EFFECTIVE DOMINATION AND THE BOUNDED JUMP

KENG MENG NG AND HONGYUAN YU

Abstract. We study the relationship between effective domination properties and the bounded jump. We answer two open questions about the bounded jump: (1) We prove that the analogue of Sacks jump inversion fails for the bounded jump and the *wtt*-reducibility. (2) We prove that no c.e. bounded high set can be low by showing that they all have to be Turing complete. We characterize the class of c.e. bounded high sets as being those sets computing the Halting problem via a reduction with use bounded by an ω-c.e. function. We define several notions of a c.e. set being effectively dominant, and show that together with the bounded high sets they form a proper hierarchy.

1. Introduction

The purpose of this paper is to study the effects strong reducibilities have on classical computability concepts. In this paper we look at a well-known classical computability notion - domination. The goal of this paper is to study to what extent strong reducibilities have on domination and related concepts.

There have been many results in the literature connecting the computational strength of an oracle A to the complexity of the class of functions dominated by or dominating A-computable functions. For instance, a hyperimmune-free degree can be characterized as the Turing degree of an oracle A such that each A-computable function is dominated by some computable function. The fact that a hyperimmune-free oracle A is "computably dominated" suggests that A should be easy to guess and that it should be computationally weak as an oracle. This intuition turns out to be only somewhat true, and applies to properties where the computational difficulty can be expressed as total A-computable functions. For instance, the difficulty of computing A' cannot be expressed in terms of the rate of growth of A-computable functions (it can only be coded as *partial* A-computable functions). Indeed, it is still open what exactly the jump of a hyperimmune-free degree is, although some partial results are obtained in [10]. Nevertheless, it is easy to control the double jump of a hyperimmune-free degree, since the double jump has to do with deciding the totality of an A-computable procedure, and one can easily prove that double jump inversion holds for the hyperimmune-free degrees. The hyperimmune-free basis theorem for effectively closed sets also shows that hyperimmune-free degrees are computationally feeble.

Another well-known interaction between the computational power of an oracle and domination comes from the class of *array computable degrees*. This was introduced by Downey, Jockusch and Stob to capture the computational strength of an oracle required to carry out "multiple permitting" arguments (see [7]). A

Key words and phrases. *wtt*-degrees, dominant function, bounded jump, jump inversion, high degrees.

The first author is partially supported by MOE2015-T2-2-055 and RG131/17.

related notion is that of non-totally ω-c.e. (see Downey, Greenberg and Weber [6]). These classes relate the computational power of an oracle A needed to carry out certain constructions with the growth rates of A-computable functions. Recent work of Ambos-Spies and Losert study these in greater detail. This is generalized by Downey and Greenberg and they defined a hierarchy of Turing degrees by measuring the amount of multiple permitting strength present; for this we refer the reader to the monograph [5] or the survey paper [4].

Domination properties are also intimately related to the jump classes. Recall that A (or its Turing degree) is low$_n$ if $A^{(n)} \leq_T \emptyset^{(n)}$, and is high$_n$ if $A^{(n+1)} \geq_T \emptyset^{(n)}$. For a Δ^0_2 set A this means that the degree of the n^{th} jump of A is as low or as high as it possibly can be; and the jump is one of the standard ways of measuring the computational strength of an oracle. For instance, a Δ^0_2 Turing degree $\boldsymbol{a} \leq \mathbf{0}'$ is high if $\boldsymbol{a}' = \mathbf{0}''$, that is, $\boldsymbol{a}'$ is the greatest possible. This property implies that a high degree is computationally complex, and that $\boldsymbol{a}$ carries a lot of information similar to $\mathbf{0}'$. A high c.e. degree can be seen as being computationally complex in that many different constructions can be carried out below it; indeed "high permitting" allows one for instance to construct a minimal pair of c.e. degrees below a given high c.e. degree. The idea of high permitting is that almost every request for a permission to change the set built below the given high degree will be eventually granted. This intuition is formalized by Martin, who showed that a Turing degree $\boldsymbol{a}$ is high iff $\boldsymbol{a}$ is *dominant*, that is, $\boldsymbol{a} \geq deg_T(f)$ for some function f which dominates every total computable function. This means that when constructing a set $B \leq_T f$, if we could formulate a sequence of "permissions" as a total computable function ψ, then f being dominant means that its value $f(x)$ on almost every input x has to increase after the stage where $\psi(x)$ is defined. Martin's result has been generalized by Harris [9] to give a characterization of non-low$_n$ degrees for all $n < \omega$ in terms of domination properties.

Given the number of different results in the literature relating computational strength with domination properties, it is natural to investigate what happens when one considers variations on domination by imposing strict effective conditions. Our goal here is to investigate and define effective versions of being dominant, and to see if there are any notions of highness related to these effective versions of being dominant, in a way similar to Martin's theorem above. The smallest step one could take to effectivize being dominant is to say that $A \geq_T f$ for some function f such that for every e, if φ_e is total then $\varphi_e(x) < f(x)$ for every $x > g(e)$, where g is Δ^0_2. That is, we place an effective bound on the input $x = g(e)$ beyond which domination must happen. Notice that if we allow g to be Δ^0_3 or even $g \leq_T f$, then the resulting notion is simply equivalent to being the usual notion of high.

The notion above was defined and studied in [8], where the authors called it *strongly dominant*. They proved that a Δ^0_2 degree is strongly dominant if and only if the degree is $\mathbf{0}'$, therefore giving rise to nothing new. Here we will consider several other ways to vary the parameters.

Definition 1.1. We call a c.e. set A $(\leq_r, \mathcal{C})$-dominant if $A \geq_r f$ for some function f such that for every e, if φ_e is total then $\varphi_e(x) < f(x)$ for every $x > g(e)$, where $g \in \mathcal{C}$.

We shall be interested in $\mathcal{C} = \Delta^0_2, \omega$-c.e. or Δ^0_1, and only focus on c.e. sets A. We prove that for $\leq_r = \leq_T$, being $(\leq_T, \mathcal{C})$-dominant is equivalent to being Turing complete for any of the three choices for $\mathcal{C}$. This suggests that we should consider $\leq_r$ to be a reducibility stronger than $\leq_T$.

Many natural Turing reductions have the property that the use of the oracle is bounded by a computable function. We will refer to such reductions as *weak truth table reductions*, and write $A \leq_{wtt} B$. This is also commonly known in the literature as bounded Turing (bT) reducibility. A *truth-table reduction* is a weak truth table reduction that is total on all oracles, and we write $A \leq_{tt} B$. Notice that a truth table reduction Γ can be expressed as finite truth tables, one for each $\Gamma(n)$, where the input rows are the digits of the oracle while the outputs are the values of the functional. A weak truth table reduction can be expressed as partial truth tables, where not necessarily every row has an output value.

In this study we will consider $\leq_r = \leq_{wtt}$, which means that we have to look at various *wtt*-degrees within the complete Turing degree.

Recall that one of the goals of this investigation is to relate domination properties with the computational strength of the oracle. Since we are now considering effective versions of being dominant and looking at *wtt*-degrees, one naturally wonders if this has to do with the bounded jump, more specifically, being bounded high. Is there an appropriate analogue of Martin's theorem that allows us to express being bounded high in terms of being $(\leq_{wtt}, \mathcal{C})$-dominant?

We will look at the bounded jump and bounded high *wtt*-degrees and clarify the relationship between bounded high sets and the effective versions of being dominant introduced above. Along the way we also provide the solutions to some open questions in [1, 2]. The bounded jump was introduced by Anderson and Csima [1] in order to define a jump operator for the *wtt*-degrees. They wanted the jump operator to be bounded in the use of the oracle (much like how the *wtt*-reducibility is defined), and all of the properties usually associated with the (Turing) jump operator to hold as well.

Definition 1.2 (Anderson, Csima [1])**.** For any set A, the bounded jump of A, denoted by A^b, is defined as $\left\{x \mid \exists\, i < x \ \left(\varphi_i(x)\downarrow \ \& \ \Phi_x^{A\upharpoonright\varphi_i(x)}(x)\downarrow\right)\right\}$.

They proved that the bounded jump operator is strictly increasing on the *wtt*-degrees and that the sets *wtt*-below the n^{th}-iterate of the bounded jump of $\emptyset$ are exactly the ω^n-c.e. sets. One of the reasons for looking at the bounded jump comes from an earlier result of Downey, Csima and Ng [3], where they proved that the analogue of Shoenfield jump inversion fails to hold for the *wtt*-degrees. Namely, they showed that there is a Σ_2^0 set $C >_{tt} \emptyset'$ such that for every $D \leq_T \emptyset'$, $D' \not\equiv_{wtt} C$. However, this result of Downey, Csima and Ng exploits the fact that the Turing jump is defined with respect to Turing (and not *wtt*-) reducibilities.

Indeed, by considering jump inversion for the *bounded jump* with the *wtt*-reducibility, Anderson and Csima [1] were able to show that the analogue of Shoenfield jump inversion *did* hold: They could show that if C is any set such that $\emptyset^b \leq_{wtt} C \leq_{wtt} \emptyset^{2b}$, there is a set $A \leq_{wtt} \emptyset^b$ such that $A^b \equiv_{wtt} C$. Anderson and Csima [1] asked if the analogue of Sacks jump inversion holds in this setting, namely, if one could make A in their result c.e. In Theorem 2.1 we answer their question by proving that this is impossible:

Theorem 2.1. *There exists an $\omega + 1$-c.e. set A such that for any c.e. set V, we have either $V^b \not\leq_{wtt} A \oplus \emptyset'$ or $A \not\leq_{wtt} V^b$.*

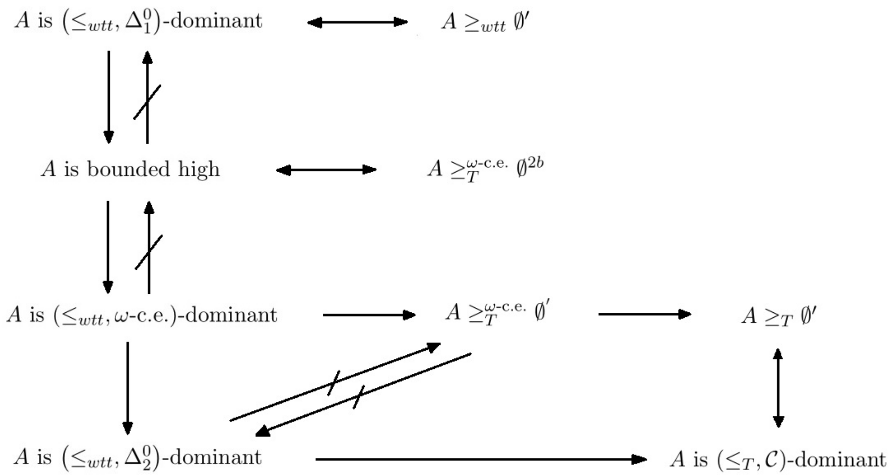

FIGURE 1. The relationships between various properties, where A is a c.e. set. Each implication is strict, while a crossed out arrow indicates a non-implication.

The notion of a bounded high (and bounded low) set can be defined using the bounded jump: A set A is bounded low if $A^b \leq_{wtt} \emptyset^b$ and is bounded high if $A^b \geq_{wtt} \emptyset^{2b}$. Anderson, Csima and Lange [2] showed that the information coded in the bounded jump is quite different from the standard jump; for instance they constructed a c.e. bounded low set that is high (in the standard sense). They also constructed an ω-c.e. low set that is bounded high, and asked if it was possible for a c.e. set to be low and bounded high at the same time.

We answer their question in the negative by showing that every c.e. bounded high set must already be Turing complete. In fact, we were able to obtain a characterization of the c.e. bounded high sets as being those sets which compute $\emptyset^{2b}$ with an ω-c.e. bound on the use:

Theorem 3.2. *The following are equivalent for a c.e. set A.*

(i) *A is bounded high, i.e. $A^b \geq_{wtt} \emptyset^{2b}$.*
(ii) *$A^b \geq_1 \emptyset^{2b}$.*
(iii) *$A \geq_T^{\omega\text{-c.e.}} \emptyset^{2b}$.*

Here we say that $A \geq_T^{\omega\text{-c.e.}} B$ iff there is a Turing functional Φ and an ω-c.e. function φ such that $B = \Phi^A$ where the use is bounded by φ. (Note that this relation is *not* transitive). As a corollary to Theorem 3.2 we obtain [2, Theorem 3.2] that the analogue of the jump theorem fails for the bounded jump.

Given that the bounded jump is a *wtt*-degree notion rather than a Turing degree notion, it is natural to explore the possibility of bounded high as being the notion that would capture the exact computational strength of an oracle which is $(\leq_{wtt}, \mathcal{C})$-dominant. We show that being bounded high and c.e. lies strictly between being $(\leq_{wtt}, \Delta^0_1)$-dominant and being $(\leq_{wtt}, \omega\text{-c.e.})$-dominant. In Section 4 we will also separate the various notions of effective domination. Our results in Section 4 are summarized in Figure 1.

We expect these effective domination properties to be very useful in permitting constructions. All the properties studied here are of c.e. sets and imply being Turing complete, so we are essentially looking at sets with the complete Turing degree, but possibly incomplete *wtt*-degree.

For standard notations and basic computability notions, we refer the reader to [11].

1.1. **Organization of the paper.** The paper is organized as follows. In Section 2 we prove that the analogue of Sacks' jump inversion fails for the bounded jump and the *wtt*-reducibility. In Section 3 we give a characterization of the c.e. bounded high sets. Finally in Section 4 we explore the effective domination notions in greater detail. We first show that being $(\leq_T, \mathcal{C})$-dominant is the same as being simply Turing complete for $\mathcal{C} = \Delta^0_2, \omega$-c.e. or Δ^0_1. We next establish all the non-trivial implications and non-implications in Figure 1.

2. Sacks' Jump Inversion for bounded Turing degrees with the bounded jump

We show that the analogue of Sacks' jump inversion fails with respect to $\leq_{wtt}$ and the bounded jump.

Theorem 2.1. *There exists an $\omega+1$-c.e. set A such that for any c.e. set V, we have either $V^b \not\leq_{wtt} A \oplus \emptyset'$ or $A \not\leq_{wtt} V^b$.*

2.1. **Requirements and notations.** Let $(\Delta_e, \Gamma_e, \delta_e, \gamma_e, V_e)$ be an effective listing of all possible 5-tuples where Δ_e and Γ_e are Turing functionals, δ_e and γ_e are partial computable functions, V_e is a c.e. set. We assume that the use of Δ_e is bounded by δ_e and the use of Γ_e is bounded by γ_e. We want to construct a $\omega+1$-c.e. set A and satisfy the following requirement for each e:

$$R_e:\ V^b_e \neq \Delta_e^{A\oplus\emptyset'\upharpoonright\delta_e} \text{ or } A \neq \Gamma_e^{V^b_e\upharpoonright\gamma_e}$$

Then $A \oplus \emptyset'$ will be the desired set. Note that the requirements automatically ensure that $A \oplus \emptyset' >_{wtt} \emptyset'$: If $A \oplus \emptyset' \equiv_{wtt} \emptyset'$ then when $V_e = \emptyset$ we are not able to satisfy R_e.

Let $M_n = \max\limits_{0\leq i\leq k<n} \varphi_i(k)$. Obviously $n \mapsto M_n$ is an ω-c.e function with the obvious approximation $M_n[s]$; for each n, $M_n[s]$ increases at most n^2 times. For each n, we define a sequence of functions $\{\psi^n_k\}_{k<n^2}$ by the following. We set $\psi^n_k(z) =$ the k^{th} value in the approximation of M_n for every z. Let $i^n_0 < i^n_1 < \cdots < i^n_{n^2-1}$ be indices for the sequence, i.e. $\varphi_{i^n_k} = \psi^n_k$, for all $k < n^2$. These functions are either constant or empty functions. In the sequel we will usually denote $m = n^2$.

We now make a few definitions which are used in the proof.

Definition 2.2. $\sigma_{e,i}$ is said to be currently *free* if $V^b(\mathtt{w}_{e,i}) = 0$. We write $\sigma \preccurlyeq \sigma'$ if for all w, if $\sigma(w) = 1$, then $\sigma'(w) = 1$. We write $\sigma \prec \sigma'$ if $\sigma \preccurlyeq \sigma'$ and $\sigma \neq \sigma'$.

2.2. **Strategy for a single R_e.** For the purpose of discussion, we drop the subscript e. The following are the steps for R_e.

(1) Pick a fresh witness $z > r_e$, where $r_e := \max\{z_{e-1}, l_e\}$, and $l_e := \max\{\delta_{e'}(\mathtt{w}_{e',i}) \mid i < 2^{\gamma_{e'}(z_{e'})},\ e' < e\}$. (The parameter $\mathtt{w}_{e',i}$ will be defined later).

(2) Wait for $\gamma(z)\downarrow$. Then we consider the indices $i_0^{\gamma(z)} < i_1^{\gamma(z)} < \cdots < i_m^{\gamma(z)}$.

There are $2^{\gamma(z)}$ many possible versions of $V^b \upharpoonright \gamma(z)$. We assign a binary string σ of length $\gamma(z)$ to each version of $V^b \upharpoonright \gamma(z)$. We order these $2^{\gamma(z)}$ strings lexicographically; we denote the i^{th} string by $\sigma_{e,i}[s]$. When the context is clear we simply write σ_i, $i < 2^{\gamma(z)}$.

We pick fresh subwitnesses $\mathtt{w}_{e,\sigma_{e,0}}, \ldots, \mathtt{w}_{e,\sigma_{e,2^{\gamma(z)}-1}}$; in particular they are all larger than $i_m^{\gamma(z)}$ and all subwitnesses for all other requirements. (A witness z is used directly for diagonalization, the associated list of subwitnesses $\mathtt{w}_{e,\sigma_{e,i}}$ are used to force changes in V). Each subwitness $\mathtt{w}_{e,\sigma_{e,i}}$ is associated with the string $\sigma_{e,i}$.

We keep $V^b(\mathtt{w}_\sigma) = 0$ when the current $V^b(w) = 0$ for some w with $\sigma(w) = 1$.

(3) We wait for $\delta\left(\mathtt{w}_{e,\sigma_{e,i}}\right)\downarrow$ for all $i < 2^{\gamma(z)}$, then we intialize all lower piority requirements.

(4a) Case a: If $V^b\left(\mathtt{w}_{\sigma_c}\right) = 0$, where $\sigma_c := V^b[s] \upharpoonright \gamma(z)$ the current value of V^b up to $\gamma(z)$, we wait for the following to be true for all $i < 2^{\gamma(z)}$:

$$V^b\left(\mathtt{w}_{\sigma_{e,i}}\right) = \Delta\left((A \oplus \emptyset') \upharpoonright \delta\left(\mathtt{w}_{\sigma_{e,i}}\right); \mathtt{w}_{\sigma_{e,i}}\right) \text{ and } A(z) = \Gamma\left(\sigma_c; z\right).$$

This is called a *recovery for R_e*. If this happens, we are able to make $V^b(\mathtt{w}_{\sigma_c}) = 1$ by enumerating the computation $\Phi_{\mathtt{w}_{\sigma_c}}\left(V \upharpoonright M_{\gamma(z)}; \mathtt{w}_{\sigma_c}\right)\downarrow= 0$. Observe that $\mathtt{w}_{\sigma_c} > i_m^{\gamma(z)}$ and so the use $M_{\gamma(z)}$ is bounded by $\varphi_i\left(\mathtt{w}_{\sigma_c}\right)$ for some $i < \mathtt{w}_{\sigma_c}$.

(4b) Case b: If $V^b\left(\mathtt{w}_{\sigma_c}\right) = 1$, we wait for a recovery for R_e. Then we toggle z in A, i.e. set $A_{s+1}(z) = 1 - A_s(z)$.

(5a) Case a: Suppose the strategy was last at Step (4a). Let τ be the value of σ_c when we last acted in Step (4a). Wait for a recovery for R_e.

(5a1) Suppose that $V^b(\mathtt{w}_\tau) = 1 = \Delta\left((A \oplus \emptyset') \upharpoonright \delta\left(\mathtt{w}_\tau\right); \mathtt{w}_\tau\right)$ i.e. the value of $\Delta\left((A \oplus \emptyset') \upharpoonright \delta\left(\mathtt{w}_\tau\right); \mathtt{w}_\tau\right)$ has changed from 0 to 1. Note that this can only happen under a $(A \oplus \emptyset') \upharpoonright \delta\left(\mathtt{w}_\tau\right)$- change; in fact, if we were careful about restraining A between Steps (4a) and (5a), we must have a $\emptyset' \upharpoonright \delta\left(\mathtt{w}_\tau\right)$-change.

In this case we look at the new current value of $\sigma_c = V^b[s] \upharpoonright \gamma(z)$. If σ_c is free, we go to (4a). Otherwise, go to (4b). Notice that σ_c may or may not be equal to τ.

(5a2) Suppose that $V^b(\mathtt{w}_\tau) = 0 = \Delta\left((A \oplus \emptyset') \upharpoonright \delta\left(\mathtt{w}_\tau\right); \mathtt{w}_\tau\right)$, i.e. V must have changed to now cause $V^b(\mathtt{w}_\tau) = 0$. Again look at the current value of $\sigma_c = V^b[s] \upharpoonright \gamma(z)$. If σ_c is free, we go to (4a). Otherwise, go to (4b). Again note that σ_c may or may not be equal to τ, but an important point is that if $\tau \not\preccurlyeq \sigma_c$ then we can conclude that τ is now free. (See Lemma 2.4).

(5b) Case b: Suppose the strategy was last at Step (4b) where we toggled z. Let τ be the value of σ_c when we last acted in Step (4b). Wait for a recovery for R_e.

Now look at the current value of $\sigma_c = V^b[s] \upharpoonright \gamma(z)$. If σ_c is free, we go to (4a). Otherwise, go to (4b).

2.3. **Construction.** We apply the Recursion Theorem to obtain an effective list of indices of Turing functionals, and we will pick the subwitnesses $\mathtt{w}_\sigma$ for the different requirements in the construction from this list. Note that the indices i_k^n do not require the Recursion Theorem.

R_e is said to require attention at stage s if one of the following situations happens:

(S1) $s > e$, R_{e-1} has already picked its witness z_{e-1} but R_e hasn't.
Action: Act as in (1) of the basic strategy.

(S2) The last time R_e required attention in (S1) and $\gamma_{e,s}(z_e)\downarrow$.
Action: Act as in (2) of the basic strategy.

(S3) The last time R_e required attention in (S2), $\delta_{e,s}(\mathtt{w}_{e,\sigma_{e,i}})\downarrow$ for all $i < 2^{\gamma_{e,s}(z_e)}$ and R_e has recovered.
Action: If the current σ_c is free, go to (S4), otherwise go to (S5).

(S4) The last time R_e required attention in (S4) or (S5), R_e has recovered, and $V^b(\mathtt{w}_{\sigma_c}) = 0$.
Action: Act as in (4a) of the basic strategy.

(S5) The last time R_e required attention in (S4) or (S5), R_e has recovered, and $V^b(\mathtt{w}_{\sigma_c}) = 1$.
Action: Toggle z by setting $A_s(z) = 1 - A_{s-1}(z)$.

At stage s, act for the highest priority requirement requiring attention, and initialize all lower priority requirements.

2.4. **Verification.** We now verify that the construction works. We let $\sigma_c^e[s]$ be the current value of $V_e^b \upharpoonright \gamma_e(z_e)[s]$. Where the context is clear we simply write σ_c.

Lemma 2.3. *If $s < s'$ and $\sigma_c^e[s] \prec \sigma_c^e[s']$ and $\sigma_c^e[t] \neq \sigma_c^e[s']$ for all $s \leq t < s'$, then $\sigma_c^e[s']$ is free at the beginning of stage s'.*

Proof. Let $\tau = \sigma_c^e[s']$. Since $\sigma_c^e[s] \prec \tau$, there exists $w_0 < \gamma_e(z_e)$ such that $\tau(w_0) = 1$ and $\Phi_{w_0}^{V\upharpoonright M}(w_0)[s]\uparrow$. We claim that τ is free at stage s: Otherwise $V^b(\mathtt{w}_\tau)[s] = 1$, and must have received this definition under (4a) of the basic strategy at some stage $t < s$ such that $\sigma_c^e[t] = \tau$. At that stage we must have $\Phi_{w_0}^{V\upharpoonright M}(w_0)[t]\downarrow$. Note that at stage t we set $V^b(\mathtt{w}_\tau)[s] = 1$ with use $V \upharpoonright M[t]$. However, between t and s, V must change below $M[t]$, which contradicts $V^b(\mathtt{w}_\tau)[s] = 1$. Therefore, τ must be free at s.

Since $\sigma_c^e[t] \neq \tau$ for all $s \leq t < s'$, this means that τ is free at stage s'. □

Lemma 2.4. *If $s < s'$ and $\sigma_c^e[s] \not\prec \sigma_c^e[s']$, then $\sigma_c^e[s]$ is free at the beginning of stage s'.*

Proof. Since $\sigma_c^e[s] \not\prec \sigma_c^e[s']$, there exists w_0, such that $\sigma_c^e[s](w_0) = 1$ and $\Phi_{w_0}^{V\upharpoonright M}(w_0)[s']\uparrow$. We can argue that $\sigma_c^e[s]$ is free at the beginning of s' in the same way as in Lemma 2.3. □

Lemma 2.5. *The number of non-free $\sigma_{e,i}$ between two consecutive togglings of z_e will decrease by 1 if there is no $\emptyset'$-change below* $\max\left\{\delta_e\left(\mathtt{w}_{e,\sigma_{e,i}}\right) \mid i < 2^{\gamma_e(z_e)}\right\}$.

Proof. Note that z_e is only toggled under action (S5). Hence the only possible transitions between recovery stages are either of the form (S5) $\to$ (S4) $\to \cdots \to$ (S4) $\to$ (S5), or (S5) $\to$ (S5). We keep track of the number of non-free $\sigma_{e,i}$ at the beginning of each recovery stage. We go through each possible transition:

S(4) $\to$ S(4):: Suppose the strategy for R_e acts under S(4) at stage s and then under S(4) at the next recovery stage $s' > s$. Note that $\sigma_c^e[s]$ is free at the beginning of stage s and becomes non-free after the action at

s. However, between s and s' we do not change A below $\delta\left(\mathtt{w}_{\sigma_c^e[s]}\right)$. By assumption there is also no $\emptyset'$ change, and so the only way for recovery to take place at s' is for $\sigma_c^e[s]$ to become free again at the beginning of s'. Thus the total number of non-free $\sigma_{e,i}$ remains the same (or decreases). Note that we are not claiming that $\sigma_c^e[s] = \sigma_c^e[s']$, which might be false.

S(4) $\to$ S(5):: This case is the same as the previous. We still have that the total number of non-free $\sigma_{e,i}$ remains the same (or less).

S(5) $\to$ S(4):: Action (S5) does not turn any free σ into a non-free one, so the number of non-free $\sigma_{e,i}$ remains the same (or less).

S(5) $\to$ S(5):: Same as above.

Now let $s < s'$ be two consecutive stages where we toggle z_e. The above shows that the number of non-free $\sigma_{e,i}$ at the beginning of stage s is no less than the number at the beginning of s'. We need to show that it is in fact strictly less. Note that $\Gamma^{\sigma_c^e[s]}(z_e) \neq \Gamma^{\sigma_c^e[s']}(z_e)$ and so $\sigma_c^e[s] \neq \sigma_c^e[s']$.

We first assume that $\sigma_c^e[s] \prec \sigma_c^e[s']$. We know that $\sigma_c^e[s']$ is not free at the beginning of s'. Suppose that $\sigma_c^e[s']$ is free at the beginning of t for some $s \leq t < s'$. This is impossible because only (S4) can cause $\sigma_c^e[s']$ to become non-free, but then at the next recovery stage it has to become free again. Therefore, $\sigma_c^e[s']$ is non-free at the beginning of every stage t, $s \leq t \leq s'$. But this means that there cannot be a recovery stage t such that $s \leq t < s'$ and $\sigma_c^e[t] = \sigma_c^e[s']$, otherwise we will toggle z_e in between s and s'. By Lemma 2.3 $\sigma_c^e[s']$ is free at the beginning of s', a contradiction.

This means that $\sigma_c^e[s] \not\prec \sigma_c^e[s']$. Since $\sigma_c^e[s]$ is not free at the beginning of s, by Lemma 2.4, the number of free strings must decrease. $\square$

Lemma 2.6. *Each R_e can be initialized only finitely often and the number of times z_e is toggled is at most $2^{\gamma_e(z_e)} \cdot m_e$, where $m_e = \max\left\{\delta_e\left(\mathtt{w}_{e,\sigma_{e,i}}\right) \mid i < 2^{\gamma_e(z_e)}\right\}$.*

Proof. For $e = 0$, R_0 can never be initialized. Now suppose for all $e' < e$, $R_{e'}$ can be initialized only finitely often, and the number of changes for $z_{e'}$ is at most $2^{\gamma_{e'}(z_{e'})} \cdot m_{e'}$. Note that R_e is initialized only when for some $e' < e$, $R_{e'}$ acts. By inductive hypothesis, we can conclude R_e can be initialized by $R_{e'}$ only finitely often for each $e' < e$: This is because if some $R_{e'}$ requires attention infinitely often then it must eventually be repeatedly attended to under (S4), but as $V_{e'}^b$ and $\Delta_{e'}^{A\oplus\emptyset'}$ are Δ_2^0 sets, there cannot be any more recovery stages after they are stable on the parameters assigned to $R_{e'}$.

We now count the number of times z_e can be toggled. We will toggle z_e at a stage s only under (S5) and when the current $\sigma_c^e[s]$ is not free. By Lemma 2.5 we can only toggle z_e at most $2^{\gamma_e(z_e)}$ many times before every $\sigma_{e,i}$ becomes free, after which the only way we can toggle z_e again is for there to be a $\emptyset' \upharpoonright m_e$-change. Therefore, the total number of times z_e can be toggled is bounded by $2^{\gamma_e(z_e)} \cdot m_e$. $\square$

Lemma 2.7. *A is $\omega + 1$ c.e.*

Proof. Each number z is picked to be a follower z_e of at most one requirement, and toggling of z only starts when all parameters and uses of R_e is found. Thereafter, by Lemma 2.6, z is toggled at most $2^{\gamma_e(z_e)} \cdot m_e$ times. Since this latter bound is partial computable, this means that A is $\omega + 1$-c.e. $\square$

Lemma 2.8. *All requirements R_e are satisfied.*

Proof. By Lemma 2.6, R_e is initialized only finitely often. Now suppose R_e is not initialized any more after stage $s > e$. If the last time R_e requires attention is under (S1), then $\gamma_e(z_e)\uparrow$ and R_e is satisfied. If this is (S2) then $\delta_e(\mathtt{w}_{e,\sigma_{e,i}})\uparrow$ for some i. If the last time R_e requires attention is in (S4) or (S5), then we never recover and $V^b \neq \Delta^{A\oplus\emptyset'}$ or $A \neq \Gamma^{V^b}$. □

3. Characterizing the c.e. bounded high sets

3.1. **A useful lemma.** Before we begin this section, we start by showing a very useful fact which we will later use; we show that $\emptyset^{2b}$ is effectively 1-complete amongst the ω^2-c.e. sets. We say that $\langle e,k\rangle$ is a *ω^2-c.e. index* if $\varphi_e(x,s)$ and $\varphi_k(x,s)$ are total computable functions and provide an enumeration of a ω^2-c.e. set X. Recall this means that for every n, $X(n) = \lim_s \varphi_e(n,s)$ and for every s, if $\varphi_e(n,s) \neq \varphi_e(n,s+1)$ then $\varphi_k(n,s) >_{\mathcal{O}} \varphi_k(n,s+1)$ where $rng\,(\varphi_k)$ is a set of strong ordinal notations for ordinals less than ω^2.

This is essentially [1, Lemma 5.5], for completeness, we present a similar proof below.

Proposition 3.1 (Anderson, Csima)**.** *There is a total computable function p such that for every $\langle e,k\rangle$, if $\langle e,k\rangle$ is a ω^2-c.e. index for the set X, then $\varphi_{p(\langle e,k\rangle)}$ is total and witnesses $X \leq_1 \emptyset^{2b}$.*

Proof. We first define a number of auxiliary functions to help us construct p. Let

$$N(e,n,s) = \begin{cases} \#\{t < s : \varphi_e(n,t+1) < \varphi_e(n,t)\}, & \text{if } \varphi_e(n,t)\downarrow \text{ for all } t \leq s, \\ \uparrow, & \text{otherwise.} \end{cases}$$

Hence $N(e,n,s)$ is partial computable. We also define the c.e. set

$$W_{j(e,n)} = rng\,(\lambda s\ N(e,n,s)).$$

Obviously j is given by the s-m-n Theorem. $W_{j(e,n)}$ is an initial segment of ω and records the number of changes in $\varphi_e(n,-)$. Now define the (sequence of) partial computable (either constant or nowhere defined) functions

$$\varphi_{i(e,k,n,c)} = \lambda x\ \langle j(e,n), C+2\rangle.$$

Here $C = C(e,k,n,c)$ is the value obtained by the following procedure: Search for the least s such that $\varphi_k(n,t)\downarrow$ for all $t \leq s$ and $\varphi_k(n,s) + \omega\cdot c \leq \varphi_k(n,0)$, where the last expression is interpreted as ordinals. The idea is that s is the least such that $\varphi_k(n,s)$ has decreased by at least c many limit ordinals compared to the start $\varphi_k(n,0)$. Let $C = \varphi_{i(e,k,n,c-1)}(0) + C'$ where $C' \in \omega$ and η is a limit ordinal such that $\varphi_k(n,s) = \eta + C'$. Intuitively, C is meant to capture the maximum number of times $\varphi_k(n,-)$ can change before it drops by more than c many limit ordinals. Again, the function i is given by the s-m-n Theorem.

Here we give an example of how C, C' and s are defined. Suppose for some e,k,n, the function $\varphi_k(n,s)$ behaves as follows. $\varphi_k(n,0) = \omega\cdot 5+10, \varphi_k(n,10) = \omega\cdot 5+1, \varphi_k(n,20) = \omega\cdot 5, \varphi_k(n,30) = \omega\cdot 4+26, \varphi_k(n,40) = \omega\cdot 4+15, \varphi_k(n,50) = \omega\cdot 4+2, \varphi_k(n,60) = \omega\cdot 4, \varphi_k(n,70) = \omega\cdot 3+1$. The value of $\varphi_k(n,t)$ retains the previous value otherwise. Then for $c = 0$, the value of s is 0 and C' is 10. Therefore, $C(e,k,n,0)$ is any upperbound for 10. We chose to let $C(e,k,n,c) = \varphi_{i(e,k,n,c-1)}(0) + C'$ rather than just $C(e,k,n,c) = C(e,k,n,c-1) + C'$ for purely technical reasons. For $c = 1$, the value of s is 30 and the value of C' is 26, so $C(e,k,n,1) \geq 10+26$. Finally, $C(e,k,n,2) \geq 10+26+1$.

Finally, define the sequence of Turing functionals

$$\Phi_{r(e,n,d)}$$

by the following. For each minimal string σ where $\sigma(\langle j(e,n),0\rangle) = \sigma(\langle j(e,n),1\rangle) = \cdots = \sigma(\langle j(e,n),z-1\rangle) = 1$ and $\sigma(\langle j(e,n),z\rangle) = 0$ for some z, we search for the least s such that $N(e,n,s) = z-1$. Define $\Phi^{\sigma}_{r(e,n,d)}(x)\downarrow$ for all x if $\varphi_e(n,s) = 1$; the output is unimportant. Clearly Φ is a Turing functional, so we let r be a 1-1 computable function given by the s-m-n Theorem.

Finally define $\varphi_{p(\langle e,k\rangle)}(n) = r(e,n,d)$ where $\varphi_k(n,0) <_{\mathcal{O}} \omega\cdot d$.

Now we verify that p has the desired properties. Fix e,k such that $\langle e,k\rangle$ is an ω^2-c.e. index for X. Then $\varphi_{p(\langle e,k\rangle)}(n)$ is total because $\varphi_k(n,0)\downarrow$ for all n. Fix n. Now we want to argue that $X(n) = 1$ iff $r(e,n,d) \in \emptyset^{2b}$. This is equivalent to saying that $\Phi^K_{r(e,n,d)}(r(e,n,d))\downarrow$ with use bounded by $\varphi_i(r(e,n,d))$ for some $i < r(e,n,d)$.

First of all suppose that $X(n) = 1$. Let $z-1$ be the number of times in total that $\varphi_e(n,-)$ changes. Therefore $W_{j(e,n)} = [0,z-1]$. Now if we take $\sigma = K \upharpoonright \langle j(e,n),z\rangle + 1$, the construction ensures that we define $\Phi^{\sigma}_{r(e,n,d)}(r(e,n,d))\downarrow$. Since $z-1$ is the total number of changes in $\varphi_e(n,-)$, it follows that $\langle j(e,n),(z-1)+2\rangle < \max\{\varphi_{i(e,k,n,0)},\varphi_{i(e,k,n,1)},\cdots,\varphi_{i(e,k,n,d)}\}$. (We do not specify the input to these functions since they are either constant or everywhere undefined). By choosing d large enough, $\langle j(e,n),z+1\rangle < \varphi_i(r(e,n,d))$ for some $i < d < r(e,n,d)$. The use of Φ is $|\sigma| = \langle j(e,n),z\rangle + 1 < \langle j(e,n),z+1\rangle < \varphi_i(r(e,n,d))$, as required. Hence, $r(e,n,d) \in \emptyset^{2b}$.

Now suppose that $X(n) = 0$. Let $z-1$ be the number of times in total that $\varphi_e(n,-)$ changes, and again we have $W_{j(e,n)} = [0,z-1]$. This means that $K(\langle j(e,n),0\rangle) = K(\langle j(e,n),1\rangle) = \cdots = K(\langle j(e,n),z-1\rangle) = 1$ and $K(\langle j(e,n),z\rangle) = 0$. If $\Phi^K_{r(e,n,d)}(r(e,n,d))\downarrow$ then there is some minimal $\sigma \subset K$ that corresponds to the construction. This means that $\sigma = K\upharpoonright\langle j(e,n),z\rangle+1$, but this is a contradiction because $\varphi_e(n,s) = 0$ for the least s where the value is stable. So, $\Phi^K_{r(e,n,d)}(r(e,n,d))\uparrow$. □

3.2. **The characterization.** We characterize the c.e. bounded high sets as those which compute $\emptyset^{2b}$ with an ω-c.e. bound on the use.

Theorem 3.2. *The following are equivalent for a c.e. set A.*

(i) *A is bounded high, i.e. $A^b \geq_{wtt} \emptyset^{2b}$.*
(ii) *$A^b \geq_1 \emptyset^{2b}$.*
(iii) *$A \geq_T^{\omega\text{-}c.e.} \emptyset^{2b}$.*

As a corollary, each c.e. bounded high set is Turing complete, and therefore cannot be low. This answers a question in [2]. We first prove the easier direction of Theorem 3.2.

Lemma 3.3. *Let A be any set such that $A \geq_T^{\omega\text{-}c.e.} \emptyset^{2b}$. Then $A^b \geq_1 \emptyset^{2b}$.*

Proof. Fix a Turing functional Δ and an ω-c.e. function δ such that $\emptyset^{2b} = \Delta^A$ with use bounded by δ. Suppose that $\delta[s]$ is a computable approximation of δ with the number of mind changes bounded by the computable function h. As in Section 2.1, we define the computable sequence of indices $\{i^n_k\}$ such that each $\varphi_{i^n_k}$ is either a total constant function, or the empty function, and such that $\varphi_{i^n_k}(z) =$ the $k+1^{th}$ value in the approximation of $\delta(n)[s]$ for every z.

Define the computable function f such that $f(n)$ is the index of a Turing functional such that

$$\Phi^X_{f(n)}(z) = \begin{cases} \Delta^X(n), & \text{if } \Delta^X(n)\downarrow = 1, \\ \uparrow, & \text{otherwise.} \end{cases}$$

We pick $f(n) > i^n_k$ for every $k < h(n)$.

We claim that for each n, $n \in \emptyset^{2b}$ iff $f(n) \in A^b$. First suppose that $\emptyset^{2b}(n) = 1$. Then $\Delta^A(n) = 1$. This means that $\Phi^A_{f(n)}(f(n))\downarrow$, with use bounded by $\delta(n)$. But $\delta(n) = \varphi_{i^n_k}(f(n))$ for some $k < h(n)$, which means that $i^n_k < f(n)$. Therefore $f(n) \in A^b$. Now suppose that $\emptyset^{2b}(n) = 0$. This means that $\Delta^A(n) = 0$ which means that $\Phi^A_{f(n)}(f(n))\uparrow$. So $f(n) \notin A^b$. □

3.3. **Proof of (i) ⇒ (iii).**

3.3.1. *Notations and conventions.* The rest of Section 3.3 will be devoted to the proof of (i) ⇒ (iii). Fix a c.e. set A such that A is bounded high. The construction will define an ω^2-c.e. set F, and by the Recursion Theorem we assume that we are given the ω^2-c.e. index for F in advance. Thus by Proposition 3.1 we can fix a Turing functional Ψ and a computable function ψ such that $F = \Psi^{A^b}$ with use bounded by ψ. Our goal is to define a Turing functional Δ and an ω-c.e. function δ such that $\emptyset^{2b} = \Delta^A$ with use bounded by δ. Since A is a c.e. set, we are going to define Δ implicitly by specifying a computable approximation $\delta[s]$ of δ. Whenever A changes below the use $\delta(n)[s]$ we have the option to increase $\delta(n)[s]$. Whether or not we choose to increase $\delta(n)$ we will enumerate a new axiom for $\Delta^A(n)$ using the new value of $A \upharpoonright \delta(n)$, and obviously output the current value of $\emptyset^{2b}(n)[s]$. At the end we shall verify that we only increase δ computably bounded many times.

We shall adopt the convention that $\delta(x)[s]$ and $\psi(x)[s]$ are increasing in the variable x. As before we let $M_n = \max_{0\le i\le k<n} \varphi_i(k)$. Obviously $n \mapsto M_n$ is an ω-c.e function with the obvious approximation $M_n[s]$; for each n, $M_n[s]$ increases at most n^2 times. Our goal is to keep $\delta(n)[s] \geq M_{\psi(n)}[s]$ for every n and s. Of course $M_{\psi(n)}$ can increase at any time, while we can only increase $\delta(n)$ if $A \upharpoonright \delta(n)$ changes. Therefore, whenever $A \upharpoonright \delta(n)$ changes, we check to see if $\delta(n)[s] \geq M_{\psi(n)}[s]$ is still true; if so, we retain the current value of $\delta(n)[s]$ (and enumerate a new axiom for $\Delta^A(n)$). If no, we increase the value of $\delta(n)$ to make it larger than $M_{\psi(n)}$ (and enumerate a new axiom for $\Delta^A(n)$). This will be the only reason we increase $\delta(n)$, and therefore we will only increase δ computably bounded many times.

3.3.2. *The construction of F and δ.* By speeding up the enumerations of A and A^b, we shall assume that whenever we toggle a number n in F, we immediately get a recovery of $F(n) = \Psi(A^b; n)$. This recovery can be one of two types; either we get $A^b[s] \upharpoonright \psi(n) \prec A^b[s+1] \upharpoonright \psi(n)$ (see Definition 2.2), or not. The former case we call a *bad recovery* and the latter case is a *good recovery.* Note that in the case of a good recovery we get an A-change below $M_{\psi(n)}[s]$.

At stage $s = 0$ we set $F = \emptyset$ and $\delta(n) = M_{\psi(n)}$ for all n.

At stage $s > 0$ if $M_{\psi(0)}$ increases we initialize Δ by clearing all axioms enumerated in Δ and resetting $\delta(n) = M_{\psi(n)}$ for all n.

Next we pick the least $n < s$ *requiring attention*: We say that n requires attention if either $\emptyset^{2b}(n) \neq \Delta^A(n)$ or $\delta(n+1) < M_{\psi(n+1)}$ holds. For the least n requiring attention at stage s, we toggle $F(n)$.

3.3.3. *Verification.* We now verify that the construction works.

Lemma 3.4. *F is ω^2-c.e.*

Proof. We count the number of times $F(n)$ is toggled. Suppose $s < t$ are two consecutive stages where $F(n)$ is toggled. At stages s and t we have $\delta(0) = M_{\psi(0)}$ (otherwise we would initialize Δ at the beginning of the stage). Therefore, by the minimality of n at both s and t, we have $\delta(n) = M_{\psi(n)}$.

First, if δ is initialized between s and t then $M_{\psi(0)}$ must have increased between s and t. Suppose this does not happen, and suppose further that $A_s \upharpoonright \delta(n)[s] \neq A_t \upharpoonright \delta(n)[s]$.

We claim that in this case, either $\emptyset^{2b}(n)[s] \neq \emptyset^{2b}(n)[t]$ or $M_{\psi(n+1)}[s] \neq M_{\psi(n+1)}[t]$. Since n requires attention at stage t, we have either $\emptyset^{2b}(n)[t] \neq \Delta^A(n)[t]$ or $\delta(n+1)[t] < M_{\psi(n+1)}[t]$ holds. If $\delta(n+1)[t] < M_{\psi(n+1)}[t]$ holds, and since the change $A_s \upharpoonright \delta(n)[s] \neq A_t \upharpoonright \delta(n)[s]$ always allows us to correct $\delta(n+1)$, we can conclude that $M_{\psi(n+1)}[s] \neq M_{\psi(n+1)}[t]$ must hold. On the other hand suppose at stage t we have $\emptyset^{2b}(n)[t] \neq \Delta^A(n)[t]$ instead. Again observe that the change $A_s \upharpoonright \delta(n)[s] \neq A_t \upharpoonright \delta(n)[s]$ will also allow us to correct $\Delta^A(n)[s]$ (if necessary), this means that $\emptyset^{2b}(n)[s] \neq \emptyset^{2b}(n)[t]$ must hold. This proves the claim.

Now assume that δ is not initialized between s and t and that $A_s \upharpoonright \delta(n)[s] = A_t \upharpoonright \delta(n)[s]$. Clearly we have $M_{\psi(n)}[s] = M_{\psi(n)}[t]$, because otherwise $M_{\psi(n)}[t] > M_{\psi(n)}[s] = \delta(n)[s] = \delta(n)[t]$, which contradicts the minimality of n at stage t. At stage s when we toggle $F(n)$ we must obtain a bad recovery at $s+1$, in particular, this means that $A^b[s] \upharpoonright \psi(n) \prec A^b[s+1] \upharpoonright \psi(n)$. Since $M_{\psi(n)}[s] = M_{\psi(n)}[t]$, this means that $A^b[s+1] \upharpoonright \psi(n) \preccurlyeq A^b[t] \upharpoonright \psi(n)$; otherwise we can find some $k < \psi(n)$ such that $A^b(k)[s+1] = 1$ and $A^b(k)[t] = 0$. That means that between stages $s+1$ and t, A has to change below $M_{\psi(n)}[s+1] = M_{\psi(n)}[s] = \delta(n)[s]$, a contradiction.

Putting together the facts in the previous paragraphs, we see that between stages s and t, either $M_{\psi(0)}$ has increased (causing δ to be initialized), or $\emptyset^{2b}(n)[s] \neq \emptyset^{2b}(n)[t]$, or $M_{\psi(n+1)}[s] \neq M_{\psi(n+1)}[t]$, or $A^b[s] \upharpoonright \psi(n) \prec A^b[t] \upharpoonright \psi(n)$. The third case can only happen at most $\psi(n+1)^2$ many times, and the last case can only happen at most $\psi(n)$ many times in a row. Therefore, an ω^2-c.e. approximation to $F(n)$ can be obtained from an ω^2-c.e. index for $\emptyset^{2b}$. (The Recursion Theorem is then applied to this approximation). $\square$

Lemma 3.5. *$\emptyset^{2b} \leq_T^{\omega\text{-c.e.}} A$.*

Proof. Since Δ is reset finitely often, we consider the final value of Δ. $\delta(n)$ is increased only to match the changes in $M_{\psi(n)}$, and so δ is an ω-c.e. function. Fix n and suppose that $\emptyset^{2b}(n) \neq \Delta^A(n)$, and fix a stage in the construction where this disagreement is stable. But this means that n will require attention at every stage after that. This means that at almost every stage, $F(m)$ is toggled for some $m \leq n$, but this contradicts Lemma 3.4. $\square$

This concludes the proof of (i) $\Rightarrow$ (iii) and of Theorem 3.2.

4. Effectivizing domination properties

There are two natural ways to interpret the statement "f is reducible to A", where $A \in 2^\omega$ and $f \in \omega^\omega$. The first is to say that $f = \Phi^A$ for some r-functional, where $r = T, wtt$ or tt. This is the definition we adopt in this paper when we say

$f \leq_r A$. The second way is to declare that $Graph(f) \leq_r A$. If $r = T$ then both interpretations are equivalent. However if $\leq_r$ is a strong reducibility, we have to be careful. In Theorem 4.1 we will show that the second interpretation gives nothing new for $Graph(f) \leq_{wtt} A$ and $Graph(f) \leq_{tt} A$. This justifies our use of the first definition when defining the different variations on effective domination.

Theorem 4.1. *Let A be a Δ^0_2 set. The following are equivalent.*

(i) $A \equiv_T \emptyset'$.
(ii) $Graph(f) \leq_{wtt} A$ *for some strongly dominant f, i.e. f is dominant with a Δ^0_2 g.*
(iii) $Graph(f) \leq_{tt} A$ *for some f where f is dominant with a computable g.*
(iv) *A is $\left(\leq_T, \Delta^0_2\right)$-dominant.*
(v) *A is $(\leq_T, \omega$-c.e.$)$-dominant.*
(vi) *A is $\left(\leq_T, \Delta^0_1\right)$-dominant.*
(vii) *There is some $f \leq_T A$ and some g partial computable relative to $\emptyset'$ such that for each total φ_e, we have $g(e)\downarrow$ and $f(x) > \varphi_e(x)$ for every $x > g(e)$.*

Proof. (i) $\Leftrightarrow$ (iv) is [8, Theorem 4.8]. (iii) $\Rightarrow$ (ii) $\Rightarrow$ (iv) are trivial.

(i) $\Rightarrow$ (iii): Suppose that $\emptyset' = \Gamma^A$ for some Turing functional Γ. Note that $\varphi_e(x)\downarrow$ iff $\langle e, x\rangle \in \emptyset'$. We will describe how to define the tt-reduction Φ. Fix a string $\sigma \in 2^{<\omega}$, and $x \in \omega$. We declare $\Phi^\sigma(\langle x, |\sigma|\rangle) = 1$ if the following conditions hold:

- For every $e < x$, $\Gamma^\sigma(\langle e, x\rangle)\downarrow$.
- For each $e < x$ such that $\Gamma^\sigma(\langle e, x\rangle) = 1$ we have $\varphi_{e,|\sigma|}(x)\downarrow$.
- We have not already declared $\Phi^\tau(\langle x, |\tau|\rangle) = 1$ for any $\tau \subset \sigma$.

Otherwise we declare $\Phi^\sigma(\langle x, |\sigma|\rangle) = 0$. These conditions are computable in σ and x, and thus we can phrase Φ as a Turing functional. It is clear that Φ is consistent, as the output of $\Phi^X(\langle x, y\rangle)$ is determined solely by $X \upharpoonright y$. It is also obvious that Φ is a tt-functional, since for every $\langle e, x\rangle$ and every $\sigma \in 2^x$, $\Phi^\sigma(\langle e, x\rangle)\downarrow$.

Now we argue that $\Phi^A = Graph(f)$ for some $f \in \omega^\omega$. For each x there can be at most one y such that $\Phi^A(\langle x, y\rangle) = 1$ because of the third condition. Since $\Gamma^A = \emptyset'$, for a long enough initial segment $\sigma \subset A$, the first two conditions above must be met. Hence there must be exactly one $y = f(x)$. Finally we argue that for every e and $x > e$, if $\varphi_e(x)\downarrow$ then $\varphi_e(x) < f(x)$. Since $\Gamma^A(\langle e, x\rangle) = \emptyset'(\langle e, x\rangle) = 1$, this means that $\varphi_{e,y}(x)\downarrow$. By the usual convention, $\varphi_{e,y}(x) < y$.

(i) $\Rightarrow$ (vi): Since (iii) trivially implies (vi), we in fact have that (i) to (vi) are equivalent.

(vii) $\Rightarrow$ (iv): Fix $f \leq_T A$ and g partial computable in $\emptyset'$. Since we only really need an upperbound on each $g(e)$, we can assume that there is a computable function $g(e, s)$ which is non-decreasing in variable s, and such that for every e, if $g(e)\downarrow$ then $\lim_s g(e, s) = g(e)$. Now define $\hat{g}(e) = g(e, s)$ for the least s such that either $\forall t > s, g(e, t) = g(e, s)$ or $\varphi_e(s)\uparrow$. Then $\hat{g}$ is total because φ_e is total implies that $\lim_s g(e, s)$ exists. Thus $\hat{g} \leq_T \emptyset'$. Furthermore if φ_e is total then $\hat{g}(e) = g(e)$. $\square$

Proposition 4.2. *Let A be a Δ^0_2 set. Then A is $\left(\leq_{wtt}, \Delta^0_1\right)$-dominant iff $A \geq_{wtt} \emptyset'$.*

Proof. Suppose that $\emptyset' = \Gamma^A$ with computably bounded use. Define $f(x)$ to be the least s such that $\varphi_{e,s}(x)\downarrow$ for every $e < x$ such that $\Gamma^A(\langle e, x\rangle) = 1$. Then $f \leq_T A$ is total and has computably bounded use. Furthermore for every $x > e$, $f(x) > \varphi_e(x)$ if the latter is defined.

Now assume that $f = \Psi^A$ with a computable bound ψ on the use, and such that for every total φ_e, $f(x) > \varphi_e(x)$ for every $x > g(e)$ where g is computable. Now define the sequence of partial computable functions $\{\varphi_{q(n)}\}_{n\in\omega}$ where q is total computable, and for every n,

$$\varphi_{q(n)}(z) = \begin{cases} \uparrow, & \text{if } n \notin \emptyset', \\ s, & \text{if } n \text{ enters } \emptyset' \text{ at stage } s. \end{cases}$$

Then it is clear that for every n, $n \in \emptyset'$ iff $n \in \emptyset'\,[f(g(q(n)) + 1)]$, which is of course wtt-computable from A. □

In contrast to Proposition 4.2, we will later show (in Theorem 4.6) that we can have a wtt-incomplete $(\leq_{wtt}, \omega$-c.e.$)$-dominant set. Nevertheless, together with c.e. $(\leq_{wtt}, \omega$-c.e.$)$-dominance does imply a stronger property than being Turing complete:

Proposition 4.3. *Let A be a c.e. set which is $(\leq_{wtt}, \omega$-c.e.$)$-dominant. Then $A \geq_T^{\omega\text{-c.e.}} \emptyset'$.*

Proof. This proposition is an effective version of [8, Theorem 4.8]. However, due to the fact that this proposition requires the computable bounds for the key steps to be defined and verified, we will present a more direct approach than in the proof given in [8]. Our proof here is also shorter and simpler. Fix a c.e. set A and a wtt-functional Ψ and an ω-c.e. function g such that for every total φ_e, we have $\Psi^A(x) > \varphi_e(x)$ for every $x > g(e)$. Assume the use of Ψ is bounded by a computable function ψ. Fix an ω-c.e. approximation $g(e)[s]$ of $g(e)$, and assume that $g(e)[s]$ is non-decreasing in s.

Our goal is to describe a Turing functional Δ such that $\emptyset' = \Delta^A$ with use bounded by an ω-c.e. function δ. We first define a sequence of integers $\{\mathtt{I}(x, n)\}$ and a sequence of partial computable functions $\varphi_{q(x)}$ for a computable function q, whose index is given by the recursion theorem. For each x, $\mathtt{I}(x, n)$ will be increasing in n, and will only be defined for finitely many n. We now describe how to define $\mathtt{I}(x, n)$ for a fixed x. At stage 0, define $\mathtt{I}(x, 0) = g(q(x))[0]$ and set $\varphi_{q(x)}(y) = 0$ for all $y \leq \mathtt{I}(x, 0)$. Now assume that $\mathtt{I}(x, n) \geq g(q(x))[t_n]$ is defined and $\varphi_{q(x)}(y)$ are defined for all $y \leq \mathtt{I}(x, n)$ at stage t_n. Wait for the first stage $t > t_n$ such that we have either $x \in \emptyset'[t] - \emptyset'[t-1]$ or $g(q(x+1))[t] \neq g(q(x+1))[t-1]$. If t exists, we define $\varphi_{q(x)}(y) = t$ for $y = \mathtt{I}(x, n)+1, \mathtt{I}(x, n)+2, \cdots$, one at at time, until we find some stage $t_{n+1} > t$ such that either A changes below $\psi\left(\max\{\mathtt{I}(x, n), g(q(x))[t]\} + 1\right)$ or $g(q(x))[t_{n+1}] \neq g(q(x))[t]$. Notice that if we find t and begin extending the domain of $\varphi_{q(x)}$, then t_{n+1} must exist: This is because $\varphi_{q(x)}\left(\max\{\mathtt{I}(x, n), g(q(x))[t]\} + 1\right) = t > \Psi^A\left(\max\{\mathtt{I}(x, n), g(q(x))[t]\} + 1\right)[t]$. Once t_{n+1} is found we stop extending the domain of $\varphi_{q(x)}$ and set $\mathtt{I}(x, n+1) = \max dom\left(\varphi_{q(x)}\right)$ or $g(q(x))[t_{n+1}]$, whichever is larger. If $g(q(x))[t_{n+1}] > \max dom\left(\varphi_{q(x)}\right)$ we extend $dom\left(\varphi_{q(x)}\right)$ to make them equal.

First observe that for each x, $\mathtt{I}(x, n)$ is defined for at most $1 + \#$ changes in $g(q(x+1))$ many n. Now define $\delta(x) = \max_{y\leq x} \max_n \psi\left(\mathtt{I}(y, n) + 1\right) + \max_{y\leq x} \max_s \psi\left(g(q(y))[s] + 1\right)$. Clearly δ is ω-c.e. We now argue that $\emptyset'(x)$ can be computed using $\delta(x)$ many bits of A. First fix a stage s after which $g(q(0))$ does not change, and where $A_s \restriction \delta_s(x) = A \restriction \delta_s(x)$. We first claim that $\delta_s(x) = \delta(x)$. Suppose not. Fix the least $y \leq x$ with some least n such that $\mathtt{I}(y, n+1)$ is defined after stage s. (Of course, $\delta_s(x)$ might also increase due to a change in $g(q(z))$ for some z after stage s, but if this is the case then $z > 0$ and the construction will force a new definition of $\mathtt{I}(z-1, m)$).

The procedure to find t_{n+1} has started after stage s (more specifically, the corresponding $t > s$, otherwise we would have waited for the procedure to finish and choose s larger), and either A changes below $\psi(\max\{\mathtt{I}(y,n), g(q(y))[t]\}+1)$ or $g(q(y))[t_{n+1}] \neq g(q(y))[t]$. The latter is not possible, otherwise $y > 0$ and the construction will force a new definition of $\mathtt{I}(y-1, m)$ after stage $t > s$, contradicting the minimality of y. Thus we may assume that A changes below $\psi(\max\{\mathtt{I}(y,n), g(q(y))[t]\}+1)$ after stage t. Again by the minimality of y, we have $g(q(y))[t] = g(q(y))[s]$. This means that $\psi(\max\{\mathtt{I}(y,n), g(q(y))[t]\}+1) \leq \delta_s(x)$, and so it is impossible for A to change below this value after stage $t > s$, a final contradiction.

Now as $\delta_s(x) = \delta(x)$, it is clear that x cannot enter $\emptyset'$ after stage s, otherwise the construction will force a new definition of $\mathtt{I}(x, m)$ after stage s. □

Theorem 4.4. *There exists a c.e. set A such that $A \geq_T^{\omega\text{-c.e.}} \emptyset'$ and A is not even $(\leq_{wtt}, \Delta^0_2)$-dominant.*

Proof. We construct a c.e. set A and an ω-c.e. function δ. At the end we will give a procedure to decide $\emptyset'$ using only δ many bits of A. We wish to meet the requirements

$$R_e : \text{If } \Psi_e^A \text{ is total and } \lim_s g_e(k,s) \text{ exists for every } k \text{ then there is } k \text{ such that}$$
$$\varphi_k \text{ is total and } \varphi_k(x) > \Psi_e^A(x) \text{ for some } x > \lim_s g_e(k,s).$$

Here Ψ_e^A is the e^{th} possible *wtt*-functional with partial computable use function ψ_e. We also assume that $g_e(k,s)$ is non-decreasing in variable s. We define a sequence of partial computable functions $\varphi_{q(e)}$ and assume that the index of the computable function q is given by the recursion theorem. For convenience we let $\mathtt{N}_{e,s}$ be the number of $t < s$ such that $g_e(q(e),t) < g_e(q(e),t+1)$, or e, whichever is larger.

At stage $s = \langle e, s' \rangle$ of the construction we first check if there is any x such that $x \in \emptyset'[s] - \emptyset'[s-1]$, and if so, we enumerate the current value $\delta_s(x)$ into A and lift $\delta_{s+1}(y)$ to a large fresh value for all $y \geq x$. Next, we act for R_e as follows, depending on the first case which applies:

- If $\mathtt{N}_e$ has increased since the last time R_e acted, or if R_e has never acted before, or if $\Psi_e^A(z_e) \geq \varphi_{q(e)}(z_e)$ are both defined, we abandon the previous follower of R_e, and pick a fresh follower z_e, where $\varphi_{k(e)}(z_e)\uparrow$ and $z_e > g_e(k(e), s)$.
- If $\psi_e(z_e)\downarrow$ and $\delta_s(x) < \psi_e(z_e)$ for some least $x > \mathtt{N}_e$, we enumerate $\delta_s(x)$ into A and lift $\delta_{s+1}(y)$ to a large fresh value for all $y \geq x$.
- If $\Psi_e^A(z_e)\downarrow$ and $\varphi_{q(e)}(z_e)\uparrow$, then we set $\varphi_{q(e)}(z_e) = s$; in particular $\varphi_{q(e)}(z_e)$ is now larger than $\Psi_e^A(z_e)[s]$.
- Otherwise, we define $\varphi_{q(e)}(m)$ for the least m not yet in the domain. The output value is irrelevant.

We now check some properties satisfied by the construction. We claim that the function δ is ω-c.e. Notice that $\delta(x)$ can only be increased when acting for some R_e while $\mathtt{N}_e < x$, and hence $e < x$. (Another possibility is due to coding when some $y \leq x$ enters $\emptyset'$, but this happens at most x times). If $\delta(x)$ is increased when acting for R_e, then the same R_e can do this again only if either $\mathtt{N}_e$ increases, or if A changes below $\psi_e(z_e)$. In the latter case, this A change has to be due to the enumeration of $\delta(y)$ for some $y < x$, and the former case can only apply at most x times before $\mathtt{N}_e$ exceeds x. Hence a bound for the number of changes in $\delta(x)$ can be computed recursively.

We now argue that $\emptyset'(x)$ can be computed using at most $\delta(x) + 1$ many bits of A. Find a stage s such that $A_s \upharpoonright \delta_s(x) + 1 = A \upharpoonright \delta_s(x) + 1$. Notice that $\delta_s(x) = \delta(x)$ because any change in $\delta_s(x)$ after stage s is always accompanied by the enumeration of some number $\leq \delta_s(x)$. This means that $x \in \emptyset'$ iff $x \in \emptyset'[s]$, otherwise the construction will enumerate $\delta_s(x)$ into A. Hence $A \geq_T^{\omega\text{-c.e.}} \emptyset'$.

Next we check that R_e is satisfied for each e. Suppose that Ψ_e^A is total and $\lim_s g_e(q(e), s)$ exists. In particular, $\mathtt{N}_{e,s}$ is eventually stable. Since $\delta(x)$ will eventually settle for all $x \leq \mathtt{N}_e$, hence at almost every stage of the form $s = \langle e, s' \rangle$, we must have z_e stable and $\varphi_{q(e)}(z_e) > \Phi_e^A(z_e)$, and we act by the last item. This means that $\varphi_{q(e)}$ will be total. Moreover z_e is always picked larger than $g_e(q(e))$. $\square$

We now investigate the relationship between being $(\leq_{wtt}, \omega\text{-c.e.})$-dominant and being high with respect to the bounded jump. For c.e. sets, domination is a strictly weaker property.

Theorem 4.5. *Each bounded high c.e. set is $(\leq_{wtt}, \omega$-c.e.$)$-dominant.*

Proof. Let A be c.e. and bounded high. In Theorem 3.2 it was shown that this is equivalent to $\emptyset^{2b} \leq_T^{\omega\text{-c.e.}} A$. Proposition 3.1 also shows that $\emptyset^{2b}$ is effectively 1-complete amongst the ω^2-c.e. sets. In this proof we will build an ω^2-c.e. set F, and by the Recursion Theorem we assume we know the ω^2-c.e. index for F during the construction. Hence, we may fix Δ as a witness for $F = \Delta^A$ with use bounded by an ω-c.e. function δ. We assume an approximation of δ such that for every x and s, $\delta_{s+1}(x) \geq \delta_s(x)$.

We describe how to define F. We begin with $F_0(\langle e, x \rangle) = 0$ for all e, x. We assume, by speeding up the construction, that at the beginning of every stage s we have $F(\langle e, x \rangle) = \Delta^A(\langle e, x \rangle)$, and that if we toggle $\langle e, x \rangle$ in F at stage s (this means that we set $F_{s+1}(\langle e, x \rangle) = 1 - F_s(\langle e, x \rangle))$, Δ^A instantly corrects itself at the beginning of the next stage. We do the following actions uniformly for each e. At stage s, pick the least number x such that:

- $\varphi_e(z) \downarrow$ for every $\delta(\langle e, x \rangle) < z \leq \delta(\langle e, x+1 \rangle)$, and
- either $\delta_{s^-}(\langle e, x+1 \rangle) \neq \delta_s(\langle e, x+1 \rangle)$ or s^- does not exist.

Here s^- is the previous stage where we toggled $\langle e, x \rangle$. If this least number x exists at stage s, toggle $\langle e, x \rangle$.

Now observe that the number of times we toggle $\langle e, x \rangle$ is at most 1+ the number of times $\delta(\langle e, x+1 \rangle)$ increases. Since δ is ω-c.e. this means that F is $\omega + 1$-c.e. and we can certainly convert the description above into an ω^2-c.e. index for F: To be explicit, we begin the ω^2-c.e. approximation to F by setting the initial value of the ordinal bounding function of F to be ω on every input. Then we wait for the reduction from $F \leq_1 \emptyset^{2b}$ to converge on more and more inputs until we have enough information to compute the number of times we will need to toggle $\langle e, x \rangle$. Then we can decrease the value of the ordinal bounding function on input $\langle e, x \rangle$ to this number.

We point out a very subtle fact here. Even though F is ultimately ω-c.e. however, in the construction, we have to work with a ω^2-c.e. index for F and apply the Recursion Theorem with respect to the ω^2-c.e. index for F. This is because we have to wait for the 1-reduction from $F \leq_1 \emptyset^{2b}$ to reveal itself to be total before we can even compute an index for Δ and δ and hence compute the number of times we have to toggle each $F(\langle e, x\rangle)$. If we do not begin the construction by initially starting the ordinal bounding function for F at the value ω, then the 1-reduction from $F \leq_1 \emptyset^{2b}$ will be partial, and we will not be able to compute the number of times each $F(\langle e, x\rangle)$ is to be toggled.

Next we show how to define $f \leq_{wtt} A$ and an ω-c.e. function g witnessing the $(\leq_{wtt}, \omega\text{-c.e.})$-dominance of A. Let $f(x)$ output the first stage of the construction where $A \upharpoonright x$ is stable, and $g(e) = \delta(\langle e, 0\rangle)$. Clearly $f \leq_{wtt} A$ and g is ω-c.e. Fix e such that φ_e is total, and fix $z > g(e) = \delta(\langle e, 0\rangle)$. We wish to show that $f(z) > \varphi_e(z)$. Let x be such that $\delta(\langle e, x\rangle) < z \leq \delta(\langle e, x+1\rangle)$ and let s_0 be the first stage in the construction where $\delta(\langle e, x+1\rangle)$ does not increase anymore. Let s_1 be the first stage where $\varphi_e(z)$ first converges. It suffices to show that $A \upharpoonright z$ is not yet stable at stage s_1, because then $f(z) > s_1 > \varphi_{e,s_1}(z)$. Suppose that $s_0 \leq s_1$. It is easy to verify that at or after stage s_1 we must toggle $\langle e, x\rangle$ at least once more, which means that A must change below $\delta(\langle e, x\rangle) < z$ after stage s_1. On the other hand if $s_0 > s_1$ then it is also easy to see that we must toggle $\langle e, x\rangle$ at least one more time after s_0. In either case, $A \upharpoonright z$ is not yet stable at stage s_1. □

Theorem 4.6. *There exists a $(\leq_{wtt}, \omega$-c.e.$)$-dominant c.e. set which is not bounded high.*

Proof. We shall build the c.e. set A, ω^2-c.e. set F, *wtt*-functional Φ and an ω-c.e. function g satisfying the following requirements:

$$N_e : \rho_e \text{ is total } \Rightarrow \Psi_e^A \neq F,$$
$$P_e : \varphi_e \text{ is total} \Rightarrow \Phi^A(x) > \varphi_e(x) \text{ for all } x > g(e).$$

Here, Ψ_e is the e^{th} Turing functional with use bounded by ψ_e. We assume a computable approximation of ψ_e such that the number of changes in $\psi_e(x, s)$ is bounded by the (partial) computable function ρ_e. As $\emptyset^{2b}$ is 1-complete amongst the ω^2-c.e. sets, the N-requirements ensure that $A \not\geq_T^{\omega\text{-c.e.}} \emptyset^{2b}$, and therefore A is not bounded high.

Without loss of generality we assume that $dom(\varphi_e)$ is an initial segment of ω at every stage. We arrange the requirements $N_0 < P_0 < N_1 < \cdots$. At stage $s > 0$ we define what it means for a requirement to *require attention*:

- N_e requires attention at stage s if $\rho_e(e)\downarrow$ and $\Psi_e^A(e)[s]\downarrow = F(e)$.
- P_e requires attention at stage s if $s > 2s^-$, $\max dom(\varphi_e) > s^-$ and for every $k \leq s^-$ where N_k has acted before, we want $\max dom(\varphi_e) > \psi_k(k, s)$. Here s^- is the previous stage where P_e had acted, or the current value of $g(e)+1$, whichever is larger.

We define the intervals $I_x = \left[2^x, 2^{x+1}\right)$ for $x \in \omega$. The functional Φ^A will have computable use $\max I_x$, and every time we wish to change the value of $\Phi^A(x)$ we will enumerate the next element of I_x into A and redefine $\Phi^A(x)$; of course we have to ensure that we only request for at most $2^x - 1$ many changes to the value of $\Phi^A(x)$ during the construction.

Construction. At stage $s = 0$ of the construction we set $g(e)[s] = e$ for every e. Also define $\Phi^A(x) = 0$ for every x. Now suppose $s > 0$. We pick the highest priority requirement which requires attention at stage s (and which can act), and act for it by doing the following. If the requirement is N_e, we *toggle* $F(e)$ by defining $F_{s+1}(e) = 1 - F_s(e)$, and increase $g(k)[s] = s$ for every P_k such that $k > \max\{\rho_e(e), e\}$. If the requirement is P_e, then for every value of x such that $g(e)[s] < x \leq \max dom\,(\varphi_e)$ and where $\Phi^A(x) \leq \varphi_e(x)$, we increase the value of $\Phi^A(x)$ to s (by first enumerating the next element of I_x into A).

Verification. We now verify that the construction works. We argue that N_e is satisfied, and acts only finitely often. Suppose N_e is not satisfied, and that $\Psi_e^A = F$ with ρ_e total and bounds the number of changes in ψ_e. Hence N_e will require attention at almost every stage of the construction. Since each N_k of higher priority only acts finitely often (inductive hypothesis), and each P_k of higher priority will only be allowed to act with an increasing delay, this means that N_e will infinitely often be allowed to act. Let s_0 be any stage where N_e is allowed to act. We claim that before $\psi_e(e,s)$ next increases, there are at most $2(\max\{\rho_e(e), e\}+1)$ many stages $t > s_0$ such that we change A below $\psi_e\,(e, s_0)$. At stage s_0 we increase $g(k) > \psi_e\,(e, s_0)$ for every $k > \max\{\rho_e(e), e\}$. Therefore at such a stage $t > s_0$, we must be acting for some P_k where $k \leq \max\{\rho_e(e), e\}$ (noting that $\min I_x > x$ for all x). But each such P_k can only contribute at most two different $t > s_0$ - the second time P_k acts after stage s_0 we must see $\max dom(\varphi_k) > \psi_e\,(e, s_0)$. The third and subsequent time P_k acts after s_0 (and before $\psi_e(e,s)$ increases) will only increase $\Phi^A(x)$ for $x > \psi_e\,(e, s_0)$. Hence there are at most $2(\max\{\rho_e(e), e\} + 1)$ many such stages t. Since $\psi_e(e,s)$ can increase at most $\rho_e(e)$ many times, this means that $F(e)$ is toggled at most $2(\max\{\rho_e(e), e\} + 1)\,(\rho_e(e) + 1)$ many times. Hence N_e will act only finitely often, and F is clearly $\omega + 1$-c.e. Since N_e only acts finitely often, it is clear that the final toggle by N_e must diagonalize without a further $\Psi_e^A(e)$ change, a contradiction.

We now verify that the P requirements succeed. First of all, g is ω-c.e. because we only increase $g(k)$ at most $2(\max\{\rho_e(e), e\} + 1)\,(\rho_e(e) + 1)$ many times for each N_e such that $k > \max\{\rho_e(e), e\}$. There are at most k many such N_e possible, and for each e we will increase $g(k)$ at most $2(\max\{\rho_e(e), e\}+1)\,(\rho_e(e) + 1) < 2(k+1)^2$ many times. So the number of changes to $g(k)$ is bounded by $2k(k + 1)^2$. For each x, we will request to increase the value of $\Phi^A(x)$ at most once for each P_k where $k = g(k)[0] < x \leq 2^x - 1$. Thus, we will never run out of space in I_x. Now we fix P_e and argue that it works. Suppose φ_e is total, and assume for a contradiction that there are only finitely many stages we act for P_e. Let s_1 be the last time we acted for P_e or the stable value of $g(e) + 1$, whichever is larger. For all large enough s, we have $s^- = s_1$, and it is clear that P_e will require attention at all large enough s. This means that we must act for P_e after stage s_1, a contradiction. Since P_e acts infinitely often, it is clear by the construction that $\Phi^A(x) > \varphi_e(x)$ for every $x > g(e)$. $\square$

Given that Sacks' Jump inversion fails for the bounded jump, we will next describe how to directly construct a c.e. set which is bounded high and wtt-incomplete. As a corollary we obtain the failure of the analogue of the Jump Theorem for the bounded jump (see [2]).

Proposition 4.7. *There is a c.e. set A which is bounded high and $A <_{wtt} \emptyset'$.*

Proof. We wish to construct a c.e. set A and a Turing functional Δ such that $\emptyset^{2b} = \Delta^A$ with use bounded by an ω-c.e. function δ. We have to satisfy the requirements

$$R_e : \psi_e \text{ is total } \Rightarrow F \neq \Psi_e^A,$$

where F is an ω-c.e. set we build and Ψ_e is a Turing functional whose use is bounded by the partial computable function ψ_e.

We define the intervals $I_x = \left[2^x, 2^{x+1}\right)$ for $x \in \omega$. The functional Δ^A will have use $\max I_{\langle x,y\rangle}$ for some y. For simplicity we denote the (current) use of $\Delta^A(x)$ as $\max I_{\langle x,\delta(x)\rangle}$. Every time we wish to change the value of $\Delta^A(x)$ we will enumerate the next element of $I_{\langle x,\delta(x)\rangle}$ into A and redefine $\Delta^A(x)$; of course we have to ensure that we only request for at most $2^{\langle x,\delta(x)\rangle} - 1$ many changes to the value of $\Phi^A(x)$, or we have to increase $\delta(x)$.

We fix an ω^2-c.e. approximation of $\emptyset^{2b}$. At stage s let $k(x,s)$ and $j(x,s)$ be such that the ordinal bound in the ω^2-c.e. approximation of $\emptyset^{2b}(x)$ is $\omega \cdot k(x,s) + j(x,s)$. Without loss of generality we assume that $k(x,0) = x$ for every x.

Construction. At stage $s = 0$ we set $\delta(x) = x$ for every x. At stage $s > 0$, we do the following:

(i) For each $x < s$ such that $k(x,s) \neq k(x,s-1)$, enumerate the next element of $I_{\langle x,\delta(x)\rangle}$ into A and pick a new fresh value for $\delta(x)$, and redefine $\Delta^A(x)$ with the corresponding new use. Initialize each R_e for $e \geq x$ by setting the value of its follower z_e undefined.

(ii) For each $x < s$ such that $j(x,s) \neq j(x,s-1)$, enumerate the next element of $I_{\langle x,\delta(x)\rangle}$ into A and redefine $\Delta^A(x)$ with a different value, if necessary.

(iii) Pick the least e such that R_e requires attention: This means that either the follower z_e is not yet picked, or $\psi_e(z_e)\downarrow$ and $F(z_e) = \Psi_e^A(z_e)$. For the least such e, we act for R_e by doing the following. If $z_e\uparrow$, we pick a fresh value for z_e. Otherwise if $\psi_e(z_e)\downarrow$ and $F(z_e) = \Psi_e^A(z_e)$ we toggle $F(z_e)$ by setting $F_s(z_e) = 1 - F_{s-1}(z_e)$. If there are any $x > e$ such that $I_{\langle x,\delta(x)\rangle}$ contains an element less than $\psi_e(z_e)$, we enumerate the next element of $I_{\langle x,\delta(x)\rangle}$ into A and pick a new fresh value for $\delta(x)$, and redefine $\Delta^A(x)$ with the corresponding new use.

Verification. We now verify that the construction works. First of all, observe that R_e is initialized only when $k(x)$ changes for some $x \leq e$. Since we assumed $k(x,0) = x$ and $k(x,-)$ cannot increase, this means that R_e is initialized at most $\frac{e(e+1)}{2}$ many times. Next we count the number of times $\delta(x)$ can be increased, for a fixed x. This can be increased in step (i) or (iii) of the construction. In step (i) we must have a $k(x)$ change, so this is at most x many times. In step (iii) it must be due to some R_e for $e < x$ acting. Each such e can increase $\delta(x)$ at most once unless R_e is initialized and gets a new follower z_e. Since each R_e is initialized at most $\frac{e(e+1)}{2}$ many times, we can certainly compute a bound for the number of times $\delta(x)$ is increased. Hence δ is ω-c.e.

Next, we check that $\Delta^A(x)$ is well defined with use bounded by $\max I_{\langle x,\delta(x)\rangle}$. This is because each time we wish to update the value of $\Delta^A(x)$ or increase the value of $\delta(x)$ we always enumerate the next element of $I_{\langle x,\delta(x)\rangle}$ into A. We need to check that we do not run out of elements in $I_{\langle x,\delta(x,s)\rangle}$ for each s. Only in step (ii) do we use up an element of $I_{\langle x,\delta(x)\rangle}$ without also increasing $\delta(x,s)$. Let $t < s$ be the stage where $\delta(x,s)$ was picked, and obviously $k(x)$ cannot have changed between t and s, otherwise step (i) would have increased $\delta(x)$. Therefore the total number of elements we would use up from $I_{\langle x,\delta(x)\rangle}$ is at most $j(x,t)$ (since $j(x)$ cannot increase unless $k(x)$ decreases), plus possibly one more last time if we ever increase $\delta(x)$. Since $\delta(x,s)$ is picked fresh at stage t, the size of $I_{\langle x,\delta(x,s)\rangle}$ is large enough, so we never use up all elements of $I_{\langle x,\delta(x,s)\rangle}$. So indeed, $\Delta^A(x)$ is well defined with use bounded by $\max I_{\langle x,\delta(x)\rangle}$. Clearly $\Delta^A(x) = \emptyset^{2b}(x)$, because each change in the approximation of $\emptyset^{2b}(x)$ is followed by a change in either $k(x)$ or $j(x)$, and either step (i) or step (ii) will ensure we redefine $\Delta^A(x)$.

Next we check that F is ω-c.e. Fix z_e and we need to figure out how many times we might toggle $F(z_e)$. Let u be the stage where z_e is picked. The very first time we act for R_e with the follower z_e after stage u we ensure that we move $\min I_{\langle x,\delta(x)\rangle}$ beyond $\psi_e(z_e)$ for all $x > e$. Therefore, the only reason we may have to toggle $F(z_e)$ again must be due to $\Delta^A(x)$ being redefined for some $x \leq e$. We need to count how many times this can happen for each $x \leq e$. If $\Delta^A(x)$ is redefined with $\delta(x)$ increased, then as the new value of $\delta(x)$ is picked fresh, this same x cannot again cause $F(z_e)$ to be toggled. If $\Delta^A(x)$ is redefined with no increase in $\delta(x)$, then this is due to step (ii) of the construction, and between stage u and this action, $k(x)$ cannot have changed, otherwise we would initialize R_e. Therefore the number of times this can happen is bounded by $j(x,u)$. Since z_e is picked fresh at stage u, we have $j(x,u) < z_e$. Thus the number of times $F(z_e)$ is toggled is at most $(e+1)(z_e+1)$. Hence F is ω-c.e.

Finally we verify that each R_e is satisfied. Each R requirement is initialized only finitely often (in fact, only $\frac{e(e+1)}{2}$ many times) and hence must settle on a final follower z, and only toggle $F(z)$ finitely many times. Fix e such that ψ_e is total. Then $\psi_e(z_e)\downarrow$ for the final follower z_e of R_e, and step (iii) of the construction ensures that $F(z_e) \neq \Psi_e^A(z_e)$. □

Corollary 4.8 (Anderson, Csima, Lange [2])**.** *There are c.e. sets A and B such that $A^b \leq_1 B^b$ and $A \not\leq_{wtt} B$.*

Proof. Take A to be $\emptyset'$ and B to be any wtt-incomplete c.e. set that is bounded high from Proposition 4.7. By Theorem 3.2, $B^b \geq_1 A^b$. □

Theorem 4.9. *There exists a $\left(\leq_{wtt}, \Delta_2^0\right)$-dominant c.e. set A such that $A \not\geq_T^{\omega\text{-c.e.}} \emptyset'$.*

Proof. The proof is very similar to the proof of Theorem 4.6. We follow the proof of Theorem 4.6 closely. We shall build the c.e. set A, ω-c.e. set F, wtt-functional Φ and a Δ_2^0 function g satisfying the following requirements:

$$N_e : \rho_e \text{ is total } \Rightarrow \Psi_e^A \neq F,$$
$$P_e : \varphi_e \text{ is total} \Rightarrow \Phi^A(x) > \varphi_e(x) \text{ for all } x > g(e).$$

Here, Ψ_e is the e^{th} Turing functional with use bounded by ψ_e. We assume a computable approximation of ψ_e such that the number of changes in $\psi_e(x,s)$ is bounded by the (partial) computable function ρ_e. We also require the convention that $\psi_e(x,s) < \psi_e(x+1,s)$ for all e,x,s. As $\emptyset'$ is wtt-complete amongst the ω-c.e. sets, the N-requirements ensure that $A \not\geq_T^{\omega\text{-c.e.}} \emptyset'$. Notice that the reducibility $\leq_T^{\omega\text{-c.e.}}$ is in general not transitive, however, $A \geq_T^{\omega\text{-c.e.}} \emptyset' \geq_{wtt} F$ *does* imply that $A \geq_T^{\omega\text{-c.e.}} F$.

In Theorem 4.6, the constructed set F is $\omega+1$-c.e. and g is ω-c.e. whereas in the current proof, the set F is ω-c.e. while g is allowed to be merely Δ_2^0. Now we need some new ingredients. Instead of a single follower, N_e will require $e+1$

many followers. These will be $\langle e, d_0\rangle, \langle e, d_1\rangle, \cdots, \langle e, d_e\rangle$, where $d_0 = 0$ and $d_{i+1} = \rho_e(\langle e, d_i\rangle)$ for all $i < e$.

We arrange the requirements $N_0 < P_0 < N_1 < \cdots$. At stage $s > 0$ we define what it means for a requirement to *require attention*:

- N_e requires attention at stage s if for all $i \leq e$, $\rho_e(\langle e, d_i\rangle) \downarrow$ and $\Psi_e^A(\langle e, d_i\rangle)[s] \downarrow = F(\langle e, d_i\rangle)$.
- P_e requires attention at stage s if $s > 2s^-$, $\max dom(\varphi_e) > s^-$ and for every $k \leq s^-$ where N_k has acted before, we want $\max dom(\varphi_e) > \psi_k(\langle k, d_k\rangle, s)$. Here s^- is the previous stage where P_e had acted, or the current value of $g(e) + 1$, whichever is larger.

As before, we define the intervals $I_x = \left[2^x, 2^{x+1}\right)$ for $x \in \omega$. The functional Φ^A will have computable use $\max I_x$, and every time we wish to change the value of $\Phi^A(x)$ we will enumerate the next element of I_x into A and redefine $\Phi^A(x)$; of course we have to ensure that we only request for at most $2^x - 1$ many changes to the value of $\Phi^A(x)$ during the construction.

Construction. At stage $s = 0$ of the construction we set $g(e)[s] = e$ for every e. Also define $\Phi^A(x) = 0$ for every x. Now suppose $s > 0$. We pick the highest priority requirement which requires attention at stage s (and which can act), and act for it by doing the following. If the requirement is N_e, we toggle $F(\langle e, d_i\rangle)$ where $i \leq e$ is the least such that there are at most i many indices $k < e$ with the property that $x_k \leq \psi_e(\langle e, d_i\rangle)$ where x_k is the least element to be enumerated the next time P_k acts (see below). Note that $i = e$ has the property so the least such i must exist. Also we increase $g(k)[s] = s$ for every P_k such that $k \geq e$.

If the requirement is P_e, then for every value of x such that $g(e)[s] < x \leq \max dom(\varphi_e)$ and where $\Phi^A(x) \leq \varphi_e(x)$, we increase the value of $\Phi^A(x)$ to s (by first enumerating the next element of I_x into A).

Verification. We now verify that the construction works. We first check that F is ω-c.e. Fix e, and we shall examine the number of times where each $F(\langle e, d_i\rangle)$ can be toggled during the construction. We claim that for $0 < i \leq e$, between two successive toggles of $F(\langle e, d_i\rangle)$, $\psi_e(\langle e, d_{i-1}\rangle)$ must change. Fix such an i, and assume that $F(\langle e, d_i\rangle)$ is toggled at stages $s_0 < s_1$. At stage s_0, there are at most i many indices $k < e$ with the property that $x_k \leq \psi_e(\langle e, d_i\rangle)$. But at stage s_0 we also increase $g(k) > \psi_e(\langle e, d_i\rangle, s_0)$ for every $k \geq e$. This means that between s_0 and s_1 we must have P_k acting for one of these i many indices $k < e$. If $\psi_e(\langle e, d_{i-1}\rangle, s_0) = \psi_e(\langle e, d_{i-1}\rangle, s_1)$ did not change between s_0 and s_1, then at stage s_1 there will be at most $i - 1$ many indices k left with $x_k \leq \psi_e(\langle e, d_{i-1}\rangle, s_0) = \psi_e(\langle e, d_{i-1}\rangle, s_1)$. So at stage s_1 we would have not have chosen to toggle $F(\langle e, d_i\rangle)$. For $i = 0$, we can only toggle $F(\langle e, d_0\rangle)$ if there are no x_k smaller than the ψ_e-use, so $F(\langle e, d_0\rangle)$ is toggled at most once. Therefore, the number of times that $F(\langle e, d_{i+1}\rangle)$ is toggled is bounded by $\rho_e(\langle e, d_i\rangle) = d_{i+1} < \langle e, d_{i+1}\rangle$. Thus, F is ω-c.e.

The fact that each N is satisfied and acts only finitely often is as before; basically the final toggle on any follower of N must result in a permanent diagonalization.

We now verify that the P requirements succeed. First of all, $g(k)$ is only increased when N_e acts for some $e \leq k$, and each N_e acts only finitely often. Hence g is Δ^0_2. For each x, we will request to increase the value of $\Phi^A(x)$ at most once for each P_k

where $k = g(k)[0] < x \leq 2^x - 1$. Thus, we will never run out of space in I_x. Now we fix P_e and argue that it works. Suppose φ_e is total, and assume for a contradiction that there are only finitely many stages we act for P_e. Let s_1 be the last time we acted for P_e or the stable value of $g(e)+1$, whichever is larger. For all large enough s, we have $s^- = s_1$, and it is clear that P_e will require attention at all large enough s. This means that we must act for P_e after stage s_1, a contradiction. Since P_e acts infinitely often, it is clear by the construction that $\Phi^A(x) > \varphi_e(x)$ for every $x > g(e)$. □

References

[1] B. Anderson and B. Csima. A bounded jump for the bounded Turing degrees. *Notre Dame Journal of Formal Logic*, 55(2):245–264, 2014.

[2] B. Anderson, B. Csima, and K. Lange. Bounded low and high sets. *Archive for Mathematical Logic*, 2017. To appear.

[3] B. Csima, R. Downey, and K.M. Ng. Limits on jump inversion for strong reducibilities. *Journal of Symbolic Logic*, 76(4):1287–1296, 2011.

[4] R. Downey and N. Greenberg. A hierarchy of computably enumerable degrees. Submitted.

[5] R. Downey and N. Greenberg. A transfinite hierarchy of lowness notions in the computably enumerable degrees, unifying classes, and natural definability. Submitted.

[6] R. Downey, N. Greenberg, and R. Weber. Totally ω-computably enumerable degrees and bounding critical triples. *Journal of Mathematical Logic*, 7:145–171, 2007.

[7] R. Downey, C. Jockusch, and M. Stob. Array nonrecursive sets and multiple permitting arguments. *Recursion theory week (Oberwolfach, 1989), Lecture Notes in Math*, 1432:141–173, 1990.

[8] R. Downey, A. Melnikov, and K.M. Ng. Abelian p-groups and the Halting problem. *Annals of Pure and Applied Logic*, 167(11):1123–1138, 2016.

[9] Kenneth Harris. PhD thesis, University of Chicago.

[10] K.M. Ng, F. Stephan, Y. Yang, and L. Yu. The computational aspects of hyperimmunefree degrees. *Proceedings of the 12th Asian Logic Conference*, pages 271–284, 2012.

[11] R.I. Soare. *Recursively enumerable sets and degrees : a study of computable functions and computably generated sets.* Perspectives in mathematical logic. Berlin ; New York : Springer-Verlag, c1987., 1987.

(Ng, Yu) School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371, Republic of Singapore