$ if there is $\beta < \alpha$ and there is an infinite time Turing machine T such that $x \in A$ if and only if there is $y \in \mathbb{R}$ such that T accepts (x, y), and T halts on any input (w, z) in fewer than β steps.

Here "deciding $x \in A$ in fewer than β many steps" can be taken to mean that the machine *rejects* or *accepts* in less than β many steps. If β is a limit ordinal, then we may equivalently ask that in less than β many steps it halts with a 1 or 0 on the output tape depending on whether x is, or is not, in A. P is defined as P^{ω^ω} and NP as NP^{ω^ω} with the notation to be suggestive of 'polynomial' (although we are of the opinion that this is at most suggestive, and we remain unconvinced that there is any analogy of substance with the classical P/NP notions). We then have:

Theorem 2 *([2] Thm 3.2) The classes NP^α for $\omega+2 \leq \alpha \leq \omega_{1\,\mathrm{ck}}$ are all identical to the class Σ^1_1 of lightface analytic sets. In particular, NP = $NP_{\omega+2}$, and so membership in any NP set can be verified in only ω many steps. Similarly, the corresponding classes co-NP^α are all identical to the Π^1_1 sets. Consequently, NP$\cap$co-NP is exactly the class Δ^1_1 of hyperarithmetic sets.*

Here $\omega_{1\,\mathrm{ck}}$ is the supremum of all recursive ordinals, and $\omega^x_{1\,\mathrm{ck}}$ will be used to denote the supremum of all ordinals recursive in x (in both cases this means recursive in the usual, ordinary sense). We see then in the last theorem how the implicit existential quantifier over reals as guesses in the definition of NP surfaces in its classification. Clearly $P \subseteq NP \cap \mathrm{co}\,NP$, and so the last theorem (due to Schindler) then shows however that "$P \neq NP$".

Theorem 3 *([2] Thms 3.1, 5.4) $P \neq NP \cap \mathrm{co}\,NP$. In fact $P^\alpha \neq NP^\alpha \cap \mathrm{co}\,NP^\alpha$ for $\omega + 2 \leq \alpha < \omega_{1\,\mathrm{ck}}$. However $P^\alpha = NP^\alpha \cap \mathrm{co}\,NP^\alpha$ for $\alpha = \omega_{1\,\mathrm{ck}}$.*

The difference in the two parts of the result above reflects the difference in $\alpha < \omega_{1\,\mathrm{ck}}$ being clockable, and $\omega_{1\,\mathrm{ck}}$ not being so: indeed it starts a gap of clockable ordinals of length ω: no ordinal $\beta \in [\omega_{1\,\mathrm{ck}}, \omega_{1\,\mathrm{ck}} + \omega)$ is clockable; the next clockable is $\omega_{1\,\mathrm{ck}} + \omega$.

We may widen the definition to allow not just constant bounds on the lengths of computations. In the following, we say that $f : \mathbb{R} \longrightarrow \mathrm{On}$ is a *Turing invariant function* if x, y have the same (ordinary) Turing degree, then $f(x) = f(y)$.

Definition 5 *Let $f : \mathbb{R} \longrightarrow \mathrm{On}$ be a Turing invariant function. (i) $A \in P^f$ if there is an infinite time Turing machine deciding each $x \in A$ in fewer than f(x) many steps.*

(ii) $A \in NP^f$ when there is an infinite time Turing machine T such that $x \in A$ if and only if there is $y \in \mathbb{R}$ such that T accepts (x, y), and T halts on any input (x, y) in fewer than f(x) many steps.

Of particular interest is the function $f_0(x) = \omega^x_{1\,\mathrm{ck}} + 1$. They show:

Theorem 4 $P^{f_0} = P^{\omega_{1\,\mathrm{ck}}}$.

They remark at the beginning of this section that P^{f_0} appears at first more generous than the earlier classes, because computations on inputs are now allowed up to ω^x_{1ck} steps.

> "The equality $[P^{f_0}] = P^{\omega_{1\,\mathrm{ck}}}$ should be surprising, because it means that although the computations deciding $x \in A$ for $A \in P^{f_0}$ are allowed to compute up to $\omega^x_{1\,\mathrm{ck}}$, in fact there is an algorithm needing uniformly fewer than $\omega_{1\,\mathrm{ck}}$ many steps. An affirmative answer to the following question would explain this phenomenon completely.
>
> "**Question 4.3** *Suppose an algorithm halts on each input x in fewer than $\omega^x_{1\,\mathrm{ck}}$ steps. Then does it halt uniformly before $\omega_{1\,\mathrm{ck}}$?* "

They also note that $P^{\omega_{1\,\mathrm{ck}}+1} = P^{\omega_{1\,\mathrm{ck}}}$: this is *prima facie* also surprising since stating that $A \in P^{\omega_{1\,\mathrm{ck}}+1}$ requires only that an algorithm determining membership in A must halt before $\omega_{1\,\mathrm{ck}}$, whereas for the latter class an algorithm with *uniform* bound $\beta < \omega_{1\,\mathrm{ck}}$ is required. The *Uniform Bounding Lemma* below answers Question 4.3 in the affirmative, and the *Bounding Lemma* (also below) explains the second phenomenon, as well as having as a direct corollary the following theorem:

Theorem 5 *If β is admissible then it is not nondeterministically clockable.*

Besides the mentioned questions, we can make some further comments and improvements on one or two of their other theorems.

We drop the subscript ck and write ω^x_1 for the first ordinal not recursive in x etc. In the sequel we let WF (WO) denote the set of real numbers coding wellfounded (respectively wellordered) relations. For $y_0 \in \mathrm{WO}$ we let $\|y_0\| \in \mathrm{On}$ denote its ordinal rank.

Theorem 6 *(Uniform Bounding Lemma) Let $F : \mathbb{R} \longrightarrow \mathbb{R}$ be ITTM - computable and total as witnessed by φ_e. If $\forall x \varphi_e(x){\downarrow}^{<\omega^x_1}$ then $\exists\gamma < \omega_{1ck} \forall x \varphi_e(x){\downarrow}^{<\gamma}$.*

Proof Let y be a code of a computation sequence $\varphi_e(x){\downarrow}$ witnessing that it halts. We think of such a code y as coding a sequence of "snapshots" of the tapes' contents etc, along an ordering coded into y; we let $\mathrm{Field}(y)$ be this ordering. Then such a y_x which is moreover wellfounded, exists in $L[x]$. So let y_x be the $L[x]$-least code for such a wellordered sequence. Let $\Phi(y, e, x)$ abbreviate:

"y is the $L[x]$-least code for a wellordered halting computation sequence witnessing $\varphi_e(x){\downarrow}$"

Claim $\Phi(y, e, x) \Longleftrightarrow L_{\omega^x_1}[x] \models \Phi(y, e, x)$.

Proof: Note that as φ_e is total, our assumption that $\varphi_e(x)\downarrow^{<\omega_1^x}$ implies, by running the algorithm P_e inside $L_{\omega_1^x}[x]$, that the latter has the $L[x]$-least witness y_x to $\Phi(y, e, x)$. Also notice that we cannot have for some $y' <_{L[x]} y_x$ that $L_{\omega_1^x}[x] \models \Phi(y', e, x)$. This could only possibly occur if y' coded some illfounded computation that was merely wellfounded in $L_{\omega_1^x}[x]$, and whose wellfounded part was of ordinal length ω_1^x. However as $y_x \in L_{\omega_1^x}[x]$,and $\|y_x\| < \omega_1^x$ the computation coded by y' would have to halt at stage $\|y_x\|$ too. This is absurd.

QED Claim

$\Phi(y, e, x)$ can be expressed as a Σ_1 statement over $L_{\omega_1^x}[x]$. Moreover, again using the totality of φ_e:

$$\neg\Phi(y, e, x) \Longleftrightarrow L_{\omega_1^x}[x] \models \exists z(\Phi(z, e, x) \wedge z \neq y).$$

Hence $\Phi(y, e, x)$ is Δ^1_1. Hence

$$B = \{y_0 | \exists x \exists y(\Phi(y, e, x) \wedge y_0 = \mathrm{Field}(y))\} \in \Sigma^1_1 \cap \mathrm{WO}\,.$$

By (lightface)Σ^1_1-boundedness, (see, *e.g.,* [5] 4A.6) $\{\|y_0\| : y_0 \in B\}$ is bounded in $\omega_{1\,\mathrm{ck}}$.

QED

At the close of Section 4 of a previous version of their paper there was some speculation that one might have in general that $NP^f = \breve{\Gamma}^f$, the dual class of Γ^f. (Note that this is indeed true for the case of $f(x) = \omega_{1ck}$: here $NP^f = \Sigma^1_1 = \breve{\Gamma}^f$.) We gave a counterexample to this which is the Lemma that follows, wherein P^f equals $\Delta(\Gamma^f)$. The point here is to ask the question "where does the existential witness y to x being in some NP^f set live?"

Call $f : \mathbb{R} \to On$ *suitable* if $x \leq_T y \to \omega + 1 \leq f(x) \leq f(y)$. Let f be suitable such that $L_{f(x)}[x] \models$ KPI (KPI is the theory asserting that the universe is an admissible set which is a union of such.) Let:

$$\Gamma^f = \{A \subseteq \mathbb{R} : \exists \Sigma_1 \varphi \forall x[x \in A \longleftrightarrow L_{f(x)}[x] \models \varphi[x]\ ]\}.$$

Lemma 7 *(i)* $NP^f \subseteq \Gamma^f$ *(and hence* $P^f \subseteq NP^f \cap \mathrm{co}\,NP^f \subseteq \Gamma^f \cap \mathrm{co}\,\Gamma^f$*.) Hence in general for such* f, $NP^f \neq \breve{\Gamma}^f$.
(ii) If additionally f *satisfies* $\forall x\, f(x) \leq \Sigma^x$ *then* $P^f = NP^f \cap \mathrm{co}\,NP^f = \Gamma^f \cap \mathrm{co}\,\Gamma^f$.

Proof Assume that $NP^f \subseteq \Gamma^f$ is proven. That $P^f \subseteq NP^f \cap$ co-$NP^f \subseteq \Gamma^f \cap \mathrm{co}\,\Gamma^f$ is straightforward. If f is chosen so that Γ^f is not self-dual, the final sentence then trivially follows. (Examples of such f are easily found: let $f(x)$ be the least μ so that $M = L_\mu[x]$ is an admissible limit of admissibles; then $\rho^1_M = \omega$ and thus M's Σ_1 truth set is not in M, and so provides an example of a set in $\Gamma^f \backslash \breve{\Gamma}^f$.)

For the proof that $NP^f \subseteq \Gamma^f$, let φ_e be such that $\forall x, y \varphi_e(x, y)\downarrow^{<f(x)}$ and $A = \{x : \exists y\ \varphi_e(x, y)\downarrow 1\}$. So where can we find such a witnessing y if x is in A?

Suppose $x \in A$ and y witnesses this. Suppose $\varphi_e(x, y)\downarrow^\gamma 1$ with $\gamma < f(x)$. Let $u_\gamma \in \mathcal{M}_x =_{\rm df} L_{f(x)}[x]$, $u_\gamma \in$ WO, with $\|u_\gamma\| = \gamma$. Let

$$B = \{y : \exists z (z \text{ codes a wellfounded computation sequence witnessing } \varphi_e(x, y)\downarrow^{\|u_\gamma\|} 1)\}$$

$B \neq \varnothing$, and $B \in \Sigma^1_1(x, u_\gamma)$. The Kleene Basis Theorem (relativised to x, u_γ) then states that $\exists y_0 \in B\ y_0 \leq_T \mathcal{O}^{x,u_\gamma}$ (see, *e.g.,* [7] Theorem III.1.3, $\mathcal{O}$ here is Kleene's $\mathcal{O}$ notation.) However then there is such a $y_0 \in \mathcal{M}_x$, as $\mathcal{O}^{x,u_\gamma} \in \mathcal{M}_x$ by our KPI assumption (recall that $\mathcal{O}^{x,u_\gamma}$ is Σ_1-definable over the least admissible set containing x, u_γ). So now

$$\forall x[x \in A \Longleftrightarrow \mathcal{M}_x \models \text{“ } \exists y_0 \varphi_e(x, y_0)\downarrow 1\text{”}]$$

and this yields a defining Σ_1 formula for A, putting A into Γ^f. This concludes the proof of (i). Now assume f is as in (ii). Suppose $A \in \Gamma^f \cap \mathrm{co}\,\Gamma^f$. We show that $A \in P^f$. As $A \in \Gamma^f$ there is a Σ_1 φ so that $\forall x[x \in A \longleftrightarrow L_{f(x)}[x] \models \varphi[x]\ ]$.

Let P_e be the program that searches for a code of some $L_\alpha[x]$ that witnesses that $\varphi[x]$ holds, and halts with output 1, if it finds such. Since $f(x) \leq \Sigma^x$ it can look for codes of such $L_\alpha[x]$ for any $\alpha < \Sigma^x$. Then $\forall x[x \in A \longleftrightarrow P_e(x) \downarrow 1]$. However we also have that $A \in$ co-Γ^f. So there is another program $P_{e'}$ that similarly searches for witnesses that to the fact that $x \notin A$: thus we have $\forall x[x \notin A \longleftrightarrow P_{e'}(x) \downarrow 1]$. So let e'' be the index of a program simulating these two programs together, looking for the first to halt, etc. This will halt before $f(x)$. Hence $A \in P^f$.

QED

We can get another Bounding Lemma:

Theorem 8 *(Bounding Lemma) Suppose β be admissible. Let F be ITTM-computable, and total so that $\forall x \varphi_e(x)\downarrow^{\leq\beta}$ where φ_e computes F. Then $\exists \gamma < \beta$ $\forall x \varphi_e(x)\downarrow^{<\gamma}$.*

Proof Suppose the theorem false as witnessed by the total function $\varphi = \varphi_e$.

Then β is obviously countable. By a theorem of H. Friedman and Jensen any countable admissible β is ω_1^r for some $r \subseteq \omega$ (*cf.*[6]). Let T be the following theory consisting of the following sets of sentences in the language $\mathcal{L}_{\in,\dot{r}}$ augmented by a new constant c:

(i) KP $+\dot{r} \subseteq \omega$; (ii) the $\in$-diagram of $\langle L_\beta[r], \in\rangle$;

(iii) "$\forall x[x \in \dot{y} \longrightarrow \bigvee\!\!\!\bigvee_{z\in y} x = \dot{z}]$" for all $y \in L_\beta[r]$.

(iv) "$\gamma \in c \wedge$c is an ordinal" for all $\gamma < \beta$.

(v) "$\forall a \leq c L_a[\dot{r}] \not\models$ KP"

(vi) "$\exists x \exists f[f$ maps c order preserving into Field(y) where y codes a halted course of computation of the form $\varphi(x)\downarrow$."

Claim If $T_0 \subseteq T$, $T_0 \in L_\beta[r]$, then T_0 has a model.

Proof Let $\delta < \beta$ be the least ordinal not "mentioned" in T_0. Find a (well-founded) KP model $\mathcal{N}$, with $r \in \mathcal{N}$, and with an $x \in \mathcal{N}$, with $\mathrm{On}^{\mathcal{N}} > \delta$, and so that $\mathcal{N} \models \neg\varphi(x)\downarrow^{\leq\delta}$. Then $\exists f \in \mathcal{N}$ with $f : \delta \longrightarrow$ Field(y) where $y \in \mathcal{N}$ codes the course of computation. Let δ interpret c. QED

By the Barwise Compactness Theorem T has a model $\mathcal{M}$. By (i)-(iii) this is a KP model whose $L[r]$-part end-extends $L_\beta[r]$, and moreover $\mathrm{WFP}(\mathcal{M}) \cap \mathrm{On} = \beta$ (by virtue of (v).) Let $x_0 \in \mathcal{M}$ witness (vi). Then we shall have that for every $\delta < \beta, \mathcal{M} \models$" $\neg\varphi(x_0)\downarrow^{<\delta}$ ". However in V we have then that $\varphi(x_0)\downarrow^{\beta}$. Moreover note that β is x_0-admissible (otherwise we could Σ_1-define inside $\mathcal{M}$, β from x_0 and ordinal parameters less than β). However we have just argued that β is x_0-clockable! This contradicts Theorem 1 above. QED

Hence in the terminology of [2] "$P_\beta = P_{\beta+1}$" and "$NP_\beta = NP_{\beta+1}$" so this shows that the requirement on β not being a limit of non-clockables, can be lifted from their [2] Theorem 5.10.

In section 6 of [2] they consider the P^f/NP^f classes restricted to sets of integers. The above arguments show that for many of them $P^f = NP^f$!. We shall use our following unpublished result which is cited in their paper as Lemma 5.8.

Lemma 9 *([11] Lemma 2.5) If α is a clockable ordinal, then every ordinal less than the next admissible ordinal beyond α is writable in time $\alpha + \omega$.*

Lemma 10 *Let $\beta \leq \lambda$ be such that β is an admissible limit of admissibles but is not interior to any gap in the clockables (i.e., it is a limit of clockables).*

Then

$P^\beta \cap \mathcal{P}(\mathbb{N}) = NP^\beta \cap \mathcal{P}(\mathbb{N})$.

Proof: Let $A \in NP^\beta \cap \mathcal{P}(\mathbb{N})$. Let φ_e witness this: $\forall n, y\ \varphi_e(n,y)\downarrow^{<\beta}$ and $\forall n[n \in A \iff \exists y \varphi_e(n,y)\downarrow 1]$. The Bounding Lemma shows that there is a smaller bound $\gamma_0 < \beta$ for the lengths of all these computations. Hence if $n \in A$ then there is a y witnessing this, with $\varphi_e(n,y)\downarrow 1$ and converging in $\leq \gamma_0$. steps. Let $u \in L_\beta \cap \mathrm{WO}$ have rank γ_0. Set:

$$B_n = \{z : \exists y(z \text{ codes a wellfounded computation witnessing } \varphi_e(n,y)\downarrow^{\|u\|} 1)\}$$

Again $\varnothing \neq B_n \in \Sigma^1_1(u)$. As above, appealing to the Kleene Basis theorem again, there are witnessing $z, y_0 \in L_{\gamma_0^+ +1}$ if $n \in A$ (where γ_0^+ is next admissible above γ_0.) In other words to test for membership in A all we have to do is search through potential NP-witnesses y in $L_{\gamma_0^+ +1} \in L_\beta$. But this puts $A \in \Delta_1^{L_\beta}(\{\gamma_0\})$. By our assumption on β, by Lemma 10, γ_0 is itself writable by some program φ_f in time $< \gamma_0^+$. Putting this together $A \in \Delta_1^{L_\beta}$, so $A \in P_\beta$. QED

It would be interesting to have similar results for $\mathcal{P}(\mathbb{R})$ rather than just for $\mathcal{P}(\mathbb{N})$ here.

References

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

[2] V. Deolalikar, J.D. Hamkins, and R-D. Schindler. $P \neq NP \cap co - NP$ for the infinite time Turing machines. *Journal of Logic and Computation*, 15:577–592, October 2005.

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

[4] Y. Moschovakis. *Elementary Induction on Abstract structures*, volume 77 of *Studies in Logic series*. North-Holland, Amsterdam, 1974.

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

[6] G. E. Sacks. Countable admissible ordinals and hyperdegrees. *Advances in Mathematics*, 19:213–262, 1976.

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

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

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

[10] 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.

[11] P.D. Welch. Arithmetical quasi-inductive definitions and the transfinite action of one tape Turing machines. *typescript*, 2004.