On equivalence relations and bounded turing degrees

Yu, Hongyuan

2018

Yu, H. (2018). On equivalence relations and bounded turing degrees. Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/75914>

<https://doi.org/10.32657/10356/75914>

**ON EQUIVALENCE RELATIONS AND BOUNDED
TURING DEGREES**

YU HONGYUAN

School of Physical and Mathematical Sciences

A thesis submitted to Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2018

Acknowledgements

First and foremost, I would like to show my warm gratitude to my supervisor, Prof. Ng Keng Meng for his excellent guidance, selfless help, continuous encouragement and patience throughout my Ph.D. studies. I am very grateful for his friendship and for giving me enough chances to improve my taste and confidence in doing good research. His deep intuition and insightful comments regarding my research topics influenced me a lot. For all above and more, I am indebted.

Besides my supervisor, I would like to thank Prof. Xing Chaoping, Prof. Wu Guohua, who taught me the graduate courses in the last few years. I also want to thank Prof. Chan Song Heng for there great help on many graduate matters.

I would like to extend my appreciation to Prof. Yang Yue and Prof. Frank Stephan, who have exposed me to many research topics in logic seminars and workshops at National University of Singapore. Special thanks goes to Prof. Yang Yue for providing me the opportunity to give a report in the logic seminar.

I would also like to thank my friends, Zhang Yichi, Yuan Bowen for their help in Ph.D. studies. This is a good opportunity to thank Dr. Lim Kay Jin, Dr. Ku Cheng Yeaw and Ms. Chan Hui Yee for their great help on my dissertation issues and many other academic matters.

During my graduate studies, I was supported by an NTU Research Scholarship. I would like to take this opportunity to thank Nanyang Technological University for the financial support.

The last but not least, I am deeply grateful to my parents and my sister for their devoted love, continuous support and persistent encouragement.

Chapters 2,3,4 and 5 of this thesis resulted in two publications and are co-authored with my supervisor, Prof. Ng Keng Meng.

Contents

ACKNOWLEDGEMENTS	iii
ABSTRACT	vii
1 GENERAL INTRODUCTION	1
1.1 Preliminaries	1
1.2 The Ershov hierarchy	3
1.3 The Recursion Theorem	4
1.4 Organization of the thesis	5
2 THE DEGREE STRUCTURE OF EQUIVALENCE RELATIONS UNDER COMPUTABLE REDUCIBILITY	7
2.1 Introduction	7
2.1.1 Motivations	7
2.1.2 Preliminaries	12
2.1.3 Organization of the chapter	14
2.2 Enumerations of a class and universality	15
2.2.1 n -c.e. equivalence relations	15
2.2.2 ω -c.e. equivalence relations	17
2.2.3 Π_1^0 equivalence relations	19
2.3 Downward density and c.e. equivalence relations	23
2.4 Downward density and Π_1^0 equivalence relations	32
2.5 Upward density	45

3	SACKS JUMP INVERSION FOR THE WEAK TRUTH TABLE DEGREES AND THE BOUNDED JUMP	51
3.1	Introduction	51
3.1.1	Motivations and preliminaries	51
3.1.2	Organization of the chapter	52
3.2	The proof of Theorem 3.1.2	53
3.2.1	Requirements and notation	53
3.2.2	Strategy for a single R_e	54
3.2.3	Construction	56
3.2.4	Verification	57
4	CHARACTERIZING THE C.E. BOUNDED HIGH SETS	61
4.1	Introduction	61
4.1.1	Motivations and preliminaries	61
4.1.2	Organization of the chapter	62
4.2	A useful lemma	62
4.3	The characterization	65
4.3.1	Proof of (i) $\Rightarrow$ (iii)	66
5	EFFECTIVIZING DOMINATION PROPERTIES	71
5.1	Introduction	71
5.1.1	Motivations and preliminaries	71
5.1.2	Organization of the chapter	75
5.2	Results and separations	75

Abstract

This thesis examines three areas in computability theory. In Chapter 2 we look at certain classes of equivalence relations under computable reducibility. In Chapters 3 and 4 we examine the bounded jump operator and the notion of highness for the bounded jump. In Chapter 5 we look at different ways of effectivizing the properties of a dominant function.

In Chapter 2 we will examine the α -c.e. and Δ_2^0 equivalence relations, for $\alpha \leq \omega$, under computable reducibility. We investigate questions such as the existence of a universal element of each class, the issue of having an enumeration of the class, as well as structural properties such as upward and downward density of the class.

In Chapter 3 we prove that the analogue of Sacks jump inversion for the weak truth-table degrees and the bounded jump does not hold. This contrasts with an earlier result [1] that the analogue of Shoenfield jump inversion does hold. Our result answers a question in [1].

In Chapter 4 we characterize the c.e. sets that are bounded high as those sets that compute the Halting problem with a Turing functional where the use is bounded by an ω -c.e. function. This means that every c.e. bounded high set is already Turing complete, and certainly cannot be low, answering a question in [2].

In Chapter 5 we suggest different ways of strengthening the property of a c.e. set being dominant. We provide separations of these different notions and show that together with bounded highness, these different properties form a proper hierarchy.

Chapter 1

GENERAL INTRODUCTION

1.1 Preliminaries

This thesis is concerned with the subject of computability theory, which is a branch of mathematical logic. Classically, computability theory began as a way of understanding the algorithmic content of mathematical objects and mathematical proofs. The main model of computation is that of a *Turing machine*. This can be extended to include an oracle read-only tape, which gives rise to the notion of *relative computability*.

The most fundamental concept in computability theory is that of Turing reducibility. This is a way of measuring relative computability between two sets $A, B \subseteq \mathbb{N}$. We say that $A \leq_T B$, or A can be computed from B , if there is a finite mechanical procedure, allowing finite but otherwise unrestricted access to knowledge of the oracle B , such that the membership $n \in A$ can be decided in finitely many steps. The use of a computation is the largest bit position of the oracle accessed during the computation. A set $A \subseteq \mathbb{N}$ is computable if it can be computed without any queries to the oracle. The e^{th} partial computable function is denoted as φ_e , while the e^{th} Turing functional is Φ_e .

This pre-ordering on $\mathcal{P}(\mathbb{N})$ induces an equivalence relation $\equiv_T$ in the natural way, where $A \equiv_T B$ iff $A \leq_T B$ and $B \leq_T A$. These equivalence classes are called

Turing degrees. We write the Turing degree of a set A as $\deg_T(A)$, or sometimes $\mathbf{a}$. The resulting quotient structure $(\mathcal{D}, \leq_T)$ is an upper semilattice, though not a lattice. The bottom element in $(\mathcal{D}, \leq_T)$ is $\mathbf{0}$, which is the Turing degree of the computable sets. At the next level of the arithmetical hierarchy are the computably enumerable (c.e.) sets. A non-empty set is c.e. if there is an effective procedure which at each stage possibly outputs a member of the set, not necessarily in any specific order. The class of c.e. sets arises naturally in the study of proofs; the set of consequences of a computable set of axioms can be computably enumerated. The e^{th} c.e. set in the canonical listing of all c.e. sets, is denoted by W_e . One of the most natural examples of a c.e. set is the halting problem:

$$\emptyset' = \{e \in \mathbb{N} \mid \varphi_e(e) \downarrow\}.$$

If one performs this enumeration relative to a set A then one gets the halting problem relative to A , denoted by A' . This is called the *jump operator*, and induces a function defined on $(\mathcal{D}, \leq_T)$ in the natural way.

Kleene and Post [24] formally introduced the jump as an operator on degrees. In the paper, they asked a number of fundamental questions about $(\mathcal{D}, \leq_T)$, including the definability of the jump operator, the density of $(\mathcal{D}, \leq_T)$ and the range of the jump operator. Friedberg [17] answered the range question by showing that the range of the jump operator is the class of degrees computing $\mathbf{0}'$. Shoenfield [29] then showed that the range of the jump operator when the domain is restricted to the degrees below $\mathbf{0}'$, is exactly the class of degrees c.e. in and above $\mathbf{0}'$. Since the jump $\mathbf{a}'$ of any degree $\mathbf{a}$ is necessarily above $\mathbf{0}'$, and if $\mathbf{a} \leq \mathbf{0}'$ then necessarily $\mathbf{a}'$ is c.e. in and above $\mathbf{0}'$, these two old theorems showed that the range of the jump operator (even on a restricted domain) is exactly what we expect. Sacks [28] then showed that in fact, the range of the jump operator restricted to the c.e. sets, is also the class of degrees c.e. in and above $\mathbf{0}'$. These three jump inversion results have other consequences, and lead for instance, to the understanding of the question of the rigidity of $(\mathcal{D}, \leq_T)$, and also showed that the c.e. degrees can be stratified non-trivially into subclasses according to the complexity of the jump.

Finally we state two usual conventions in computability theory. Firstly, we will associate finite binary strings with code numbers for them. Secondly, we can view a real number as an infinite binary sequence, and it can also be viewed as a subset of the natural numbers via the standard identification. For other standard notations and basic computability notions, we refer the reader to Soare [30] or Cooper [7].

1.2 The Ershov hierarchy

A different way of categorizing the class of sets computed from $\emptyset'$ is by the Ershov hierarchy. By Shoenfield limit lemma [29], for any set A , $A \leq_T \emptyset'$ if and only if it is computably approximable, that is, there is a computable function f such that $\lim_s f(x, s) = A(x)$ for all x .

We can classify the complexity of a set computable from $\emptyset'$ set by the complexity of a computable approximation of the set. A set A is n -c.e. for $n \in \omega$ if there is a computable function f such that $\lim_s f(x, s) = A(x)$ for all x , where the number of mind changes is bounded by n : $\#\{s \mid f(x, s) \neq f(x, s+1)\} \leq n$, and that $f(x, 0) = 0$. It is ω -c.e. if the number of mind changes is bounded by a computable function rather than the constant n . By a theorem of Post, a set is Δ_2^0 if the number of mind changes is finite. The idea is that the more complex a set is, the more changes is required in any approximation of the set.

We can extend the definition of an ω -c.e. set to an α -c.e. set for an ordinal $\alpha < \omega^2$. Namely, a set A is α -c.e. if there are total computable functions φ_e and φ_k such that for every n , $A(n) = \lim_s \varphi_e(n, s)$, and for every s , if $\varphi_e(n, s) \neq \varphi_e(n, s+1)$ then $\varphi_k(n, s)$ codes an ordinal greater than $\varphi_k(n, s+1)$. We also require $\varphi_k(n, 0)$ to code an ordinal less than α , and be non-increasing in s . Clearly, for $\alpha = \omega$ the two definitions are equivalent. For example, an $(\omega+1)$ -c.e. set A is one which can be approximated by a computable function f , where the number of mind changes is bounded by a partial computable function ψ , where for every n , if $\psi(n) \uparrow$ then $f(n, s) = f(n, 0)$ for every s . We wish to compare this to an ω -c.e. set, where the mind change function is bounded by a *total* computable function, regardless

of whether there are any changes to the approximation of the set. This difference might not seem to be significant, but $(\omega + 1)$ -c.e. sets are in fact very different from ω -c.e. sets, and we shall explore this in Chapter 3.

Finally a function $F : \omega \mapsto \omega$ is α -c.e. if $F(n) = \lim_s \varphi_e(n, s)$ for φ_e, φ_k as above.

1.3 The Recursion Theorem

The Recursion Theorem is one of the most important results in the field, and is also known as the Kleene Fixed Point Theorem ([23]). Throughout this thesis, we will often be appealing to the Recursion Theorem, and thus we will state the theorem and explain its usage here. The Recursion Theorem states that for any total computable function f , there is some d (which can be found effectively in an index for f) such that $\varphi_{f(d)} = \varphi_d$. This is commonly applied in arguments in computability theory, for example, see Harris [21].

A common way of applying the Recursion Theorem in a construction is to use it to obtain the index of a computable function or c.e. set that we build *during* the construction, and we can in fact use the index of the object we build to define the object itself. This seemingly paradoxical assumption is actually sound, by an application of the Recursion Theorem. We usually describe a construction by specifying instructions on how to build the desired object, usually a computable function or a c.e. set. This construction uses a parameter, say c , which it assumes is the index of the object it constructs, say V . During the construction, we will often compare $W_c = V$ and proceed with the construction only if this appears to hold. Provided the instructions are uniform and effective in c , the result of this series of constructions is a computable function f such that $V = W_{f(c)}$. Now by applying the Recursion Theorem to f , we obtain a fixed point d such that $W_d = V$. Thus for at least one of the construction in the family, we assume correctly the index of the constructed object V . During a proof we will simply say “by the Recursion Theorem, we assume we are given the index of the object we build”, and we mean

that we apply the Recursion Theorem in the above-mentioned way.

Often we will not only construct a single object, but a family of c.e. sets. In order for us to assume that we are simultaneously given indices for all the c.e. sets we build, we shall need to apply the Recursion Theorem in a slightly different way. We again build a family of constructions, each with a different parameter c . In each construction of the family, say the construction with parameter c , we build an infinite collection of c.e. sets $\{V_y\}_{y \in \omega}$. This construction will assume that the sequence of indices are given by φ_c . That is, we assume $V_y = W_{\varphi_c(y)}$ for all y . Again, provided this description is uniform and effective in c and y , this series of constructions will in fact produce a computable function f such that in the c -th construction, we have $V_y = W_{f(c,y)}$ for all y . Applying the s-m-n and Recursion Theorems to f , we obtain a fixed point d such $\varphi_d(y) = f(d, y)$ for every y . Thus for at least one of the construction in the family, namely the one with parameter d , we are right about guessing that the index of the sequence $\{V_y\}$ is given by φ_d . When we use the Recursion Theorem this way, we will say “by the Recursion Theorem, we assume that we are given a sequence of indices for the objects we build”.

1.4 Organization of the thesis

The rest of this thesis is divided into three chapters. In Chapter 2 we study the degree structure of certain classes of arithmetical equivalence relations under computable reducibility. We also prove results about density and the existence of an effective enumeration of the class, and the existence of a universal element of the class. In Chapter 3 we prove that the analogue of Sacks jump inversion for the weak truth-table reducibility and the bounded jump fails. In Chapter 4 we give a characterization of the c.e. sets that are bounded high. In Chapter 5 we investigate different ways of effectivizing the property of being dominant.

Chapter 2

THE DEGREE STRUCTURE OF EQUIVALENCE RELATIONS UNDER COMPUTABLE REDUCIBILITY

2.1 Introduction

2.1.1 Motivations

An important theme in the study of mathematics is the classification of mathematical structures according to various different criteria. If we can identify when two different structures are “equivalent”, then we could reduce the study of all structures in a certain class to just examining the essential properties in each equivalence class of structures. The notion of dimension in algebra is an example of such a classification. A common tool in this endeavour is to define a *reduction*; this is a map which reduces a (possibly) more complicated problem to a simpler question.

In the context of Borel theory, a large body of work exists on classifying problems via the Borel reducibility. These reducibilities focus on classifying equivalence relations on an uncountable domain, for example, Friedman and Stanley [19] applied

this to study the isomorphism problem on the countable models of a theory.

A more recent direction of investigation has been directed towards studying equivalence relations on the domain ω . By restricting the domain down to a countable set, the usual tools of computability theory can be applied to supply meaningful definitions, concepts and proof techniques. The main reducibility used to calibrate the relative complexity of equivalence relations on ω is *computable reducibility*:

Definition 2.1.1. Let E and F be equivalence relations on ω . We say that E is computably reducible to F , denoted as $E \leq_c F$, if there exists a computable function $f : \omega \rightarrow \omega$ such that

$$\forall x, y \in \omega, \quad (x, y) \in E \Leftrightarrow (f(x), f(y)) \in F.$$

This definition is obviously closely related to Borel reducibility; in the Borel case one studies equivalence relations on 2^ω where the reductions are continuous or sometimes Borel. Indeed, Coskey, Hamkins and Miller [8] and later Miller and Ng [26] studied the countable analogues of several standard equivalence relations arising in Borel theory. This reducibility has gone by many different names in the literature, having been called m -reducibility in [3, 20, 4] and FF -reducibility in [16], in addition to a version on first-order theories which was called Turing-computable reducibility [5, 6]. Computable reducibility was also used in [22, 26]; we prefer this name since we wish to avoid confusion with the usual m -reducibility and Turing-reducibility between sets of numbers.

The computable reducibility was first introduced by Ershov in [15]. However his approach was much more indirect in that he phrased the notions inside the language of categories. Indeed his motivation was to study the theory of numberings, which is of fundamental interest in computable mathematics. In recent years there has been a sudden increase in interest in computable reducibility between equivalence relations on ω .

The objective of this chapter is to initiate a systematic study of the degrees induced by computable reducibility: The preordering $\leq_c$ induces an equivalence

relation on the class of all equivalence relations. The class of all R that are c -equivalent to S is called the c -degree of S , denoted as $\deg_c(S)$. In this chapter we shall be mainly interested in the α -c.e. equivalence relations for $\alpha \leq \omega$ and the Π_1^0 equivalence relations; the descriptive complexity of an equivalence relation is obviously evaluated when it is viewed as a subset of $\mathbb{N} \times \mathbb{N}$. A c -degree is α -c.e. if it contains an α -c.e. member.

The concept of *degrees* has been a central notion since the dawn of computability theory, and much effort has been directed towards the understanding of the algebraic structure of the degrees of different classes of sets. For instance, the c.e. m -degrees, 1-degrees, (weak) truth-table degrees and Turing degrees have all been well-studied. The original view that many people had of these structures were that they would be very simple and easy to classify. However, it turns out that these structures, particularly the c.e. Turing degrees, are much more complicated than originally thought. The study of the Turing degrees of larger classes of sets have also been extensively carried out, for instance, the d -c.e. Turing degrees and the Δ_2^0 Turing degrees have both turned out to be of interest, and shown to be structurally very different from the c.e. Turing degrees. Our aim here is not to dwell upon these results as these are all readily available in the literature. Rather, we wish to point out that this earlier work serves as the primary motivation for the questions we wish to investigate in this chapter.

One of the most basic questions to ask of a partial order is the existence of a greatest element. Given a class $\mathcal{C}$ of c -degrees, we call $\mathbf{u} \in \mathcal{C}$ *universal* if $\mathbf{u} \geq \mathbf{a}$ for every c -degree $\mathbf{a} \in \mathcal{C}$. That is, a universal degree is the greatest element of the class. We note here that the question of a *least* c -degree is uninteresting, for the computable equivalence relations generate an initial segment of c -degrees of length $\omega + 1$.

The question of universality has already been studied for some classes of c -degrees. For instance, Lachlan [25] studied certain kinds of universal c.e. equivalence relations (ceers), called precomplete ceers. We remark that ceers are also

known as positive equivalence relations, since they are closely related to positive numberings. Gao and Gerdes [20] and Andrews et al. [3] studied the c.e. c -degrees extensively, particularly in [3], where different types of universal c.e. c -degrees were identified and compared. The question of universality was then studied further in [22], where the authors showed the existence of a universal Π_1^0 c -degree, and that no universal degree exists for the class of Δ_2^0 c -degrees.

The notion of a universal element of a class is closely related to the existence of a computable enumeration of the members of that class. If one can effectively list out all elements of the class, then taking the disjoint union of the members of the effective list will usually produce a universal element of the class. For instance, the halting problem, which has universal m -degree amongst the c.e. sets, can be viewed as being the disjoint union of all c.e. sets (we can even effectively list all c.e. sets without repetition, a famous result of Friedberg [18]). Indeed, the existence of a universal c.e. c -degree can be easily proved in the same way; first, notice that the transitive closure of any c.e. binary relation is again c.e., and hence we can easily produce an effective list containing all c.e. equivalence relations. Taking the disjoint union of this list (with no relations between different members of the list) produces a universal c.e. equivalence relation.

The same trick works to produce a universal Σ_n^0 c -degree for each $n > 0$. Unfortunately, an effective listing of all Π_1^0 equivalence relations is less obvious. Indeed, the authors of [22] proved the existence of a universal Π_1^0 c -degree in an indirect way. For α -c.e. equivalence relations, where $1 < \alpha \leq \omega$, the question of universality was not known before this work. In Section 2.2, we answer the question of universality in these classes and also investigate what kinds of effective enumerations are possible for these classes. The results are summarized in the table on page 12.

Another fundamental question one can ask about a partial order is whether the structure is dense. Interest in this particular property is partially motivated by results in the c.e. and Δ_2^0 Turing degrees, where similar questions about density have led to a rich collection of knowledge and to new techniques being invented.

Related questions such as those on lattice embeddings have also enriched the field. We will investigate the question of density in the classes of c -degrees we are interested in. We prove that in each of these classes (α -c.e. for $1 \leq \alpha \leq \omega$, Π_1^0 and Δ_2^0 c -degrees), downward density fails but upward density holds. We believe that this is due to the fact that it is much easier to *code* given information into a relation we build, compared to *permitting* below a given equivalence relation.

In Section 2.3 we consider the problem of downward density in the structure of c.e. c -degrees. It is not hard to see that the map $A \mapsto \{(x, y) \mid x = y \text{ or } x, y \in A\}$ is an embedding (of partial orders) of the c.e. 1-degrees into the c.e. c -degrees. We call a c -degree *set-induced* if it contains a relation in the range of this embedding. An old result of Lachlan implies the existence of a set-induced c -degree for which downward density fails. These results and terminologies will be further elaborated upon in Section 2.3. We construct two different counter-examples to the downward density of the c.e. c -degrees, one which is above $\deg_c(\text{id})$, and another which is not above $\deg_c(\text{id})$. Both our counter-examples are, of course, not set-induced.

In Section 2.4 we show that downward density fails for the Π_1^0 c -degrees, and generalize this theorem by constructing an infinite sequence of Π_1^0 c -degrees $\deg_c(\text{id}) < \deg_c(A_2) < \deg_c(A_3) < \dots$, each of which is a strong minimal cover over the previous. All our counter-examples are not set-induced.

In Section 2.5 we investigate upward density in the c -degrees. We show that in contrast to downward density, upward density holds in all the classes of interest. We do this by constructing a pair of incomparable Π_1^0 c -degrees above any given non-universal Π_1^0 c -degree.

Degree class	Existence of a universal element	Existence of a computable enumeration	Downward density	Upward density
Σ_n^0 , $n \geq 1$	✓ (Folklore)	✓ (Folklore)	× (Cor 2.3.12)	✓ (Cor 2.5.5)
n -c.e., $n > 1$	✓ (Cor 2.2.2)	✓ (Thm 2.2.1)	× (Cor 2.3.12)	✓ (Cor 2.5.5)
ω -c.e.	✓ (Cor 2.2.6)	– of all relations × (Folklore) – of all degrees ✓ (Thm 2.2.5)	× (Cor 2.3.12)	✓ (Cor 2.5.5)
Π_1^0	✓ (Thm 3.3 of [22])	– of all relations × (Prop 2.2.10) – of all degree uppercones ✓ (Thm 2.2.8)	× (Thm 2.4.1)	✓ (Thm 2.5.1)
Δ_2^0	× (Thm 3.7 of [22])	× (no universal element)	× (Cor 2.3.12)	✓ (Thm 3.7 of [22])

2.1.2 Preliminaries

An equivalence relation S always refers to a set of ordered pairs, i.e. $S \subseteq \mathbb{N}^2$, and the domain of each relation is always $\mathbb{N}$.

If S and T are equivalence relations, denote $S \sqcup T$ by the equivalence relation $\{(2x, 2y) \mid (x, y) \in S\} \cup \{(2x + 1, 2y + 1) \mid (x, y) \in T\}$. Notice that $\deg_c(S \sqcup T)$ is not obviously the least upper bound of the pair $\deg_c(S)$ and $\deg_c(T)$, since if f witnesses $S \leq_c X$ and g witnesses $T \leq_c X$ then $\text{rng}(f)$ and $\text{rng}(g)$ might contain elements that are X -related; we will not know how to identify these elements inside $S \sqcup T$.

Given a binary relation S and an effective enumeration $\{S_s\}_{s \in \omega}$ of S , define $S \upharpoonright t := \{(x, y) \in S \mid x, y < t\}$ and $S_s \upharpoonright t := \{(x, y) \in S_s \mid x, y < t\}$. We sometimes append $[s]$ to an expression to denote the value of the expression evaluated at step or stage s . A number x is said to be *isolated* with respect to an equivalence relation T if $T(x, y) = 0$ for every $y \neq x$.

We fix notations for several standard equivalence relations. Let $\equiv_n$ be the computable equivalence relation with exactly n many distinct equivalence classes. (Note that not every equivalence relation with finitely many classes is computable, for example, take the equivalence relation with two classes $\{A, \omega - A\}$ for any non-computable set A). Also fix id to be the computable equivalence relation with infinitely many classes. Of course we have

$$\equiv_1 <_c \equiv_2 <_c \cdots <_c \text{id},$$

and every computable equivalence relation is of the same degree as one of these. Furthermore, this chain of degrees form an initial segment of the global c -degrees - since if R is reducible to any degree in the chain then R is computable.

In order for us to work with the various classes of equivalence relations (ceers, Π_1^0 and n -c.e.), we wish to fix an effective enumeration of all members of the class. This is unfortunately not always possible, so we fix a computable enumeration $\{S_e\}_{e \in \omega}$ of all reflexive and symmetric (c.e., Π_1^0 or n -c.e., depending on the context) binary relations. In fact, we will fix the approximation to $\{S_e\}_{e \in \omega}$ such that for all e, x, y, t , $S_e(x, x)[t] = 1$ and $S_e(x, y)[t] = S_e(y, x)[t]$.

Notice that the reflexive and symmetric closure of a binary relation has the same descriptive complexity as the relation, while the transitive closure of a binary

relation S is Σ_1^0 relative to S . Thus, the issue of whether $\{S_e\}_{e \in \omega}$ can be turned into a computable enumeration of all members of the class comes down to the difficulty of recognizing transitivity in the class; for ceers this is trivial (the transitive closure of a c.e. binary relation is still c.e.), while for Π_1^0 and n -c.e. equivalence relations this question will be addressed in the respective sections. The existence of an enumeration $\{S_e\}_{e \in \omega}$ for ω -c.e. equivalence relations has to be defined in the right way, and we will discuss this in Section 2.2.2.

2.1.3 Organization of the chapter

Chapter 2 is organized as follows. We shall be primarily interested in the class of Π_1^0 equivalence relations, the class of ω -c.e. equivalence relations and the class of n -c.e. equivalence relations for each $n \geq 1$.

In Section 2.2 we consider the problem of producing a computable enumeration of each class as well as the existence of a universal equivalence relation of the class. In Theorem 2.2.1 we provide a computable enumeration of all n -c.e. equivalence relations and deduce the existence of a universal n -c.e. equivalence relation for each $n \geq 1$. In Theorem 2.2.5 we provide a computable enumeration of ω -c.e. equivalence relations in which every degree is represented, and deduce the existence of a universal ω -c.e. equivalence relation. Then in Theorem 2.2.8 we provide a computable enumeration of Π_1^0 equivalence relations in which the upper cone of every degree is represented, and deduce the existence of a universal Π_1^0 equivalence relation. Note that the existence of a universal Π_1^0 equivalence relation is proved in [22], while a universal α -c.e. equivalence relation (for $1 < \alpha \leq \omega$) was not known before. We also show in Proposition 2.2.10 that a computable enumeration of all Π_1^0 equivalence relations is impossible.

In Section 2.3 we consider the problem of downward density in the structure of the c -degrees of c.e. equivalence relations. In Theorem 2.3.9 we construct a c.e. c -degree which is not set-induced, minimal modulo the computable c -degrees and not above $\deg_c(\text{id})$. In Theorem 2.3.11 we construct another c.e. c -degree which is

not set-induced and minimal, and this time above $\deg_c(\text{id})$.

In Section 2.4 we show that downward density fails for the Π_1^0 c -degrees. Next we generalize this theorem by constructing an infinite sequence of Π_1^0 c -degrees $\deg_c(\text{id}) < \deg_c(A_2) < \deg_c(A_3) < \dots$, each of which is a strong minimal cover over the previous. All our counter-examples are not set-induced.

Finally in Section 2.5 we prove that in contrast to downward density, upward density holds in all the classes of interest.

2.2 Enumerations of a class and universality

2.2.1 n -c.e. equivalence relations

In this section we address the question of the existence of a computable enumeration of all n -c.e. equivalence relations, and the existence of a universal n -c.e. equivalence relation. We fix some $n \in \omega$, $n \geq 1$.

Theorem 2.2.1. *There is a computable enumeration of all n -c.e. equivalence relations. More specifically, there exists a computable function $f(e, x, y, s)$ such that:*

- For all e, x, y , $\lim_{s \rightarrow \infty} f(e, x, y, s) = E_e(x, y)$.
- For all e , E_e is a n -c.e. equivalence relation.
- For each n -c.e. equivalence relation S there exists some e such that $S = E_e^1$.
- For all e, x, y , $f(e, x, y, 0) = 0$ and $\#\{s \mid f(e, x, y, s) \neq f(e, x, y, s+1)\} \leq n$.

Proof. As mentioned in Section 2.1.2 we fix $\{S_e\}_{e \in \omega}$ to be a computable enumeration of all n -c.e. binary relations which are reflexive and symmetric. For each e , we construct (an approximation to) E_e uniformly in e .

At stage $v = 0$, define $u_0 := 0$ and $E_{e,0} := \{(x, x) : x \in \omega\}$, and do nothing else. At stage $v > 0$, let t be largest such that $u_t \downarrow$, and assume that $E_{e,v-1}$ has also

¹Note that S and E_e are actually equal instead of merely $\equiv_e$. This is the best kind of enumeration one can hope for.

been defined (note that t always exists). Check if $S_{e,v} \upharpoonright t+1$ is transitive. If it is, define $u_{t+1} = v$ and $E_{e,v} = S_{e,v} \upharpoonright t+1$. If not, define $E_{e,v} := E_{e,v-1}$ and leave $u_{t+1} \uparrow$.

We now verify that E_e has the properties we want.

E_e is n -c.e.: For each x, y, t , we have that $E_{e,v-1}(x, y) \neq E_{e,v}(x, y)$ iff $v = u_{t+1}$ and $(S_{e,u_t} \upharpoonright t)(x, y) \neq (S_{e,u_{t+1}} \upharpoonright t+1)(x, y)$. In particular, if v_0 is the least (if it exists) such that $E_{e,v_0-1}(x, y) \neq E_{e,v_0}(x, y)$, then $E_{e,v_0}(x, y) = (S_{e,u_{t+1}} \upharpoonright t+1)(x, y) = 1$ where $v_0 = u_{t+1}$, and hence $x, y < t+1$. This means that after stage v_0 , all further changes in $E_{e,v}(x, y)$ must correspond to a change in $S_e(x, y)$. Therefore, $E_e(x, y)$ changes no more often than $S_e(x, y)$.

E_e is an equivalence relation: E_e is obviously reflexive since $E_{e,0}$ is. Now note that for every x, y , if $E_e(x, y) = 1$ then $S_e(x, y) = 1$ and $u_t \downarrow$ for some $t > x, y$. Therefore E_e is symmetric. Clearly $E_{e,v}$ is transitive for every v , hence E_e is transitive.

If S_e is transitive then $S_e = E_e$: If S_e is transitive then $u_t \downarrow$ for every t , and so E_e is set to copy $S_e \upharpoonright t$ for larger and larger t . $\square$

The existence of a computable enumeration of all n -c.e. equivalence relation immediately yields:

Corollary 2.2.2. *There exists a universal n -c.e. equivalence relation for any fixed $n \geq 1$.*

However, if one considers varying n over all of ω , then there will not be a universal element of the class:

Proposition 2.2.3. *The collection*

$$\{S : S \text{ is an } n\text{-c.e. equivalence relation for some } n \in \omega\}$$

has no universal element.

Proof. Given an n -c.e. equivalence relation S , the obvious thing to do is to consider the following equivalence relation (generated by) E :

$$E := \{ \langle \langle e, 0 \rangle, \langle e, 1 \rangle \rangle \mid \varphi_e(\langle e, 0 \rangle) \downarrow = x \wedge \varphi_e(\langle e, 1 \rangle) \downarrow = y \wedge S(\langle x, y \rangle) = 0 \}.$$

By the definition of E , we have for each e such that φ_e is total, $E(\langle e, 0 \rangle, \langle e, 1 \rangle) \neq S(\varphi_e(\langle e, 0 \rangle), \varphi_e(\langle e, 1 \rangle))$, which implies $E \not\leq_c S$ via φ_e .

Now for each e , $E(\langle e, 0 \rangle, \langle e, 1 \rangle)$ initially starts off taking value 0, before φ_e converges on the two inputs. Once $\varphi_e(\langle e, 0 \rangle)$ and $\varphi_e(\langle e, 1 \rangle)$ both converge, we might see $S(x, y) = 0$ where we now have to let $E(\langle e, 0 \rangle, \langle e, 1 \rangle) = 1$, and subsequently we follow the changes in $S(x, y)$. Hence E changes at most one more time than S , and so E is $n + 1$ -c.e. $\square$

2.2.2 ω -c.e. equivalence relations

Now we turn to ω -c.e. equivalence relations. In particular, we wish to investigate the question of whether there exists a universal ω -c.e. equivalence relation, and what kind of enumerations are possible of this class. The best kind of enumeration one could ask for is an effective listing $\{\langle S_e, f_e \rangle\}_{e \in \omega}$ of all ω -c.e. equivalence relations S_e along with a computable bound f_e for the number of changes in S_e . This is of course impossible, otherwise we can diagonalize out of the class. Therefore, we will fix an enumeration $\{\langle S_e, h_e \rangle\}_{e \in \omega}$ in the following sense:

Lemma 2.2.4. *There exists an effective listing $\{\langle S_e, h_e \rangle\}_{e \in \omega}$ of all ω -c.e. equivalence relations S_e such that h_e is partial computable. Here every S_e is an ω -c.e. equivalence relation and every ω -c.e. equivalence relation S is equal to S_e for some e such that h_e is total and bounds the number of mind changes of S_e .*

Proof. S_e is taken to be a slowed down version of the e^{th} possible ω -c.e. binary relation T_e . We follow the proof of Theorem 2.2.1. We start with $S_e(x, y)[0] = 0$ for all distinct x and y . For some t , if we discover that $\varphi_e(x, y)$ has converged for all $x, y < t$, we proceed to wait for convergence of all $x, y < t + 1$. Before that happens, we keep $S_e(x, y) = 0$ for all (x, y) such that $x > t$ or $y > t$, and update

S_e to copy changes in $T_e \upharpoonright t$ only when T looks transitive. When $\varphi(x, y)$ converges for all $x, y < t + 1$, if ever, and when $T_e \upharpoonright t + 1$ again looks transitive, we extend the domain of h_e and then set S_e to now copy $T_e \upharpoonright t + 1$, again updating only if $T_e \upharpoonright t + 1$ looks transitive. This ensures that S_e is always an equivalence relation, and the domain of f_e is $\{(x, y) \mid x, y < t\}$ for some $t \leq \omega$, and where $S_e = T_e$ and $h(e)$ is total in the event that T_e turns out to be ω -c.e. and transitive.

Note that if h_e is partial, then $S_e - \{(x, x) : x \in \omega\}$ is finite and certainly ω -c.e. $\square$

Unlike Theorem 2.2.1 for n -c.e. equivalence relations, Lemma 2.2.4 is not sufficient to deduce the existence of a universal ω -c.e. equivalence relation, because the effective disjoint union of all S_e is not ω -c.e. However, recall that every ω -c.e. m -degree (of subsets of ω) contains an ω -c.e. set with an approximation that changes at most identity bounded many times. If we could prove, for example, the analogous statement that there exists a computable enumeration $\{E_e\}_{e \in \omega}$ of equivalence relations with identity bounded changes, and which contains an equivalence relation of every ω -c.e. degree, then the disjoint union of E_e will provide a universal element of the class. This is provided by the following theorem:

Theorem 2.2.5. *For each e , there exists (uniformly in e) an approximation to an equivalence relation E_e such that:*

- *For every x, y , the number of changes in the approximation of $E_e(x, y)[s]$ is bounded by $\max\{x, y\}$.*
- *If h_e is total then $S_e \equiv_c E_e$.*

Proof. Fix e . Let $\hat{h}(y) = \max_{x, x' \leq y} h_e(x, x')$; then $\hat{h}$ is a partial computable function whose domain is an initial segment of ω . Without loss of generality, we assume that $\hat{h}$ is strictly increasing on its domain. Define $h^*(x) = \max\{z - 1, 0\}$ where z

is the least such that $\hat{h}(z) \geq x$. Now define

$$E_e(x, y) = \begin{cases} S_e(h^*(x), h^*(y)), & \text{if } h^*(x), h^*(y) \text{ both exist,} \\ 1, & \text{if } x = y, \\ 0, & \text{otherwise.} \end{cases}$$

Now $E_e(x, y)$ has the following obvious approximation for $x \neq y$: Begin with $E_e(x, y) = 0$ and wait for both $h^*(x)$ and $h^*(y)$ to become defined. When this happens, copy $S_e(h^*(x), h^*(y))$. Thus, the number of changes in the approximation of $E_e(x, y)$ is at most that of $S_e(h^*(x), h^*(y))$, which is in turn bounded by $h_e(h^*(x), h^*(y))$. Suppose $x < y$ (the other case $y < x$ follows symmetrically). Therefore $h^*(x) \leq h^*(y)$ by the definition of h^* . Let z_y be least such that $\hat{h}(z_y) \geq y$ (this is defined as $h^*(y) \downarrow$). If $h^*(y) = 0$ then $h^*(x) = 0$ and so $E_e(x, y)$ changes at most once. So suppose that $h^*(y) > 0$, and in particular, $h^*(y) = z_y - 1$. Since $h^*(x), h^*(y) \leq z_y - 1$, this means that $h_e(h^*(x), h^*(y)) \leq \hat{h}(z_y - 1)$. However $\hat{h}(z_y - 1) < y$ by the minimality of z_y , and so $S_e(h^*(x), h^*(y))$ changes no more than $\max\{x, y\}$ many times. This verifies the first property claimed in the theorem.

It is also clear that E_e is an equivalence relation, because S_e is. Finally, suppose that h_e is total. We wish to see that $S_e \equiv_c E_e$. Now $E_e \leq_c S_e$ is given by h^* , which must be total since h_e is total and $\hat{h}$ is strictly increasing. For the direction $S_e \leq_c E_e$, we simply take the map $z \mapsto \hat{h}(z) + 1$. Notice that $h^*(\hat{h}(z) + 1) = z$. $\square$

Corollary 2.2.6. *There exists a universal ω -c.e. equivalence relation.*

Proof. The effective disjoint union $\sqcup_{e \in \omega} E_e$ from Theorem 2.2.5 is clearly an ω -c.e. equivalence relation. Every ω -c.e. equivalence relation is equal to S_e for some e where h_e is total, which has the same degree as E_e . $\square$

2.2.3 Π_1^0 equivalence relations

We recall Theorem 3.3 of [22], where it is shown that there is a universal Π_1^0 equivalence relation:

Theorem 2.2.7 ([22, Theorem 3.3]). *There exists a universal Π_1^0 equivalence relation.*

The proof in [22] does not mention enumerations of the class, nor does it explicitly show how to uniformly transform a Π_1^0 binary relation S_e into a transitive equivalence relation. For the sake of completeness, we will give a slightly different presentation in this section.

First, we fix $\{S_e\}_{e \in \omega}$ as in Section 2.1.2. We first show that it is possible to construct an effective enumeration $\{E_e\}_{e \in \omega}$ of Π_1^0 equivalence relations in which the upper cone of every Π_1^0 degree is represented:

Theorem 2.2.8. *There exists an effective Π_1^0 enumeration of a sequence $\{E_e\}_{e \in \omega}$ such that:*

- *For every e , E_e is a Π_1^0 equivalence relation.*
- *For any Π_1^0 equivalence relation S , there exists some e such that $S \leq_c E_e$;*

Proof. Fix e , and we shall build a Π_1^0 enumeration of E_e uniformly in e such that if S_e is transitive, then $S_e \leq_c E_e$. The issue that we *must* keep E_e transitive, while S_e might not be, and we have to do it in a way where we can extend our reduction f_e witnessing $S_e \leq_c E_e$ whenever S_e looks transitive on a larger fragment.

Define $f_e(x)$ inductively by setting $f_e(0) = 0$, and $f_e(x+1)$ to be the least stage larger than $f_e(x)$ such that $S_e[f_e(x+1)] \upharpoonright x+2$ looks like an equivalence relation. Then f_e is partial computable, and is total if and only if S_e is an equivalence relation. Now define $E_e(x, y)$ (for $x \neq y$) by the following. If $x \notin \text{rng}(f_e)$ or $y \notin \text{rng}(f_e)$ then $E_e(x, y) = 0$, otherwise let a, b be such that $f_e(a) = x$ and $f_e(b) = y$, and define $E_e(x, y) = 0$ if and only if $S_e(a, b)[f_e(z)] = 0$ for some z in $\text{dom}(f_e)$. Obviously if $x = y$ we set $E_e(x, y) = 1$.

There is an obvious Π_1^0 enumeration of E_e : First, note that $\text{rng}(f_e)$ is computable (uniformly in e), since $s \in \text{rng}(f_e)$ if and only if this has happened before querying $S_e[s+1]$ (this is true even if $\text{rng}(f_e)$ is finite). Furthermore the condition “ $S_e(a, b)[f_e(z)] = 0$ for some z in $\text{dom}(f_e)$ ” is c.e. Hence E_e is Π_1^0 , with an

enumeration effectively found in e .

It is obvious that E_e is reflexive and symmetric (since S_e is at every stage). For transitivity, suppose $E(x, y) = E(y, w) = 1$ and $E(x, w) = 0$ for distinct x, y, w . Then clearly there must exist a, b, c such that $f_e(a) = x$, $f_e(b) = y$ and $f_e(c) = w$. Also fix $z \in \text{dom}(f_e)$ such that $S_e(a, c)[f_e(z)] = 0$ triggering us to define $E_e(x, w) = 0$. Now observe that $S_e[f_e(\max\{a, b, c, z\})] \upharpoonright (\max\{a, b, c, z\} + 1)$ cannot be transitive, contradicting the properties of f_e .

Finally suppose that S_e is an equivalence relation, hence f_e is total. Then f_e witnesses that $S_e \leq_c E_e$. $\square$

We deduce immediately [22, Theorem 3.3]:

Corollary 2.2.9. *There exists a universal Π_1^0 equivalence relation.*

We now turn to the interesting question of whether there exists an effective enumeration of all Π_1^0 equivalence relations, or even an enumeration representing all Π_1^0 degrees. First of all, it is easy to see that an enumeration of all Π_1^0 equivalence relations is impossible:

Proposition 2.2.10. *There does not exist an effective sequence $\{E_e\}_{e \in \omega}$ of Π_1^0 equivalence relations such that for every Π_1^0 equivalence relation S , there is some e such that $S = E_e$. (We can even replace “ $S = E_e$ ” by “ $S \leq_c E_e$ via a primitive recursive function”).*

Note that Proposition 2.2.10 does not contradict Theorem 2.2.8. In fact, it shows that the proof of Theorem 2.2.8 is sharp in the sense that the reductions f_e constructed there cannot all be total.

Proof of Proposition 2.2.10. Fix the sequence $\{E_e\}_{e \in \omega}$ as above, and we shall construct a Π_1^0 equivalence relation S such that $S \neq E_e$ for every e . In fact, we shall ensure that for every e , p_e does not witness $S \leq_c E_e$, where p_e is the e^{th} primitive recursive function.

At the beginning we define $S(\langle e, i \rangle, \langle e', j \rangle)[0] = 0$ for all e, e', i, j such that $e \neq e'$, and $S(\langle e, i \rangle, \langle e, j \rangle)[0] = 1$ for all e, i, j . For each e , we do the following

uniformly. We begin our strategy for e by first setting $S(\langle e, 0 \rangle, \langle e, 1 \rangle) = 0$, and wait for a stage s such that $E_e(p_e(\langle e, 0 \rangle), p_e(\langle e, 1 \rangle)) = 0$. For every stage t where $E_e(p_e(\langle e, 0 \rangle), p_e(\langle e, 1 \rangle))[t]$ is still $= 1$, we set $S(\langle e, t+2 \rangle, \langle e, i \rangle) = 0$ for every $i \neq t+2$ (that is, declare $\langle e, t+2 \rangle$ to be an isolated point).

Let s be the first stage such that $E_e(p_e(\langle e, 0 \rangle), p_e(\langle e, 1 \rangle))[s] = 0$ (note that s may not exist). At this point, we have made $\langle e, i \rangle$ an isolated point for all $2 \leq i < s+2$. However $S(\langle e, s+2 \rangle, \langle e, i \rangle) = 1$ for all i . We wait for either $E_e(p_e(\langle e, 0 \rangle), p_e(\langle e, s+2 \rangle)) = 0$ or $E_e(p_e(\langle e, 1 \rangle), p_e(\langle e, s+2 \rangle)) = 0$; at least one of these must hold because E_e is transitive. Suppose $E_e(p_e(\langle e, 0 \rangle), p_e(\langle e, s+2 \rangle)) = 0$. Then we declare $\langle e, 1 \rangle$ to be isolated and do nothing else for e ; otherwise declare $\langle e, 0 \rangle$ to be isolated.

We now verify that S has the properties we want. Firstly, it is clear that the above describes a Π_1^0 approximation to S , as E_e is transitive and p_e is total. Now if s is never found by the strategy for e then the e^{th} column of S will consist only of isolated points. If s is found by the strategy then the e^{th} column $S^{[e]} = \{\langle e, i \rangle \mid i \in \omega\}$ will consist of $s+1$ many isolated points, while the rest of the points belong to a single equivalence class. In either case, $S^{[e]}$ is an equivalence relation. Hence S is an equivalence relation. Finally, it is straightforward to check that for every e , p_e does not witness $S \leq_c E_e$. $\square$

Given Proposition 2.2.10 we can ask if there is an enumeration of Π_1^0 equivalence relations in which every degree is represented, like in the case of ω -c.e. equivalence relations. Part of the difficulty of obtaining such an enumeration is that the reductions from S_e to E_e must be allowed to be partial. We leave open the question:

Question 2.2.11. *Is there an effective enumeration of Π_1^0 equivalence relations $\{E_e\}_{e \in \omega}$ such that if S is a Π_1^0 equivalence relation, then there is some e such that $S \equiv_c E_e$?*

2.3 Downward density and c.e. equivalence relations

We now investigate the structure of the c.e. c -degrees, in particular, the problem of downward density. Unlike the Π_1^0 c -degrees (see Section 2.4), it is not the case that every c.e. c -degree is comparable with $\deg_c(\text{id})$. Given any c.e. set $A \subseteq \omega$, define the ceer $R_A = \{(x, y) \mid x = y \text{ or } x, y \in A\}$. Then observe that $\text{id} \leq_c R_A$ if and only if A is not simple. Therefore, even though the computable c -degrees $\deg_c(\equiv_1) < \deg_c(\equiv_2) < \dots < \deg_c(\text{id})$ is an initial segment of the c.e. c -degrees, we now get non-trivial c.e. c -degrees which are incomparable with this initial segment.

This situation is very similar to that of the c.e. 1-degrees, where it had been observed that the “bottom” degree, $\mathbf{0}_1 = \deg_1(\emptyset \oplus \omega)$, is not really the least element of the c.e. 1-degrees: The simple c.e. sets all have 1-degree incomparable with $\mathbf{0}_1$. Nevertheless, much of the earlier studies of the c.e. 1-degrees have been focused on the interval $[\mathbf{0}_1, \deg_1(\emptyset')]$. For example, Lachlan [25] showed that every finite initial segment in this interval is a distributive lattice, and hence the first-order theory of the c.e. 1-degrees is undecidable. Young [31] showed that this interval is neither an upper semilattice nor a lower semilattice. Part of the reason for this is due to the fact that results about m -degrees of c.e. sets can often be transferred to the interval $[\mathbf{0}_1, \deg_1(\emptyset')]$, as the non-simplicity of the c.e. sets in this interval often allow m -reductions to be turned into 1-reductions.

It was observed in [3] that the interval $[\mathbf{0}_1, \deg_1(\emptyset')]$ embeds into the c.e. c -degrees; in fact, the interval $[\mathbf{0}_1, \deg_1(\emptyset')]$ of c.e. 1-degrees was shown to be order-isomorphic with the interval $[\deg_c(\text{id}), \deg_c(R_{\emptyset'})]$ of c -degrees. Note that $\deg_c(R_{\emptyset'})$ is *not* the top c.e. c -degree; in fact, $R_{\emptyset'}$ is far from being universal amongst the c.e. c -degrees.

Definition 2.3.1. We call a c -degree $\mathbf{a}$ *set-induced* if there exists an infinite and co-infinite set $A \subseteq \omega$ such that $R_A \in \mathbf{a}$.

It was shown in [3] that every c -degree in the interval $[\deg_c(\text{id}), \deg_c(R_{\emptyset'})]$ is set-induced. In fact, the same proof shows a little more; for completeness, we

reproduce the argument here:

Lemma 2.3.2. *Suppose R is a non-computable equivalence relation and $A \subseteq \omega$ is any set such that $R \leq_c R_A$. Then there exists a set $B \subseteq \omega$ such that $R \equiv_c R_B$.*

Proof. Fix a computable function f witnessing $R \leq_c R_A$. Since R is not computable, $\text{rng}(f)$ is an infinite c.e. set. We let $\{d_n\}_{n \in \omega}$ be a one-one computable enumeration of $\text{rng}(f)$. Let $B = \{n \in \omega \mid d_n \in A\}$. Notice that this gives $B \leq_1 A$, as expected.

To see that $R \leq_c R_B$, use the reduction $(d_n)^{-1} \circ f$. For the other direction $R_B \leq_c R$, we map $n \mapsto$ the least x such that $f(x) = d_n$. $\square$

Lemma 2.3.2 shows not only that every c -degree in the interval $[\deg_c(\text{id}), \deg_c(R_\emptyset)]$ is set-induced, but also:

Corollary 2.3.3. *A non-computable c.e. c -degree $\mathbf{a}$ is set-induced if and only if $\mathbf{a} \leq \deg_c(R_\emptyset)$.*

Before examining the question of downward density for c -degrees, we mention one related result about c.e. 1-degrees. Recall the following definition:

Definition 2.3.4. If $\mathbf{a} < \mathbf{b}$ are two elements of a partial order, then $\mathbf{b}$ is said to be a *strong minimal cover* of $\mathbf{a}$ if the following is true of the partial order: $\forall \mathbf{c} \leq \mathbf{b} (\mathbf{c} = \mathbf{b} \vee \mathbf{c} \leq \mathbf{a})$.

Lachlan [25] proved the existence of a minimal element of the interval $[\mathbf{0}_1, \deg_1(\emptyset')]$. In fact, a closer examination of the proof reveals that there exists a c.e. 1-degree which is a strong minimal cover of $\mathbf{0}_1$ within the structure of all c.e. 1-degrees:

Theorem 2.3.5 ([25, Lachlan]). *There exists a non-simple and non-computable c.e. set A such that for every set $B \leq_1 A$, we have $A \equiv_1 B$ or B is computable.*

We now turn to the matter at hand in this section. The statement of downward density for c -degrees says that given any non-computable c.e. c -degree $\mathbf{a}$, there exists a non-computable (c.e.) c -degree $\mathbf{b} < \mathbf{a}$. We show that this fails by constructing two counter-examples: Theorem 2.3.9 provides a “minimal” c.e. c -degree

$\mathbf{a} \not\leq \deg_c(\text{id})$, while Theorem 2.3.11 provides a strong minimal cover $\mathbf{a}$ of $\deg_c(\text{id})$. However the existence of a strong minimal cover of $\deg_c(\text{id})$ in the c.e. c -degrees can already be deduced from Theorem 2.3.5:

Proposition 2.3.6. *There is a c.e. set-induced c -degree $\mathbf{a} > \deg_c(\text{id})$ such that $\mathbf{a}$ is a strong minimal cover of $\deg_c(\text{id})$.*

Proof. Consider $R_A >_c \text{id}$ where A is from Theorem 2.3.5. Now if $R \leq_c R_A$ then by Lemma 2.3.2 either R is computable, or else $R \equiv_c R_B$ for some $B \leq_1 A$. As B is not computable we have $B \geq_1 A$, which means that $R \geq_c R_A$. $\square$

Thus our counter-examples to downward density in Theorems 2.3.9 and 2.3.11 will be c.e. c -degrees that are not set-induced. It shall be convenient to isolate properties that imply that a certain c -degree is not set-induced. We identify two conditions sufficient for a degree to be not set-induced.

The first and probably the easiest way to produce an equivalence relation R that does not have set-induced c -degree is to make every R -class finite:

Lemma 2.3.7. *If R is any non-computable equivalence relation where every class is finite, then $\deg_c(R)$ is incomparable with every non-computable set-induced c -degree.*

Proof. If $R_A \leq_c R$ via a computable function f , then $f(A)$ is a finite set and hence A is computable. On the other hand if $R \leq_c R_A$ and R is non-computable, then by Lemma 2.3.2 there is a B such that $R \equiv_c R_B$. Since $R_B \leq_c R$, B is computable, a contradiction. $\square$

Unfortunately we will not use Lemma 2.3.7 to ensure that the ceers constructed in Theorems 2.3.9 and 2.3.11 are not of set-induced degree. This is because it is generally a messy task to construct non-trivial c.e. equivalence relations where every class is finite. However, we will be able to easily do this for Π_1^0 equivalence relations. Therefore Lemma 2.3.7 will be used in the proof of Theorems 2.4.1, 2.4.2 and 2.4.10 in the next section.

To ensure that the c.e. equivalence relations constructed in Theorems 2.3.9 and 2.3.11 are not of set-induced degree, we will introduce a second sufficient condition:

Lemma 2.3.8. *Let R be any non-computable c.e. equivalence relation with the property that given any c.e. set W , if infinitely many R -classes contain an element of W , then every R -class contains an element of W . Then R is a counter-example to downward density. If additionally each R -class contains at least two elements of each W which intersects infinitely many R -classes, then there is no set-induced degree below $\deg_c(R)$.*

Proof. Fix R as above. Let $S \leq_c R$ via a computable function f . If $\text{rng}(f)$ lies in only finitely many R -classes then as R is c.e., S is computable. Otherwise each R -class must contain an element of $\text{rng}(f)$. Since R is c.e., this obviously allows us to show that $R \leq_c S$.

Now suppose $R_A \leq_c R$ via f for some co-infinite set A . Then $\text{rng}(f)$ must intersect infinitely many different R -classes. Hence each R -class will contain at least two elements of $\text{rng}(f)$. Since R is not computable, there are at least two R -classes. This means that $f(x) R f(y)$ for some pair $x, y \notin A$ and $x \neq y$, a contradiction. $\square$

In Theorem 2.3.9 we will use Lemma 2.3.8 to ensure that the degree of the constructed ceer is not above a set-induced degree.

Theorem 2.3.9. *There is a non-computable c.e. c-degree $\mathbf{a} \not\leq \deg_c(\text{id})$ such that $\mathbf{a}$ is not set-induced and for every $\mathbf{b} \leq \mathbf{a}$, either $\mathbf{b}$ is computable or $\mathbf{b} \geq \mathbf{a}$.*

Proof. We are going to build a c.e. equivalence relation A and ensure the following requirements:

R_{2e} : $A \neq \varphi_e$.

R_{2e+1} : W_e intersects infinitely many A -classes $\Rightarrow$ every A -class contains at least two elements of W_e .

It should be clear that these requirements ensure the properties needed for A . The R_{2e} requirements ensure that A is not computable. Lemma 2.3.8 and the R_{2e+1} requirements together ensure that $\deg_c(A)$ is a counter-example to downward density and does not bound a set-induced degree. Since $\text{id} \equiv_c R_{\omega \oplus \emptyset}$, it follows that $\deg_c(A) \not\leq \deg_c(\text{id})$.

During the construction, the classes of A are ordered according to the magnitude of the least element of the class. The n -th class of A_s is denoted by $C_{n,s}$. Since A is c.e., the basic operation we will perform at each step s is to *collapse* C_n and C_m for certain classes C_n and C_m . This has the obvious meaning, i.e. define $C_{n,s+1} = C_{n,s} \cup C_{m,s}$ if C_n is the smaller class, and rearrange the names of all affected classes. To say that a class C_n is *larger than* m (or *another class* C_m) means that $n > m$.

The reader might expect the construction of A to be carried out on a priority tree using infinite injury, since one might want to guess if a certain c.e. set W intersects infinitely many different A -classes or not. However, the actions of the different requirements are sufficiently independent from each other that we will construct A in an “injury-free” way.

We will always prevent R_e from collapsing two classes smaller than e . Of course, R_e is still allowed to collapse two or more classes if at most one class being collapsed is smaller than e .

We now describe the basic strategy for R_{2e+1} . We act for R_{2e+1} at a stage $s > 2e + 1$ if at least $2\bar{s}$ many different A_s -classes contain an element of $W_{e,s}$, where $\bar{s}$ is the largest stage $< s$ where we had acted for R_{2e+1} . (Obviously, if $\bar{s}$ does not exist, set it to be equal to $2e + 1$). At such a stage s , we act for R_{2e+1} by doing the following. Let N be least such that $C_{N,s}$ does not yet contain at least two elements of W_e . If $N > \bar{s}$ do nothing else. Otherwise find the least pair $M > M' > \max\{N, 2e + 1\}$ such that $C_{M,s}$ and $C_{M',s}$ each contains an element of W_e ; such a pair $M' < M$ exists by assumption (note that $\max\{N, 2e + 1\} \leq \bar{s}$). Collapse the classes $C_N, C_{M'}$ and C_M .

We now describe the basic strategy for R_{2e} . At a stage $s > 2e$, we check to see if:

- For every $\langle i', j' \rangle < s$, $\varphi(i', j') \downarrow = 0$ implies that i' and j' are in different classes, and
- There exists some pair of numbers $\langle i, j \rangle < s$ such that $\varphi_e(i, j) \downarrow = 0$ and i and j currently belong to different A -classes which are both larger than $2e$.

If this is true, collapse the classes containing i and j . Otherwise do nothing else at this stage.

Construction of A : At the beginning, let $A_0 = \text{id}$, i.e. $C_{n,0} = \{n\}$ for all n . At stage $s > 0$, let all requirements R_e for $e < s$ act (if necessary) according to the basic strategy above.

We now verify that all the requirements are met. First of all, note that each class C_n can be collapsed with another class C_m infinitely often, and so each class can be infinite. However, each pair of classes C_n and C_m are collapsed with each other only finitely often. We now verify this fact. More specifically, suppose there is a pair $n < m$ such that there are infinitely many steps s in the construction in which $C_{n,s}$ and $C_{m,s}$ are collapsed with each other. Fix a pair such that $\langle n, m \rangle$ is the least; recall that the standard pairing function $\langle \cdot, \cdot \rangle$ has the property that $\langle x, y \rangle < \langle x+1, y \rangle$ and $\langle x, y \rangle < \langle x, y+1 \rangle$ for all x, y .

Now by examining the construction, only finitely many requirements (specifically, only those R_e for $e < m$) can cause the current C_n and the current C_m to be collapsed. Therefore, we can fix an e where R_e does this infinitely often. This means that e cannot be even, because if R_e ever collapses any two classes, it must be that $\varphi(i, j) = 0$ and that i, j are put the same A -class, which will block R_e from doing anything else. Therefore, e has to be odd.

By the minimality of the pair $\langle n, m \rangle$, there is a time after which every class C_k for $k \leq n$ is eventually growing monotonically (as eventually no two classes smaller or equal to n is collapsed). Formally, this means that there exists a stage t_0 such that for all $s \geq t_0$, and $k \leq n$, we have $C_{k,s} \subseteq C_{k,s+1}$. If R_e collapses C_n and C_m

after stage t_0 , then some class less than or equal to C_n will now contain at least two elements of W . By monotonicity, eventually every class less than or equal to C_n will contain at least two elements of W , and so makes it impossible for R_e to collapse C_n and C_m (in fact, any class larger than C_n) again.

This means that *every* A -class C_n eventually grows monotonically. This implies that A will (at the end) consist of infinitely many different classes. We now apply this to show that every requirement is satisfied. To show that R_{2e} is met, suppose that φ_e is total and correctly computes A . Fix any pair of elements i, j belonging to the $2e + 1$ -th and the $2e + 2$ -th classes respectively. Then $\varphi_e(i, j)$ must be equal to 0. But this means that at all large enough stages, R_{2e} would want to act to collapse these two classes. Since this is impossible, it must be because at all large enough stages, the first clause in the basic strategy for R_{2e} fails. This is also impossible if $\varphi_e = A$.

Now we argue that R_{2e+1} is met. Suppose that W_e intersects infinitely many A -classes, and let C_N be the smallest A -class which contains at most one element of W_e . Eventually from some point on, the classes $C_0, C_1, \dots, C_N$ will be growing monotonically. Since W_e intersects infinitely many A -classes, there must be infinitely many stages where we will act for R_{2e+1} . Eventually $\bar{s}$ will be larger than N , and we will act to put two elements of W_e into C_N . $\square$

We now turn to our second counter-example to downward density, this time constructing a c.e. c -degree which is above $\deg_c(\text{id})$. Unlike in the proof of Theorem 2.3.9, we cannot directly use Lemma 2.3.8 to show that the constructed c -degree is not set-induced, because we have to now make the constructed degree bound $\deg_c(\text{id})$. A modification is needed:

Lemma 2.3.10 (Modified Lemma 2.3.8). *Let R be any non-computable c.e. equivalence relation with the property that $(2n, 2m) \notin R$ for every $n \neq m$. We call an R -class even if it contains an even number, otherwise we call the R -class odd.*

Suppose R has the property that given any c.e. set W , if infinitely many odd R -classes contain an element of W , then every R -class contains at least two elements

of W . Then R is a counter-example to downward density and the only set-induced degree below $\deg_c(R)$ is $\deg_c(\text{id})$ or below.

Proof. Fix R as above. Let $S \leq_c R$ via a computable function f . If $\text{rng}(f)$ intersects only finitely many odd R -classes then as R is c.e., S is computable. Otherwise each R -class must contain an element of $\text{rng}(f)$. Since R is c.e., this allows us as before to show that $R \leq_c S$.

Now suppose $R_A \leq_c R$ via f for some co-infinite set A . Then $\text{rng}(f)$ must intersect infinitely many different odd R -classes (otherwise $R_A \equiv_c \text{id}$). Hence each R -class will contain at least two elements of $\text{rng}(f)$. Since R is not computable, there are at least two R -classes. This means that $f(x) R f(y)$ for some pair $x, y \notin A$ and $x \neq y$, a contradiction. $\square$

In Theorem 2.3.11 we will use Lemma 2.3.10 to ensure that the degree of the constructed ceer is not above a set-induced degree.

Theorem 2.3.11. *There is a c.e. c -degree $\mathbf{a} > \deg_c(\text{id})$ such that $\mathbf{a}$ is not set-induced and for every $\mathbf{b} \leq \mathbf{a}$, either $\mathbf{b}$ is computable or $\mathbf{b} \geq \mathbf{a}$. In other words, $\mathbf{a}$ is a strong minimal cover of $\deg_c(\text{id})$.*

Proof. We will modify the proof of Theorem 2.3.9. We are going to build a c.e. equivalence relation A and ensure that $(2n, 2m) \notin A$ for every $n \neq m$. We will ensure the following requirements:

R_{2e} : $A \neq \varphi_e$.

R_{2e+1} : W_e intersects infinitely many odd A -classes $\Rightarrow$ every A -class contains at least two elements of W_e .

It should be clear that these requirements ensure the properties needed for A . The R_{2e} requirements ensure that A is not computable. Lemma 2.3.10 and R_{2e+1} requirements together ensure that $\deg_c(A)$ is a counter-example to downward density and is not of set-induced degree. Clearly $\text{id} \leq_c A$ by sending $x \mapsto 2x$.

We remark that we can satisfy R_{2e+1} (trivially) by ensuring that A only has finitely many odd classes at the end; however it is for the sake of R_{2e} that we will deliberately ensure that we end up with infinitely many odd classes at the end.

We follow closely the proof of Theorem 2.3.9. Let $o(n, s)$ be the n -th odd class at stage s , i.e. $C_{o(n, s), s}$ is an odd class and there are exactly n many odd classes amongst $C_{0, s}, C_{1, s}, \dots, C_{o(n, s), s}$.

We now describe the basic strategy for R_{2e+1} . We act for R_{2e+1} at a stage $s > 2e + 1$ if at least $2\bar{s}$ many different odd A_s -classes contain an element of $W_{e, s}$, where $\bar{s}$ is the largest stage $< s$ where we had acted for R_{2e+1} . (Obviously, if $\bar{s}$ does not exist, set it to be equal to $2e + 1$). At such a stage s , we act for R_{2e+1} by doing the following. Let N be least such that $C_{N, s}$ does not yet contain at least two elements of W_e (note that $C_{N, s}$ might be an even class). If $N > \bar{s}$ do nothing else. Otherwise find the least pair $M > M' > \max\{o(N, s), o(2e + 1, s)\}$ such that $C_{M, s}$ and $C_{M', s}$ each contains an element of W_e and are both odd classes; such a pair $M' < M$ exists by assumption (note that $\max\{N, 2e + 1\} \leq \bar{s}$). Collapse the classes $C_N, C_{M'}$ and C_M .

We now describe the basic strategy for R_{2e} . At a stage $s > 2e$, we check to see if:

- For every $\langle i', j' \rangle < s$, $\varphi(i', j') \downarrow = 0$ implies that i' and j' are in different classes, and
- There exists some pair of numbers $\langle i, j \rangle < s$ such that $\varphi_e(i, j) \downarrow = 0$ and i and j currently belong to different A -classes which are both larger than $o(2e, s)$ and are both odd classes.

If this is true, collapse the classes containing i and j . Otherwise do nothing else at this stage.

Construction of A : At the beginning, let $A_0 = \text{id}$, i.e. $C_{n, 0} = \{n\}$ for all n . At stage $s > 0$, let all requirements R_e for $e < s$ act (if necessary) according to the basic strategy above.

We now verify that all the requirements are met. First of all, note that no two even classes are ever collapsed together, and thus $(2n, 2m) \notin A$ for every $n \neq m$.

Next, we observe the following facts:

- Every A -class C_n eventually grows monotonically. This is because $o(n, s) \geq n$ for every n and s , and so the same argument from Theorem 2.3.9 applies.
- For each n , there are only finitely many steps in the construction where we collapse $C_{m,s}$ and $C_{o(n,s),s}$ for some $m < o(n, s)$. To see this, note that this can only be done by some requirement R_e for $e < n$. Fix some odd R_e which does this infinitely often. Since each class eventually grows monotonically, this means that eventually R_e will be collapsing the three classes $N < M' < M$ where $N > n$. This of course means that M' and M are both larger than $o(n, s)$ which is a contradiction.
- At the end, A must contain infinitely many odd classes. This follows because of the preceding fact and also that if $C_{n,s}$ is collapsed with $C_{m,s}$ and $n < m$, then $C_{m,s}$ must be an odd class.

Since A contains infinitely many odd classes, the same argument from before works to show that R_e is met for every e . $\square$

Notice that if $R \leq_c S$ and S is a ceer, then so is R . Therefore we can immediately conclude that any class of equivalence relations containing the c.e. equivalence relations is not downwards dense:

Corollary 2.3.12. *The class of Σ_n^0 equivalence relations, the n -c.e. equivalence relations and the ω -c.e. equivalence relations are not downwards dense for $n \geq 1$.*

2.4 Downward density and Π_1^0 equivalence relations

We now turn to the structure of the Π_1^0 c -degrees. In particular, we will show in this section that downward density fails in the structure of Π_1^0 c -degrees. Before making our statements precise, we first make a few trivial observations about the

Π_1^0 c -degrees. Recall that the computable c -degrees form an initial segment of the Π_1^0 c -degrees. In fact, each Π_1^0 equivalence relation has comparable c -degree with id : If R is Π_1^0 and has infinitely many classes, then we can compute an infinite set of elements which are pairwise R -inequivalent, allowing us to reduce id to R . On the other hand, if R is Π_1^0 with finitely many distinct classes then every element is eventually R -inequivalent with all but one class, so it is computable.

Since every Π_1^0 c -degree is comparable with $\deg_c(\text{id})$, we will construct a strong minimal cover of $\deg_c(\text{id})$ which is not of set-induced degree.

Theorem 2.4.1. *There exists a Π_1^0 c -degree $\mathbf{a}_2 > \deg_c(\text{id})$ such that $\mathbf{a}_2$ is a strong minimal cover of $\deg_c(\text{id})$, and where $\mathbf{a}_2$ is not of set-induced degree.*

As we will later formally prove Theorem 2.4.2, a stronger version of this theorem, we will not prove Theorem 2.4.1. Instead, we give some relevant definitions. For any integer $x \in \omega$ and equivalence relation E , $[x]_E$ is defined to be the equivalence class of E containing x ; where the context is clear we simply write $[x]$. (Note that $[s]$ also used as a stage notation earlier, but there should be no confusion). An equivalence relation E is said to be *size- n* if for every x, y such that $E(x, y) = 1$, we have $\lfloor \frac{x}{n} \rfloor = \lfloor \frac{y}{n} \rfloor$. In other words, each size- n equivalence relation has $\#[x] \leq n$ as well as an upper bound on the set $[x]$ for every x . Notice that id is a size-1 equivalence relation.

It is easy to see that a size- n Π_1^0 equivalence relation E is computable iff the function $F_E : x \mapsto \#[x]_E$ is computable.

Our goal is to extend Theorem 2.4.1 to an infinite sequence $\deg_c(\text{id}) = \mathbf{a}_1 <_c \mathbf{a}_2 <_c \mathbf{a}_3 <_c \dots$ such that $\mathbf{a}_{j+2}$ is a strong minimal cover of $\mathbf{a}_{j+1}$ for every $j \in \omega$. In the next theorem, Theorem 2.4.2, we give the formal proof for the existence of $\mathbf{a}_2$ and $\mathbf{a}_3$. The infinite sequence $\deg_c(\text{id}) = \mathbf{a}_1 <_c \mathbf{a}_2 <_c \mathbf{a}_3 <_c \dots$ is a straightforward generalization of Theorem 2.4.2 and there are no new mathematical ideas needed apart from increased notational complexity.

Theorem 2.4.2. *There exist Π_1^0 c -degrees $\mathbf{a}_2$ and $\mathbf{a}_3$ which are not set-induced such that $\deg_c(\text{id}) < \mathbf{a}_2 < \mathbf{a}_3$, $\mathbf{a}_2$ is a strong minimal cover of $\deg_c(\text{id})$, and $\mathbf{a}_3$ is*

a strong minimal cover of $\mathbf{a}_2$.

Proof. We shall construct a size-2 Π_1^0 equivalence relation A_2 and a size-3 Π_1^0 equivalence relation A_3 satisfying the following requirements.

Q : $A_2 \leq_c A_3$.

$R_{\langle m, n \rangle}^1$: $E_m \leq_c A_2$ via $f_n \Rightarrow \exists$ computable function $g_{\langle n, m \rangle}$ s.t. $A_2 \leq_c E_m$ via $g_{\langle n, m \rangle}$ or E_m is computable.

$R_{\langle i, j \rangle}^2$: $E_i \leq_c A_3$ via $f_j \Rightarrow \exists$ computable function $h_{\langle i, j \rangle}$ s.t. $A_3 \leq_c E_i$ via $h_{\langle i, j \rangle}$ or $E_i \leq_c A_2$.

P_e^1 : $F_{A_2} \neq \varphi_e$.

P_e^2 : $A_3 \not\leq_c A_2$ via φ_e .

Here, we think of E_m and E_i as the “pull-back” of A_2 and A_3 under f_n and f_j respectively.

Recall F_E is the size function of classes in E . By Lemma 2.3.7, $\deg_c(A_2)$ and $\deg_c(A_3)$ are not set-induced.

We fix the starting values $A_{2,0} = \{(i, i), (2i, 2i+1), (2i+1, 2i) \mid i \in \omega\}$ and $A_{3,0} = \{(i, i), (3i, 3i+1), (3i+1, 3i+2), (3i+2, 3i), (3i+1, 3i), (3i+2, 3i+1), (3i, 3i+2) \mid i \in \omega\}$. Q is a global requirement and will be satisfied by ensuring that for every z , $[2z]_{A_2}$ is isolated if and only if $[3z]_{A_3}$ is isolated for every z .

Tree of strategies

Effectively order the requirements (of order type ω). We will carry out the construction on a finitely branching priority tree, where every node at the same level is assigned the same requirement. The outcomes of a node assigned an R^1 or R^2 requirement are $\text{fin} < \infty$. The outcomes of a node assigned a P^1 or P^2 requirement are $w < s$.

Notation

For any node α , define α^- to be β of the longest length such that $\beta * \infty \subseteq \alpha$. Of course, α^- need not exist. During the construction, we define δ_s to be the stage s approximation to the true path.

The length function for $R^1_{\langle m, n \rangle}$ is defined as $l(\langle m, n \rangle, s) :=$ the largest k such that for every $\langle x, y \rangle < k$, we have $f_{n,s}(x) \downarrow$, $f_{n,s}(y) \downarrow$ and $E_{m,s}(x, y) = A_{2,s}(f_{n,s}(x), f_{n,s}(y))$.

The length function for $R^2_{\langle i, j \rangle}$ is defined similarly as $L(\langle i, j \rangle, s) :=$ the largest k such that for every $\langle x, y \rangle < k$, we have $f_{j,s}(x) \downarrow$, $f_{j,s}(y) \downarrow$, and $E_{i,s}(x, y) = A_{3,s}(f_{j,s}(x), f_{j,s}(y))$.

Definition 2.4.3 (R^1 -expansionary stage). If α is assigned an R^1 strategy, then we call a stage $s > 0$ an α -expansionary stage if the following hold.

- $\alpha \subset \delta_s$.
- $l(\alpha, s) > l(\alpha, t)$, where t is the largest α -expansionary stage less than s (if it exists).
- $\#(rng(f_{\alpha,s}) - \{0, 1, \dots, t\}) > 2t$.
- There exists an even number y such that:
 - If α^- exists and α^- is assigned an R^1 requirement then $y \in S_{\alpha^-}$ (this parameter S_{α^-} will be defined later). If α^- is assigned an R^2 requirement then $\frac{3y}{2} \in S_{\alpha^-}$.
 - $y > \max S_\beta$ for all β assigned an R^1 requirement such that $\beta <_L \alpha$ or $\beta * fin \subseteq \alpha$.
 - $\frac{3y}{2} > \max S_\beta$ for all β assigned some R^2 requirement such that $\beta <_L \alpha$ or $\beta * fin \subseteq \alpha$.
 - $y > x_\beta$ (the follower assigned to the positive strategy β) for each β of higher priority than α assigned some P^1 requirement.
 - $y > \frac{2x_\beta}{3}$ for each higher priority β assigned some P^2 requirement.

$$- [y]_{A_{2,s}} \subseteq \text{rng}(f_{\alpha,s}).$$

$$- \#[y]_{A_{2,s}} = 2.$$

- $f_{\alpha,s}$ looks like a reduction on all the numbers above.

Definition 2.4.4 (R^2 -expansionary stage). If α is assigned an R^2 strategy, then we call a stage $s > 0$ an α -expansionary stage if the following hold.

- $\alpha \subset \delta_s$.
- $L(\alpha, s) > L(\alpha, t)$, where t is the largest α -expansionary stage less than s (if it exists).
- $\#(\text{rng}(f_{\alpha,s}) - \{0, 1, \dots, t\}) > 3t$.
- There exists a number y divisible by 3 such that:
 - If α^- exists and α^- is assigned an R^1 requirement then $\frac{2y}{3} \in S_{\alpha^-}$ (this parameter S_α will be defined later). If α^- is assigned an R^2 requirement then $y \in S_{\alpha^-}$.
 - $\frac{2y}{3} > \max S_\beta$ for all β assigned an R^1 requirement such that $\beta <_L \alpha$ or $\beta * \text{fin} \subseteq \alpha$.
 - $y > \max S_\beta$ for all β assigned some R^2 requirement such that $\beta <_L \alpha$ or $\beta * \text{fin} \subseteq \alpha$.
 - $\frac{2y}{3} > x_\beta$ (the follower assigned to the positive strategy β) for each β of higher priority than α assigned some P^1 requirement.
 - $y > x_\beta$ for each higher priority β assigned some P^2 requirement.
 - $[y]_{A_{3,s}} \subseteq \text{rng}(f_{\alpha,s})$.
 - $\#[y]_{A_{3,s}} = 3$.
- $f_{\alpha,s}$ looks like a reduction on all the numbers above.

Construction

At stage s , we will define the approximation to the true path δ_s of length s by induction. We initialize all nodes β to the right of δ_s , i.e., initialize the functions g_β , h_β , the set S_β and the witness x_β . Suppose that $\alpha \subset \delta_s$ has been defined. We first describe how to act for the node α , and then how to define $\delta_s(|\alpha|)$. This depends on the requirement assigned to α .

α is assigned an R^1 requirement. If the current stage s is not α -expansionary, do nothing for α and take the outcome fin . (This means of course that $\delta_s(|\alpha|) = fin$). Suppose the current stage $s > 0$ is an α -expansionary stage. We take outcome ∞ and do the following actions for α .

Let y_α be the least y satisfying the conditions of Definition 2.4.3. We fix $\{c_\alpha, c'_\alpha\} \subseteq dom(f_\alpha)$ such that $f_\alpha(c_\alpha) = y_\alpha$ and $f_\alpha(c'_\alpha) = y_\alpha + 1$. Our action for α is to enumerate y_α into S_α , which is a collection of all such y s chosen by α in this way. Definition 2.4.3 ensures that at s , there are at least $t + 1$ many distinct equivalence classes in $rng(f_\alpha)$ which are larger than t . Here we let $t' < t$ be the previous two α -expansionary stages before s (if s is the first or second expansionary stage, we give a separate description below). We know there are at most $y_{\alpha,t} < t$ many elements strictly between the class $[y_{\alpha,t'}]$ and $[y_{\alpha,t}]$, where $y_{\alpha,t'}, y_{\alpha,t}$ are the elements enumerated into S_α at stage t' and t .

We isolate every z strictly between $[y_{\alpha,t'}]$ and $[y_{\alpha,t}]$. That is, for every z with $\max[y_{\alpha,t'}] < z < \min[y_{\alpha,t}]$, we make z isolated. Also isolate $\left\lceil 3\left\lfloor \frac{z}{2} \right\rfloor \right\rceil_{A_3}$ for each such z . We now have $k_\alpha < y_{\alpha,t} < t$ many isolated points there. Pick k_α many elements $b_{\alpha,1}, b_{\alpha,2}, \dots, b_{\alpha,k_\alpha}$ from $dom(f_\alpha)$ all of which are in distinct E -classes, and which are distinct from $\{c_\alpha, c'_\alpha\}$, and all previous values in $rng(g_\alpha)$. This exists because f_α looks like a reduction, and $k_\alpha < t$. Note that if b' is a previous value in $rng(g_\alpha)$ then $f_\alpha(b')$ converges at stage t and hence $f_\alpha(b') < t$ (by the usual convention).

Now define $g_\alpha(\max[y_{\alpha,t'}] + i) = b_{\alpha,i}, i = 1, 2, \dots, k_\alpha$, $g_\alpha(y_\alpha) := c_\alpha$ and $g_\alpha(y_\alpha + 1) := c'_\alpha$.

For clarity, we now describe our actions at the first and second α -expansionary

stages. If the current stage s is the first α -expansionary stage after an initialization, define $y_{\alpha,s}$ as in Definition 2.4.3, and $g_\alpha(y_\alpha)$ and $g_\alpha(y_\alpha + 1)$, and do nothing else. Note that we should not allow α to isolate any number less than $y_{\alpha,s}$, as these classes might be restrained by a higher priority requirement. If s is the second α -expansionary stage after an initialization, define $y_{\alpha,s}$, $g_\alpha(y_\alpha)$ and $g_\alpha(y_\alpha + 1)$ as usual. *Do not* isolate any number less than $[y_{\alpha,t}]$, and define $g_\alpha(i)$ as above for every $i < \min[y_{\alpha,t}]$, mapping $g_\alpha(i) = g_\alpha(i')$ iff i and i' are in the same class.

α is assigned an R^2 requirement. This works similarly to an R^1 node. If the current stage s is not α -expansionary, do nothing for α and take the outcome *fin*. Suppose the current stage $s > 0$ is an α -expansionary stage. We take outcome ∞ and do the following actions for α .

Let y_α be the least y satisfying the conditions of Definition 2.4.4. We fix $\{c_\alpha, c'_\alpha, c''_\alpha\} \subseteq \text{dom}(f_\alpha)$ such that $f_\alpha(c_\alpha) = y_\alpha$, $f_\alpha(c'_\alpha) = y_\alpha + 1$ and $f_\alpha(c''_\alpha) = y_\alpha + 2$. Our action for α is to enumerate y_α into S_α . We isolate every z strictly between $[y_{\alpha,t'}]$ and $[y_{\alpha,t}]$ (these are A_3 -classes). Also isolate $[2\lfloor \frac{z}{3} \rfloor]_{A_2}$ for each such z . We now have $k_\alpha < y_{\alpha,t} < t$ many isolated points there. Pick k_α many elements $b_{\alpha,1}, b_{\alpha,2}, \dots, b_{\alpha,k_\alpha}$ from $\text{dom}(f_\alpha)$ all of which are in distinct E -classes, and which are distinct from $\{c_\alpha, c'_\alpha, c''_\alpha\}$, and all previous values in $\text{rng}(h_\alpha)$. Again, this exists for the same reasons as above.

Now define $h_\alpha(\max[y_{\alpha,t'}] + i) = b_{\alpha,i}$, $i = 1, 2, \dots, k_\alpha$, $h_\alpha(y_\alpha) := c_\alpha$, $h_\alpha(y_\alpha + 1) := c'_\alpha$ and $h_\alpha(y_\alpha + 2) := c''_\alpha$. At the first two α -expansionary stages, we follow the description above for R^1 .

α is assigned a P^1 requirement. If the witness x_α is not yet defined, check if there is a least even number x_α at the current stage $s > 0$ such that:

- If α^- exists and is assigned an R^1 requirement then $x_\alpha \in S_{\alpha^-}$.
- If α^- exists and is assigned an R^2 requirement then $\frac{3x_\alpha}{2} \in S_{\alpha^-}$.
- $x_\alpha > \max S_\beta$ for all β assigned an R^1 requirement, such that $\beta <_L \alpha$ or $\beta * \text{fin} \subseteq \alpha$.

- $\frac{3x_\alpha}{2} > \max S_\beta$ for all β assigned an R^2 requirement, such that $\beta <_L \alpha$ or $\beta * fin \subseteq \alpha$.
- $x_\alpha > x_\beta$ for all P^1 nodes β of higher priority.
- $\frac{3x_\alpha}{2} > x_\beta$ for all P^2 nodes β of higher priority.
- $\#[x_\alpha]_{A_{2,s}} = 2$.

If these conditions are met, pick this number x_α as the follower. (Here φ_α refers to φ_e assigned to the node α).

Otherwise, x_α has already been picked prior to the current stage. If this is so, check if $\varphi_{\alpha,s}(x_\alpha) \downarrow = 2$. If no, do nothing else. If yes, isolate all members of the class $[x_\alpha]_{A_{2,s}}$ and $[\frac{3x_\alpha}{2}]_{A_{3,s}}$. Declare diagonalization successful and never again attempt diagonalization (unless α is initialized).

If diagonalization is not yet successful, take outcome w , otherwise take outcome s .

α is assigned a P^2 requirement. If the witness x_α is not yet defined, check if there is a least number x_α divisible by 3 at the current stage $s > 0$ such that:

- If α^- exists and is assigned an R^1 requirement then $\frac{2x_\alpha}{3} \in S_{\alpha^-}$.
- If α^- exists and is assigned an R^2 requirement then $x_\alpha \in S_{\alpha^-}$.
- $\frac{2x_\alpha}{3} > \max S_\beta$ for all β assigned an R^1 requirement, such that $\beta <_L \alpha$ or $\beta * fin \subseteq \alpha$.
- $x_\alpha > \max S_\beta$ for all β assigned an R^2 requirement, such that $\beta <_L \alpha$ or $\beta * fin \subseteq \alpha$.
- $\frac{2x_\alpha}{3} > x_\beta$ for all P^1 nodes β of higher priority.
- $x_\alpha > x_\beta$ for all P^2 nodes β of higher priority.
- $\#[x_\alpha]_{A_{3,s}} = 3$.

If these conditions are met, pick this number x_α as the follower.

Otherwise, x_α has already been picked prior to the current stage. If this is so, check if $\varphi_\alpha(x_\alpha) \downarrow$, $\varphi_\alpha(x_\alpha + 1) \downarrow$, $\varphi_\alpha(x_\alpha + 2) \downarrow$ and for all $i, j \leq 2$, $A_2(\varphi_\alpha(x_\alpha + i), \varphi_\alpha(x_\alpha + j)) = 1$. If no, do nothing else. If yes, isolate all members of the class $\left[\frac{2x_\alpha}{3}\right]_{A_{2,s}}$ and $[x_\alpha]_{A_{3,s}}$. Declare diagonalization successful and never again attempt diagonalization (unless α is initialized).

If diagonalization is not yet successful, take outcome w , otherwise take outcome s .

Verification

Denote by δ the true path of the construction, i.e. $\delta = \liminf_{s \rightarrow \infty} \delta_s$. Now we are going to show that every node $\alpha \subset \delta$ ensures the satisfaction of the requirement assigned to it. Fix $\alpha \subset \delta$, and fix a stage s_0 such that for every $s > s_0$, $\delta_s \not\prec_L \alpha$.

First of all, we check that the global requirement Q is met. It is easy to see that every step in the construction isolates $[2k]_{A_2}$ iff it isolates $[3k]_{A_3}$, for each k . Thus, each A_3 class has size 1 or 3. In this case, we clearly have $A_2 \leq_c A_3$ via the reduction $2k \mapsto 3k, 2k + 1 \mapsto 3k + 1$.

Fact 2.4.5. Suppose $\alpha * \infty \subseteq \beta$. If α is an R^1 -node and β is an R^2 -node and $z \in S_\beta$ then $\frac{2z}{3} \in S_\alpha$. If α is an R^2 -node and β is an R^1 -node and $z \in S_\beta$ then $\frac{3z}{2} \in S_\alpha$. If α and β are R -nodes of the same type then $S_\beta \subseteq S_\alpha$.

Proof. Easy to check (by a straightforward induction on $|\beta|$). □

Lemma 2.4.6. *Each P^1 requirement succeeds.*

Proof. Assume that α is assigned a P^1 requirement. Suppose α is never able to find a suitable x_α . The first two conditions do not pose a problem because if α^- exists, then as $\alpha^- * \infty \subset \delta$, there will be infinitely many elements z enumerated into S_{α^-} . Furthermore for each new z that α^- adds to S_{α^-} , we must have $\#[z]_{A_2} = 2$ and $\#[\frac{3z}{2}]_{A_3} = 3$ or vice versa. The third and fourth conditions also do not pose a problem because there are only finitely many such β , and none of them will increase

S_β after stage s_0 . The fifth and sixth conditions are similar, as a P -node β along the true path will only pick finitely many different x_β . Thus eventually x_α must be picked by α (and never again redefined).

Now suppose α picks its final x_α at some stage $t > s_0$. It is not hard to see that after stage t , no node other than α itself will be able to isolate $[x_\alpha]$: For an R -node β extending or to the right of α , use the fact that β is prevented from isolating anything less than the first β -expansionary stage. For an R -node β such that $\beta * \infty \subseteq \alpha$, apply Fact 2.4.5 and the fifth bullet of the strategy.

Therefore, if $\varphi_\alpha(x_\alpha) \neq 2$, then α never acts and so obviously $F_{A_2} \neq \varphi_e$. Otherwise if $\varphi_\alpha(x_\alpha) = 2$ we will isolate $[x_\alpha]$ making $F_{A_2}(x_\alpha) = 1$. $\square$

Lemma 2.4.7. *Each P^2 requirement succeeds.*

Proof. Assume that α is assigned a P^2 requirement. As in the proof of Lemma 2.4.6, α must eventually be able to define a final x_α . After this is picked by α , no node other than α itself is able to isolate $[x_\alpha]$. Suppose $\varphi_\alpha(x_\alpha + i) \downarrow$ for $i = 0, 1, 2$. If $A_2(\varphi_\alpha(x_\alpha + i), \varphi_\alpha(x_\alpha + j)) = 0$ for some $i, j = 0, 1, 2$, then α will never do anything to $[x_\alpha]$, which means that $A_3(x_\alpha + i, x_\alpha + j) = 1$. On the other hand, if $A_2(\varphi_\alpha(x_\alpha + i), \varphi_\alpha(x_\alpha + j)) = 1$ for every i, j , then as each A_2 -class has size at most 2, this means that $\varphi_\alpha(x_\alpha + i) = \varphi_\alpha(x_\alpha + j)$ for some $i \neq j$. However α will isolate $[x_\alpha]$ which means that $A_3(x_\alpha + i, x_\alpha + j) = 0$. In either case, φ_α is not a possible reduction for $A_3 \leq_c A_2$. $\square$

Lemma 2.4.8. *Each R^1 requirement succeeds.*

Proof. Assume that α is assigned an R^1 requirement, and that f_n witnesses $E_m \leq_c A_2$ (otherwise there is nothing to prove). In particular, f_n is total, $E_m(x, y) = A_2(f_n(x), f_n(y))$ for every x, y and $l(\alpha, s) \rightarrow \infty$.

Suppose that $\alpha * fin \subset \delta$, hence there is a final α -expansionary stage. Then we claim that E_m is computable. Since there are only finitely many expansionary stages, hence either $rng(f_\alpha)$ is finite, or the number y cannot be found after the final α -expansionary stage. Obviously if $rng(f_\alpha)$ is finite, then E_m is computable.

So assume that y cannot be found. Since S_{α^-} is infinite, and $\max S_\beta$ and x_β in the second through fifth conditions for y in Definition 2.4.3 are all eventually finite, the only thing preventing y from being found is if for almost every $z, z' \in \text{rng}(f_\alpha)$ we have $A_2(z, z') = 0$. (Recall that α^- will isolate every $z \notin S_{\alpha^-}$). But this of course means that E is computable (in fact, $E \equiv_c \text{id}$).

Now suppose that $\alpha * \infty \subset \delta$. Hence, S_α is infinite. Suppose $s_1 > s_0$ is the first α -expansionary stage after the final initialization to α . We claim that no strategy can isolate any $z < y_{\alpha, s_1}$ after stage s_1 . (Note that y_{α, s_1} is the smallest element of S_α). Again we apply Fact 2.4.5 and the fifth bullet of the strategy, and note that every node of lower priority than β is also initialized when β is initialized. This means that for every $z < y_{\alpha, s_1}$, we have $[z]_{s_1} = [z]$.

We shall argue that g_α is total and $A_2(x, y) = E_m(g_\alpha(x), g_\alpha(y))$ for every x, y . Clearly g_α is total because S_α is infinite. There are three possibilities for x . First, if $\max[y_{\alpha, t'}] < x < \min[y_{\alpha, t}]$ at some α -expansionary stage s , where $t' \geq s_1$, then the construction will make x isolated. The construction at stage s defines $g_\alpha(x) = b_{\alpha, j}$ for some $b_{\alpha, j}$ found at the α -expansionary stage s , and it is clear that $g_\alpha^{-1}([b_{\alpha, j}]) = \{x\}$. Therefore, $A_2(x, y) = E_m(g_\alpha(x), g_\alpha(y)) = 0$ no matter what y is.

Let's assume now that $x < y_{\alpha, s_1}$. Now at the second expansionary stage, we will define $g_\alpha(x) = b_{\alpha, j}$ for some $b_{\alpha, j}$ found at the second α -expansionary stage. Clearly if $y \geq y_{\alpha, s_1}$ then $y \notin [x]$, and we have $A_2(x, y) = E_m(g_\alpha(x), g_\alpha(y)) = 0$. On the other hand if $y < y_{\alpha, s_1}$ then the construction at the second α -expansionary stage defines $g_\alpha(y) = b_{\alpha, k}$ where $k = j$ iff $[y]_{s_1} = [x]_{s_1}$. However, by our comments above, $[x]_{s_1} = [x]$ and $[y]_{s_1} = [y]$ and so we certainly have $A_2(x, y) = E_m(g_\alpha(x), g_\alpha(y))$.

Finally let's assume that $x = y_{\alpha, s}$ or $x = y_{\alpha, s} + 1$ for some α -expansionary stage s . Without loss of generality, assume $x = y_{\alpha, s}$. The construction defines $g_\alpha(x) = c_\alpha$ (for the value found at s), in particular, $f_\alpha(g_\alpha(x)) = x$. If $y \neq y_{\alpha, s}$ or $y_{\alpha, s} + 1$ then of course $A_2(x, y) = E_m(g_\alpha(x), g_\alpha(y)) = 0$. If $y = y_{\alpha, s} + 1$ then $f_\alpha(g_\alpha(y)) = y$ as well, and so $A_2(x, y) = A_2(f_\alpha(g_\alpha(x)), f_\alpha(g_\alpha(y))) = E_m(g_\alpha(x), g_\alpha(y))$. $\square$

Lemma 2.4.9. *Each R^2 requirement succeeds.*

Proof. Assume that α is assigned an R^2 requirement, and that f_α witnesses $E_i \leq_c A_3$. In particular, f_α is total, $E_i(x, y) = A_3(f_\alpha(x), f_\alpha(y))$ for every x, y and $L(\alpha, s) \rightarrow \infty$.

If $\alpha * \infty \subset \delta$ then we follow the proof of Lemma 2.4.8 to show that h_α is total and witnesses that $A_3 \leq_c E_i$. So we assume that $\alpha * \text{fin} \subset \delta$. Hence there is a final α -expansionary stage. We claim that $E_i \leq_c A_2$. Again since there are only finitely many expansionary stages, either $\text{rng}(f_\alpha)$ is finite, or the number y cannot be found after the final α -expansionary stage. Obviously if $\text{rng}(f_\alpha)$ is finite, then E_i is computable. So assume that y cannot be found. As before, the only thing preventing y from being found is if for almost every number z , we have that $\#([z]_{A_3} \cap \text{rng}(f_\alpha)) \leq 2$.

We first argue that $E_i \leq_c A_2 \sqcup \text{id}$. Recall that given equivalence relations S and T , we defined $S \sqcup T = \{(2x, 2y) \mid S(x, y) = 1\} \cup \{(2x+1, 2y+1) \mid T(x, y) = 1\}$. Fix y_0 such that $\#([z]_{A_3} \cap \text{rng}(f_\alpha)) \leq 2$ for every $z \geq y_0$, and consider a stage t after which $A_3 \upharpoonright y_0$ is stable. Define H to be the following partial computable function. Assume that at every stage $s > t$, exactly one new element z_s enters $\text{rng}(f_\alpha)$ (this set was assumed to be infinite). If $z_s < y_0$ and H is not yet defined on any element of $[z_s]$, then take $H(z_s) = 2s+1$. Otherwise take $H(z_s) = H(z_{s'})$ where $t < s' < s$ is such that $[z_s] = [z_{s'}]$. (Note that it is not important at what stage this is measured, as we assumed that $[z]$ is stable after stage t if $z < y_0$). Suppose $z_s \geq y_0$. If $H(z)$ is not yet defined for any z such that $\lfloor \frac{z}{3} \rfloor = \lfloor \frac{z_s}{3} \rfloor$ then define $H(z_s) = 4\lfloor \frac{z_s}{3} \rfloor$. If $H(z)$ is already defined for exactly one z such that $\lfloor \frac{z}{3} \rfloor = \lfloor \frac{z_s}{3} \rfloor$, define $H(z_s) = 4\lfloor \frac{z_s}{3} \rfloor + 2$. Otherwise $H(z)$ is already defined for two z such that $\lfloor \frac{z}{3} \rfloor = \lfloor \frac{z_s}{3} \rfloor$, and in this case we define $H(z_s) = 2s + 1$.

Now we claim that $H \circ f_\alpha$ is total and witnesses $E_i \leq_c A_2 \sqcup \text{id}$. It is clearly total since $\text{dom}(H) = \text{rng}(f_\alpha)$. Now fix x, y . We shall argue that $E_i(x, y) = (A_2 \sqcup \text{id})(H(f_\alpha(x)), H(f_\alpha(y)))$. First suppose that $f_\alpha(x) < y_0$, then we would have defined $H(f_\alpha(x)) = 2s + 1$ for some s . If $f_\alpha(y)$ is in a class different from $f_\alpha(x)$, then obviously $E_i(x, y) = 0$, and $H(f_\alpha(y))$ is either even or of the form $2v+1$

for some $v \neq s$. In either case, $E_i(x, y) = (A_2 \sqcup \text{id})(H(f_\alpha(x)), H(f_\alpha(y))) = 0$. On the other hand if $f_\alpha(y) < y_0$ and is in the same class as $f_\alpha(x)$, then $E_i(x, y) = (A_2 \sqcup \text{id})(H(f_\alpha(x)), H(f_\alpha(y))) = 1$.

Now if $f_\alpha(x), f_\alpha(y)$ are both $\geq y_0$ and $\lfloor \frac{f_\alpha(x)}{3} \rfloor \neq \lfloor \frac{f_\alpha(y)}{3} \rfloor$ then it is easy to check that $E_i(x, y) = (A_2 \sqcup \text{id})(H(f_\alpha(x)), H(f_\alpha(y))) = 0$. On the other hand suppose $\lfloor \frac{f_\alpha(x)}{3} \rfloor = \lfloor \frac{f_\alpha(y)}{3} \rfloor$. If $H(f_\alpha(x))$ and $H(f_\alpha(y))$ are both even then we are okay, because recall that $[2k]_{A_2}$ is isolated iff $[3k]_{A_3}$ is isolated by the construction, for every k . On the other hand if $H(f_\alpha(x))$ is odd, then this means that $3\lfloor \frac{f_\alpha(x)}{3} \rfloor, 3\lfloor \frac{f_\alpha(x)}{3} \rfloor + 1$ and $3\lfloor \frac{f_\alpha(x)}{3} \rfloor + 2$ are all in $\text{rng}(f_\alpha)$, which means that these elements must be isolated by the construction. This means that $E_i(x, y) = (A_2 \sqcup \text{id})(H(f_\alpha(x)), H(f_\alpha(y))) = 0$. This shows that $E_i \leq_c A_2 \sqcup \text{id}$.

Now we check that $A_2 \sqcup \text{id} \leq_c A_2$. By Lemma 2.4.6 there are infinitely many z such that z is A_2 -isolated. Fix an infinite computable set U containing only A_2 -isolated points, let $U = \{u_0 < u_1 < \dots\}$. Define G by $G(2n+1) = u_{2n+1}$ for each n , and $G(2n) = u_{2n}$ if $n \in U$, and $G(2n) = n$ otherwise. First of all G is clearly injective. It is also easy to check that G witnesses that $A_2 \sqcup \text{id} \leq_c A_2$. $\square$

This ends the proof of Theorem 2.4.2. $\square$

As mentioned earlier, our goal is to extend Theorem 2.4.2 to an infinite sequence $\text{deg}_c(\text{id}) = \mathbf{a}_1 <_c \mathbf{a}_2 <_c \mathbf{a}_3 <_c \dots$:

Theorem 2.4.10. *There exist Π_1^0 c -degrees $\text{deg}_c(\text{id}) = \mathbf{a}_1 <_c \mathbf{a}_2 <_c \mathbf{a}_3 <_c \dots$ such that $\mathbf{a}_{k+2}$ is a strong minimal cover of $\mathbf{a}_{k+1}$ for every k . All constructed degrees are not set-induced.*

Proof. This is a straightforward generalization of Theorem 2.4.2. We want to construct, for each $k > 1$, a size- k Π_1^0 equivalence relation A_k . By Lemma 2.3.7, $\text{deg}_c(A_k)$ is not set-induced.

We again begin with fix $A_{k,0} = \{(ki + j_1, ki + j_2) \mid 0 \leq j_1, j_2 < i\}$. Naturally we wish to satisfy the requirements

Q^k : $A_{k-1} \leq_c A_k$, for $2 \leq k \leq n$.

$R_{\langle m, n \rangle}^k$: $E_m \leq_c A_k$ via $f_n \Rightarrow \exists$ computable function $g_{\langle n, m \rangle}^{(k)}$ s.t. $A_k \leq_c E_m$ via $g_{\langle n, m \rangle}^{(k)}$
or $E_m \leq_c A_{k-1}$, for $2 \leq k \leq n$.

P_e^2 : $f_{A_2} \neq \varphi_e$.

$P_{e'}^k$: $A_k \not\leq_c A_{k-1}$ via $\varphi_{e'}$, for $3 \leq k \leq n$.

The strategy for Q^k is similar as before. We ensure that for every x , k and k' , we isolate $[kx]_{A_k}$ iff we isolate $[k'x]_{A_{k'}}$, and we keep $\#[x]_{A_k} = 1$ or k . This obviously ensures the global requirements are met. R_α^k works similarly as before. It measures $\text{rng}(f_\alpha)$, putting numbers y into S_α whenever it finds a new y such that $\#[y] = k$ and $[y] \subseteq \text{rng}(f_\alpha)$ (amongst other conditions). It then isolates every class not in S_α (except for those blocked by higher priority requirements), as well as the corresponding classes in the other $A_{k'}$, as required by the Q requirements. The P^k -requirements uses a class $[x_\alpha]$ where $\#[x_\alpha] = k$ to diagonalize. Since classes of A_{k-1} have size at most $k-1$, this is achieved by preventing $[x_\alpha]$ from breaking up while waiting for φ to converge. There are no additional difficulties, and we leave the formal verification of the proof to the reader. $\square$

2.5 Upward density

We now turn to the question of upward density. In Sections 2.3 and 2.4 we showed that downward density fails in the classes of c -degrees being considered in this chapter. In this section, we will show that in contrast, upward density holds in these classes. We will first prove this for the Π_1^0 c -degrees.

Theorem 2.5.1. *Let $\mathbf{u}$ be the universal Π_1^0 c -degree, and $\mathbf{a}$ be any Π_1^0 c -degree such that $\mathbf{a} < \mathbf{u}$. Then there exists two incomparable Π_1^0 c -degrees $\mathbf{b}_0$ and $\mathbf{b}_1$ such that $\mathbf{a} < \mathbf{b}_0, \mathbf{b}_1 < \mathbf{u}$.*

Proof. Fix $U \in \mathbf{u}$ and $A \in \mathbf{a}$. We will build B_0 and B_1 satisfying the following

requirements:

$$P_{2e} : B_0 \not\leq_c A \sqcup B_1 \text{ via } \varphi_e,$$

$$P_{2e+1} : B_1 \not\leq_c A \sqcup B_0 \text{ via } \varphi_e.$$

Obviously we will take $\mathbf{b}_0 = \deg_c(A \sqcup B_0)$ and $\mathbf{b}_1 = \deg_c(A \sqcup B_1)$.

The basic strategy for P_{2e} will be a version of the Sacks coding strategy. The idea is that if $\varphi_e(x)$ goes inside the even integers for infinitely many x , then by coding U inside B_0 and on the set of these x s, we can force a disagreement to appear, showing that $B_0 \not\leq_c A$ via φ_e . On the other hand if infinitely many $\varphi_e(x)$ go inside the odd integers, then it will be easy for us to ensure that $B_0 \not\leq_c B_1$ via φ_e , since we are building both sets B_0 and B_1 .

We now describe the basic strategy of P_{2e} . We begin by taking $B_{i,0} = \{(\langle e, x \rangle, \langle e, y \rangle) \mid e, x, y \in \omega\}$ for $i = 0, 1$. Denote $C_e = \{\langle e, x \rangle \mid x \in \omega\}$; thus our starting value of B_i is the disjoint union of $C_e \times C_e$ for all e . As the construction proceeds we will refine B_i on C_e . During the construction we will also declare certain numbers as being *marked*; this serves as a flag on a number.

At stage s of the construction, we act for P_{2e} by taking the following steps. If P_{2e} has just been initialized we pick a fresh value for $i(2e)$ and begin work for B_0 inside $C_{i(2e)}$, and set $M = \langle i(2e), 0 \rangle$. Note that at this point as $i(2e)$ is freshly picked, we have $B_0 \upharpoonright C_{i(2e)} = \{(x, y) \in B_0 \mid x, y \in C_{i(2e)}\} = C_{i(2e)} \times C_{i(2e)}$.

Now suppose that $i(2e)$ is defined and k is least such that $f_{2e}(k) \uparrow$. Check if $\varphi_e(M) \downarrow$ and φ_e currently looks like a reduction on the elements $\leq M$. If not, we keep $f_{2e}(k) \uparrow$ and isolate the next element of $C_{i(2e)}$ larger than M with respect to B_0 . If the conditions are met, then our action depends on the following cases:

- *M is marked.* First of all check to see if φ_e is injective on $C_{i(2e)}$ so far, and if not, we get an easy win by isolating every element of $C_{i(2e)}$ with respect to B_0 , and do nothing else for this requirement.

Otherwise we set $B_0(f_{2e}(k-1), M) = 1 - (A \sqcup B_1)(\varphi_e(f_{2e}(k-1)), \varphi_e(M))$.

Note that we can do this because M is marked means that we have kept

$B_0(f_{2e}(k-1), M) = 1$ up till now. Initialize all lower priority requirements, and isolate all the rest of $C_{i(2e)}$ with respect to B_0 . Do nothing else for this requirement.

- *The following three conditions hold:*

- M is not marked, and
- $\varphi_e(M)$ is odd, and
- either $P \notin C_{i(e')}$ for any $e' < 2e$, or P is isolated, or $\varphi_e(M) = \varphi_e(M')$ for some $M' < M$ and $M' \in C_{i(2e)}$, and where $P = \frac{\varphi_e(M)-1}{2}$.

In this case we set $f_{2e}(k) = M$ and redefine M to be the next element of $C_{i(2e)}$ larger than $f_{2e}(k)$ and not yet isolated. We mark the new M and isolate it from every element except for $f_{2e}(k)$.

- M is not marked and $\varphi_e(M)$ is even. Then we set $f_{2e}(k) = M$ and redefine M to be the next element of $C_{i(2e)}$ larger than $f_{2e}(k)$ and not yet isolated. We update the coding of U into B_0 by setting $U_s(i, j) = B_{0,s}(f_{2e}(i), f_{2e}(j))$ for all $i, j \leq k$, and set $U_s(i, k+1) = B_{0,s}(f_{2e}(i), M) = B_{0,s}(M, f_{2e}(i))$ for all $i \leq k$ and for the new M . Again we can do this because the new value of M is B_0 -related to $f_{2e}(i)$ for all $i \leq k$ just before this step.
- *Otherwise.* We isolate M and redefine it to be the next element of $C_{i(2e)}$ not yet isolated. Follow the previous case to update the coding of U into B_0 up to and including this new M (this new M is now the intended image for $k+1$).

The basic strategy for P_{2e+1} is similar to the one for P_{2e} , except we reverse the roles of B_0 and B_1 . When a requirement is initialized, we reset the parameters f , i and M associated with the requirement. We isolate all remaining elements of the previous column C_i associated with the requirement.

Construction of B_0 and B_1 : As indicated above, at the beginning we begin by taking $B_{i,0} = \{(\langle e, x \rangle, \langle e, y \rangle) \mid e, x, y \in \omega\}$ for $i = 0, 1$. At stage $s > 0$, let all requirements R_e for $e < s$ act in order according to the basic strategy above.

Verification: We now verify the construction works. It is clear that each R_e is initialized only finitely often, since if a requirement R_e initializes a lower priority one, then R_e never acts again. We first check that the constructed objects are equivalence relations:

Lemma 2.5.2. *B_0 and B_1 are equivalence relations.*

Proof. The only way for something to go wrong is for some column to be not transitive. So let's check that every column C_k is transitive with respect to B_0 (similarly for B_1):

We initially start with the full relation on C_k . If C_k is never picked by any requirement then it stays full. Otherwise some requirement P_{2e} begins work in C_k . Then P_{2e} will assign $f_{2e}(0), f_{2e}(1), \dots$ and begin isolating every other element until initialized. Additionally it also copies $U \upharpoonright \text{dom} f_{2e}$ into $B_0 \upharpoonright \text{rng} f_{2e}$ (unless the last element is marked). Since for every m , there are cofinitely many s such that $U_s \upharpoonright m$ is transitive, we can assume that $B_0 \upharpoonright \text{rng} f_{2e}$ is always transitive.

At the end, we have the following possibilities:

- f_{2e} is total. Then C_k will contain $\text{rng} f_{2e}$ and infinitely many isolated elements, where $\text{rng} f_{2e} \equiv_c U$.
- $\text{dom} f_{2e}$ is finite, and we eventually go through infinitely many different M . Then C_k contains a (copy of a) finite initial segment of U as well as infinitely many isolated elements.
- $\text{dom} f_{2e}$ is finite, and we eventually settle on a final M , where k is the largest element of $\text{dom} f_{2e}$. If M is marked then M is in the same class as $f_{2e}(k)$ and C_k contains the finite initial segment of $U \upharpoonright k$. Otherwise if M is not marked then C_k contains the finite initial segment of $U \upharpoonright k + 1$. Finally if R_{2e} is initialized or if we act one last time under the first case of the basic strategy, then C_k is also transitive. $\square$

Lemma 2.5.3. *Each R_e acts only finitely often, that is, it extends $\text{dom} f_e$ and redefines M only finitely often.*

Proof. We proceed for R_{2e} , the other case is symmetric. Since R_{2e} is initialized finitely often, fix the final $C_{i(2e)}$ assigned to R_{2e} . Suppose R_e acts infinitely often. Hence no element is ever marked by R_{2e} .

Suppose the third case is taken infinitely often by R_{2e} . Hence f_{2e} is total. Furthermore, as R_{2e} never gets stuck waiting forever, this means that $\varphi_e \circ f_{2e}$ is total, and provides a reduction from U to $A \sqcup B_1$. Since the range of this reduction is contained in the even integers, this means that $U \leq_c A$, a contradiction.

Now suppose the fourth case is taken cofinitely often by R_{2e} . In this case we will discover, for some $e' < 2e$, an infinite sequence $M_0 < M_1 < \dots$ of elements inside $C_{i(2e)}$ such that for every $j \neq j'$,

- $\varphi_e(M_j)$ is odd,
- $\frac{\varphi_e(M_j)-1}{2} \in C_{i(e')}$,
- $\frac{\varphi_e(M_j)-1}{2}$ is picked by $R_{e'}$ to be in $\text{rng} f_{e'}$ or a value of M ,
- $\varphi_e(M_j) \neq \varphi_e(M_{j'})$.

However this is impossible since by the induction hypothesis, $R_{e'}$ only goes through finitely many different values of M and $\text{dom} f_{e'}$. $\square$

Lemma 2.5.4. *Each R_e is satisfied.*

Proof. Assume that φ_e witnesses that $B_0 \leq_c A \sqcup B_1$. By Lemma 2.5.3 R_{2e} eventually stops acting, therefore, the final action taken by R_{2e} must be under the first case. If the final action was due to the fact that we discovered that φ_e is not injective on $C_{i(2e)}$, then clearly we get a contradiction. Thus we can assume that the final action under the first case was to set $B_0(f_{2e}(k-1), M) = 1 - (A \sqcup B_1)(\varphi_e(f_{2e}(k-1)), \varphi_e(M))$. Suppose s_1 is the stage where we took the final action, and $s_0 < s_1$ was the earlier stage where R_{2e} marked M . It remains to verify that $(A \sqcup B_1)(\varphi_e(f_{2e}(k-1)), \varphi_e(M))$ will never change after s_1 , and we get the desired contradiction. This can only be possible if $(\varphi_e(f_{2e}(k-1)), \varphi_e(M)) \in A_{s_1} \sqcup B_{1,s_1}$.

At the earlier stage where we marked M we must have discovered that $\varphi_e(f_{2e,s_1}(k-1))$ is odd and that either $\frac{\varphi_e(f_{2e,s_1}(k-1))-1}{2} \notin C_{i(e')}$ for any $e' < 2e$, or $\frac{\varphi_e(f_{2e,s_1}(k-1))-1}{2}$ is isolated. If the latter holds then we must in fact have $\varphi_e(f_{2e,s_1}(k-1)) = \varphi_e(M)$.

So we may suppose that $\frac{\varphi_e(f_{2e,s_1}(k-1))-1}{2} \notin C_{i(e')}$ for any $e' < 2e$ at stage s_0 . At stage s_1 , if $v \neq i(e')$ for any e' , then C_v will never be assigned to any requirement and so is never modified by the construction, where $\frac{\varphi_e(f_{2e,s_1}(k-1))-1}{2} \in C_v$. Therefore we can assume that $v = i(e')$ at stage s_1 . Since no higher priority requirement is initialized between s_0 and s_1 , we must have $e' \geq 2e$. We do not have to worry about $e' = 2e$ because $C_{i(2e)}$ must be the full relation with respect to B_1 , since it is already associated with R_{2e} working for B_0 . Thus the last case to consider is when $e' > 2e$. However $R_{e'}$ is initialized at stage s_1 , which means that there can be no more changes to C_v with respect to B_1 after stage s_1 . $\square$

This ends the proof of Theorem 2.5.1. $\square$

It is easy to modify the proof of Theorem 2.5.1 for the other classes being studied. Therefore, we have:

Corollary 2.5.5. *The classes of Π_1^0 , Σ_n^0 , n -c.e. and ω -c.e. equivalence relations are upwards dense, for $n \geq 1$.*

Proof. In each class, there is a universal member U . We apply an appropriate version of the Sacks coding strategy, and leave the proof to the reader. $\square$

Chapter 3

SACKS JUMP INVERSION FOR THE WEAK TRUTH TABLE DEGREES AND THE BOUNDED JUMP

3.1 Introduction

3.1.1 Motivations and preliminaries

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 chapter we will look at the bounded jump and answer an open question

about jump inversion asked in [1]. 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 3.1.1 (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 [9] where they proved that the analogue of Shoenfield jump inversion fails to hold for the *wtt*-degrees and the Turing jump. 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*-) reducibility.

Indeed, by considering jump inversion for the *bounded jump* with *wtt*-reducibility, Anderson and Csima [1] were able to show that the analogue of Shoenfield jump inversion *does* hold: They showed 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 3.1.2 we answer their question by proving that this is impossible:

Theorem 3.1.2. *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 $A \not\leq_{wtt} V^b$.*

3.1.2 Organization of the chapter

The rest of this chapter is devoted to the proof of Theorem 3.1.2. In Section 3.2.1 we write down the requirements and describe the notation used in the proof.

In Section 3.2.2 we describe informally the strategy of a single requirement. In Section 3.2.3 we give the formal construction, and in Section 3.2.4 we verify that the requirements are satisfied.

3.2 The proof of Theorem 3.1.2

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

3.2.1 Requirements and notation

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, and V_e is the e -th 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 an $(\omega + 1)$ -c.e. set A and satisfy the following requirement for each e :

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

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 $M_n = \max_{0 \leq i \leq k < n} \varphi_i(k)$. Here and elsewhere whenever we take max of an expression, we range over all values of the expression which are defined. Hence $n \mapsto M_n$ 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 $\{\psi_k^n\}_{k < n^2}$ by the following. We set $\psi_k^n(z) =$ the k^{th} value in the approximation of M_n for every z . Let $i_0^n < i_1^n < \dots < i_{n^2-1}^n$ be indices for the sequence, i.e. $\varphi_{i_k^n} = \psi_k^n$, for all $k < n^2$. These functions are either constant or empty functions. In the sequel we will usually denote $m = n^2$.

During the proof we will use z_e to denote a witness used by R_e for the purpose of diagonalization, and it will toggle the value of $A(z)$ back and forth. We will also pick

a number of *subwitnesses* $\mathbf{w}_{e,\tau}$ where τ ranges over the set of all finite binary strings of a certain length. These subwitnesses are meant to assist in the diagonalization of R_e by forcing changes in V_e^b ; since we do not control V_e or V_e^b , in order for us to have any influence over V_e^b , we have to do so by enumerating Turing functionals. Thus, the subwitnesses are going to be the indices of Turing functionals built during the construction, and whose indices are provided by the Recursion Theorem. $\sigma_{e,i}$ will stand for the i^{th} finite binary string (under the lexicographic order) of length $\gamma(z_e)$.

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

Definition 3.2.1. $\sigma_{e,i}$ is said to be currently *free* if $V^b(\mathbf{w}_{e,\sigma_{e,i}}) = 0$. We write $\sigma \preceq \sigma'$ if for all w , if $\sigma(w) = 1$, then $\sigma'(w) = 1$. We write $\sigma \prec \sigma'$ if $\sigma \preceq \sigma'$ and $\sigma \neq \sigma'$.

3.2.2 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 $l_e := \max\{\delta_{e'}(\mathbf{w}_{e',\sigma_{e',i}}) \mid i < 2^{\gamma_{e'}(z_{e'})}, e' < e\}$.
- (2) Wait for $\gamma(z) \downarrow$. Then we consider the indices $i_0^{\gamma(z)} < i_1^{\gamma(z)} < \dots < i_{\gamma(z)^2}^{\gamma(z)}$.

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 $\sigma_{e,i}[s]$. We simply write σ_i , $i < 2^{\gamma(z)}$ if there is no ambiguity.

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

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

(3) We wait for $\delta(\mathbf{w}_{e,\sigma_e,i}) \downarrow$ for all $i < 2^{\gamma(z)}$, then we initialize all lower priority requirements.

(4a) Case a: If $V^b(\mathbf{w}_{\sigma_c}) = 0$, 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 $i < 2^{\gamma(z)}$:

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

This is called a *recovery for R_e* . If this happens, we are able to make $V^b(\mathbf{w}_{\sigma_c}) = 1$ by enumerating the computation $\Phi_{\mathbf{w}_{\sigma_c}}(V \upharpoonright M_{\gamma(z)}; \mathbf{w}_{\sigma_c}) \downarrow = 0$. Observe that $\mathbf{w}_{\sigma_c} > i_{\gamma(z)^2}^{\gamma(z)}$ and so the use $M_{\gamma(z)}$ is bounded by $\varphi_i(\mathbf{w}_{\sigma_c})$ for some $i < \mathbf{w}_{\sigma_c}$.

(4b) Case b: If $V^b(\mathbf{w}_{\sigma_c}) = 1$, 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 $V^b(\mathbf{w}_\tau) = 1 = \Delta((A \oplus \emptyset') \upharpoonright \delta(\mathbf{w}_\tau); \mathbf{w}_\tau)$ i.e. the value of $\Delta((A \oplus \emptyset') \upharpoonright \delta(\mathbf{w}_\tau); \mathbf{w}_\tau)$ has changed from 0 to 1. Note that this can only happen under a $(A \oplus \emptyset') \upharpoonright \delta(\mathbf{w}_\tau)$ -change; in fact, if we were careful about restraining A between Steps (4a) and (5a), we must have a $\emptyset' \upharpoonright \delta(\mathbf{w}_\tau)$ -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 $V^b(\mathbf{w}_\tau) = 0 = \Delta((A \oplus \emptyset') \upharpoonright \delta(\mathbf{w}_\tau); \mathbf{w}_\tau)$, i.e. V must have changed to now cause $V^b(\mathbf{w}_\tau) = 0$. 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 $\tau \not\leq \sigma_c$ then we can conclude that τ is now free. (See Lemma 3.2.3.)

(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).

3.2.3 Construction

We apply the Recursion Theorem to obtain an effective list of indices of Turing functionals, and we will pick the subwitnesses $\mathbf{w}_\sigma$ for the different requirements in the construction from this list. These are Turing functionals which we build during the construction, and whose indices are given to us during the construction by the Recursion Theorem. 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 has not.

Action: Act as in (1) of the basic strategy.

(S2) The last time R_e requires 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 requires attention in (S2), $\delta_{e,s}(\mathbf{w}_{e,\sigma_{e,i}}) \downarrow$ 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 requires attention in (S4) or (S5), R_e has recovered, and $V^b(\mathbf{w}_{\sigma_c}) = 0$.

Action: Act as in (4a) of the basic strategy.

(S5) The last time R_e requires attention in (S4) or (S5), R_e has recovered, and $V^b(\mathbf{w}_{\sigma_c}) = 1$.

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, by resetting all parameters (the witnesses and subwitnesses) associated with the requirement.

3.2.4 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 3.2.2. *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 $V^b(\mathbf{w}_\tau)[s] = 1$, 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 $V^b(\mathbf{w}_\tau)[s] = 1$ with use $V \upharpoonright M[t]$. However, between t and s , V must change below $M[t]$, which contradicts $V^b(\mathbf{w}_\tau)[s] = 1$. 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' . $\square$

Lemma 3.2.3. *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 3.2.2. $\square$

Lemma 3.2.4. *The number of non-free $\sigma_{e,i}$ between two consecutive toggling of z_e will decrease by 1 if there is no $\emptyset'$ -change below $\max \{ \delta_e(\mathbf{w}_{e,\sigma_{e,i}}) \mid i < 2^{\gamma_e(z_e)} \}$.*

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) $\rightarrow$ (S4) $\rightarrow \dots \rightarrow$

(S4) $\rightarrow$ (S5), or (S5) $\rightarrow$ (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:

(S4) $\rightarrow$ (S4): Suppose the strategy for R_e acts under (S4) at stage s and then under (S4) 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 $\delta(\mathbf{w}_{\sigma_c^e[s]})$. 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.

(S4) $\rightarrow$ (S5): 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).

(S5) $\rightarrow$ (S4): 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).

(S5) $\rightarrow$ (S5): 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 3.2.2 $\sigma_c^e[s']$ is free at the beginning of s' , a contradiction.

This means that $\sigma_c^e[s] \not\leq \sigma_c^e[s']$. Since $\sigma_c^e[s]$ is not free at the beginning of s , by Lemma 3.2.3, the number of free strings must decrease. $\square$

Lemma 3.2.5. *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 $m_e = \max \{ \delta_e(\mathbf{w}_{e,\sigma_{e,i}}) \mid i < 2^{\gamma_e(z_e)} \}$.*

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 3.2.4 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 3.2.6. *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 are found. Thereafter, by Lemma 3.2.5, 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 3.2.7. *All requirements R_e are satisfied.*

Proof. By Lemma 3.2.5, 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 $\delta_e(\mathbf{w}_{e,\sigma_{e,i}}) \uparrow$ 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 $A \neq \Gamma^{V^b}$. $\square$

Chapter 4

CHARACTERIZING THE C.E. BOUNDED HIGH SETS

4.1 Introduction

4.1.1 Motivations and preliminaries

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 are 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 4.3.1. *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).

4.1.2 Organization of the chapter

The chapter is organized as follows. In Section 4.2 we give a different proof of the very useful fact that $\emptyset^{2b}$ is effectively 1-complete amongst the ω^2 -c.e. sets. In Section 4.3 we state and prove the characterization we obtain of the c.e. bounded high sets.

4.2 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 an ω^2 -c.e. *index* if $\varphi_e(x, s)$ and $\varphi_k(x, s)$ are total computable functions and provide an enumeration of an 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 $\varphi_k(n, s)$ codes an ordinal greater than $\varphi_k(n, s+1)$. We also require $\varphi_k(n, 0)$ to code an ordinal less than ω^2 , be non-increasing in s , and where $\text{rng}(\varphi_k)$ is a set of strong ordinal notations for ordinals less than ω^2 . Recall that a strong ordinal notation is one where the Cantor Normal form of the ordinal can be effectively computed.

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

Proposition 4.2.1 (Anderson, Csima). *There is a total computable function p such that for every $\langle e, k \rangle$, if $\langle e, k \rangle$ is an 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

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

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

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

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 functions

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

Here 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 in ordinal arithmetic. The idea is that s is least such that $\varphi_k(n, s)$ has decreased by at least c many limit ordinals compared to the starting ordinal value $\varphi_k(n, 0)$. 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. We illustrate the parameters C and c with an example. Suppose $\varphi_k(n, 0) = \omega \cdot 3 + 5$, and the changes to the values of $\varphi_k(n, -)$ are denoted by the sequence $\omega \cdot 3 + 5, \omega \cdot 3 + 4, \omega \cdot 2 + 10, \omega \cdot 2 + 9, \omega + 15, \omega + 13, 20, 5, 0$. Then when $c = 0$, the corresponding s , C' and C would be 0, 5, and 5 respectively. When $c = 1$, the corresponding s would be the least stage such that $\varphi_k(n, s) = \omega \cdot 2 + 10$, the corresponding C' would be 10, and the corresponding C would be something larger than $5 + 10$. When $c = 2$, the corresponding C' would be 15 and the corresponding C is larger than $5 + 10 + 15$.

Finally, define the sequence of Turing functionals

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

by the following. For each minimal string σ where $\sigma(\langle j(e, n), 0 \rangle) = \sigma(\langle j(e, n), 1 \rangle) = \dots = \sigma(\langle j(e, n), z-1 \rangle) = 1$ and $\sigma(\langle j(e, n), z \rangle) = 0$ for some z , we search for the least s such that $N(e, n, s) = z-1$. (Note that if $\sigma \subset \emptyset'$, then the above condition on σ is equivalent to $0, 1, \dots, z-1 \in W_{j(e, n)}$ and $z \notin W_{j(e, n)}$).

Define $\Phi_{r(e, n, d)}^\sigma(x) \downarrow$ 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 $\varphi_k(n, 0) <_{\mathcal{O}} \omega \cdot d$ for a large enough d . Notice also that d is not actively being used in the definition of the functional $\Phi_{r(e, n, d)}^\sigma(x)$; it's role is simply to ensure that a "large enough" $r(e, n, -)$ can be picked.

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 $\Phi_{r(e, n, d)}^{\emptyset'}(r(e, n, d)) \downarrow$ 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 $\sigma = \emptyset' \upharpoonright \langle j(e, n), z \rangle + 1$, the construction ensures that we define $\Phi_{r(e, n, d)}^\sigma(r(e, n, d)) \downarrow$. Since $z-1$ is the total number of changes in $\varphi_e(n, -)$, it follows that $\langle j(e, n), (z-1) + 2 \rangle < \max\{\varphi_{i(e, k, n, 0)}, \varphi_{i(e, k, n, 1)}, \dots, \varphi_{i(e, k, n, d)}\}$. (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 $\emptyset'(\langle j(e, n), 0 \rangle) = \emptyset'(\langle j(e, n), 1 \rangle) = \dots = \emptyset'(\langle j(e, n), z-1 \rangle) = 1$ and $\emptyset'(\langle j(e, n), z \rangle) = 0$. If $\Phi_{r(e, n, d)}^{\emptyset'}(r(e, n, d)) \downarrow$ then there is some minimal $\sigma \subset \emptyset'$ that corresponds to the construction. This means that $\sigma = \emptyset' \upharpoonright \langle j(e, n), z \rangle + 1$, but this is a con-

tradiction because $\varphi_e(n, s) = 0$ for the least s where the value is stable. So, $\Phi_{r(e, n, d)}^{\emptyset'}(r(e, n, d)) \uparrow$. $\square$

4.3 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 4.3.1. *The following are equivalent for a c.e. set A .*

- (i) A is bounded high, i.e. $A^b \geq_{\text{wtt}} \emptyset^{2b}$.
- (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 4.3.1.

Lemma 4.3.2. *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 3.2.1, we define the computable sequence of indices $\{i_k^n\}$ such that each $\varphi_{i_k^n}$ is either a total constant function, or the empty function, and such that $\varphi_{i_k^n}(z) =$ the $k+1^{\text{th}}$ 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

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

We pick $f(n) > i_k^n$ 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 $\Phi_{f(n)}^A(f(n)) \downarrow$, with use bounded by $\delta(n)$. But

$\delta(n) = \varphi_{i_k^n}(f(n))$ for some $k < h(n)$, which means that $i_k^n < f(n)$. Therefore $f(n) \in A^b$. Now suppose that $\emptyset^{2b}(n) = 0$. This means that $\Delta^A(n) = 0$ which means that $\Phi_{f(n)}^A(f(n)) \uparrow$. So $f(n) \notin A^b$. $\square$

4.3.1 Proof of (i) $\Rightarrow$ (iii)

Notation and conventions

The rest of this section will be devoted to the proof of (i) $\Rightarrow$ (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. Note that the Recursion Theorem is applied to produce the indices e, k where $\langle e, k \rangle$ is an ω^2 -c.e. index for F ; on the other hand, during the construction, when we describe an ω^2 -bounded enumeration of F , it is easy to translate our informal description of the enumeration into a pair of ω^2 -c.e. index for F .

Thus by Proposition 4.2.1 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 boundedly 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 $M_n = \max_{0 \leq i \leq k < n} \varphi_i(k)$. 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. 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 not, 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.

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 3.2.1), 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 $F(n)$.

Verification

We now verify that the construction works.

Lemma 4.3.3. *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 $A^b[s+1] \upharpoonright \psi(n) \preceq A^b[t] \upharpoonright \psi(n)$; 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 4.3.4. $\emptyset^{2b} \leq_T^{\omega\text{-c.e.}} A$.

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 4.3.3. □

This concludes the proof of $(i) \Rightarrow (iii)$ and of Theorem 4.3.1.

Chapter 5

EFFECTIVIZING DOMINATION PROPERTIES

5.1 Introduction

5.1.1 Motivations and preliminaries

The purpose of this chapter is to study the effects strong reducibilities have on classical computability concepts. In this chapter we look at a well-known classical computability notion - domination - and ask to what extent strong reducibilities have on domination and its 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 [27]. 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 [13]). A related notion is that of non-totally ω -c.e. (see Downey, Greenberg and Weber [12]). These classes relate the computational power of an oracle A needed to carry out certain constructions with the growth rates of A -computation functions. Recent work of Ambos-Spies and Losert study these in greater detail. Downey and Greenberg generalized this and defined a hierarchy of Turing degrees by measuring the amount of multiple permitting strength present; for this we refer the reader to the monograph [11] or the survey paper [10].

Domination properties are also intimately related to the jump classes. Recall that a set A (or its Turing degree) is low_n if $A^{(n)} \leq_T \emptyset^{(n)}$, and is high_n if $A^{(n)} \geq_T \emptyset^{(n+1)}$. For a Δ_2^0 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 is one of the standard ways of measuring the computational strength of an oracle. For instance, a Δ_2^0 Turing degree $\mathbf{a} \leq \mathbf{0}'$ is high if $\mathbf{a}' = \mathbf{0}''$, that is, $\mathbf{a}'$ is greatest possible. This property implies that a high degree is computationally complex, and that $\mathbf{a}$ carries a lot of information similar to $\mathbf{0}'$. A high c.e. degree can be seen as being computationally complex in the sense 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 was formalized by Martin, who showed that a Turing degree $\mathbf{a}$ is high iff $\mathbf{a}$ is *dominant*, that is, $\mathbf{a} \geq \deg_T(f)$ 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 [21] 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. A first 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_2^0 . 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_3^0 or even $g \leq_T f$, then the induced notions are simply equivalent to being the usual notion of high.

The notion above was defined and studied in [14], where the authors called it *strongly dominant*. They proved that a Δ_2^0 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 5.1.1. 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 $\mathcal{C} = \Delta_{2,\omega}^0$ -c.e., or Δ_1^0 , 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 $\leq_T$. In this chapter 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?

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 $(\leq_{wtt}, \Delta_1^0)$ -dominant and being $(\leq_{wtt}, \omega\text{-c.e.})$ -dominant. We will also separate the various notions of effective domination. Our results in this chapter are summarized in Figure 5.1.

We first show that being $(\leq_T, \mathcal{C})$ -dominant is the same as being Turing complete for $\mathcal{C} = \Delta_2^0, \omega\text{-c.e.}$, or Δ_1^0 , and that considering the graph of a function is also equivalent to being Turing complete. We next establish all the non-trivial implications and non-implications in Figure 5.1.

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

Figure 5.1: 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. $\mathcal{C}$ can be taken to be the class of Δ_1^0 , ω -c.e., or Δ_2^0 functions in the bottom right entry.

5.1.2 Organization of the chapter

The rest of this chapter is devoted to showing the non-trivial implications and non-implications in Figure 5.1.

5.2 Results and separations

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 will adopt when we say $f \leq_r A$. The second way is to declare that $\text{Graph}(f) \leq_r A$. If $r = T$ then both interpretations are equivalent. However if $\leq_r$ is a strong reducibility, we have to be careful. In Theorem 5.2.1 we will show that the second interpretation gives nothing new for $\text{Graph}(f) \leq_{wtt} A$ and $\text{Graph}(f) \leq_{tt} A$. This justifies our use of the first definition when defining the different variations on effective domination.

Theorem 5.2.1. *Let A be a Δ_2^0 set. The following are equivalent.*

- (i) $A \equiv_T \emptyset'$.
- (ii) $\text{Graph}(f) \leq_{\text{wtt}} A$ for some strongly dominant f , i.e. f is dominant with a Δ_2^0 function g .
- (iii) $\text{Graph}(f) \leq_{\text{tt}} A$ for some f where f is dominant with a computable function g .
- (iv) A is $(\leq_T, \Delta_2^0)$ -dominant.
- (v) A is $(\leq_T, \omega\text{-c.e.})$ -dominant.
- (vi) A is $(\leq_T, \Delta_1^0)$ -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 [14, Theorem 4.8]. (iii) $\Rightarrow$ (ii) $\Rightarrow$ (iv) are trivial. (vi) $\Rightarrow$ (v) $\Rightarrow$ (iv) $\Rightarrow$ (vii) are also 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 $\Phi^A = \text{Graph}(f)$ 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 upper bound 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 5.2.2. *Let A be a Δ_2^0 set. Then A is $(\leq_{wtt}, \Delta_1^0)$ -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 . $\square$

In contrast to Proposition 5.2.2, we will later show (in Theorem 5.2.6) that we

can have a wtt -incomplete $(\leq_{wtt}, \omega\text{-c.e.})$ -dominant set. Nevertheless, together with c.e., $(\leq_{wtt}, \omega\text{-c.e.})$ -dominance does imply a stronger property than being Turing complete:

Proposition 5.2.3. *Let A be a c.e. set which is $(\leq_{wtt}, \omega\text{-c.e.})$ -dominant. Then $A \geq_T^{\omega\text{-c.e.}} \emptyset'$.*

Proof. This proposition is an effective version of [14, 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 [14]. 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 $\{I(x, n)\}$ 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 , $I(x, n)$ will be increasing in n , and will only be defined for finitely many n . We now describe how to define $I(x, n)$ for a fixed x . At stage 0, define $I(x, 0) = g(q(x))[0]$ and set $\varphi_{q(x)}(y) = 0$ for all $y \leq I(x, 0)$. Now assume that $I(x, n) \geq g(q(x))[t_n]$ is defined and $\varphi_{q(x)}(y)$ are defined for all $y \leq I(x, n)$ 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 $y = I(x, n) + 1, I(x, n) + 2, \dots$, one at a time, until we find some stage $t_{n+1} > t$ such that either A changes below $\psi(\max\{I(x, n), g(q(x))[t]\} + 1)$ 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 $\varphi_{q(x)}(\max\{I(x, n), g(q(x))[t]\} + 1) = t > \Psi^A(\max\{I(x, n), g(q(x))[t]\} + 1)[t]$. Once t_{n+1} is found we stop extending the domain of $\varphi_{q(x)}$ and set $I(x, n+1) = \max \text{dom}(\varphi_{q(x)})$ or $g(q(x))[t_{n+1}]$, whichever is larger. If $g(q(x))[t_{n+1}] > \max \text{dom}(\varphi_{q(x)})$ we extend $\text{dom}(\varphi_{q(x)})$ to make them

equal.

First observe that for each x , $I(x, n)$ is defined for at most $1 + \#$ changes in $g(q(x + 1))$ many n . Now define

$$\delta(x) = \max_{y \leq x} \max_n \psi(I(y, n) + 1) + \max_{y \leq x} \max_s \psi(g(q(y))[s] + 1).$$

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 $A_s \upharpoonright \delta_s(x) = A \upharpoonright \delta_s(x)$. We first claim that $\delta_s(x) = \delta(x)$. Suppose not. Fix the least $y \leq x$ with some least n such that $I(y, n + 1)$ 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 $I(z - 1, m)$).

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 $\psi(\max\{I(y, n), g(q(y))[t]\} + 1)$ 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 $I(y - 1, m)$ after stage $t > s$, contradicting the minimality of y . Thus we may assume that A changes below $\psi(\max\{I(y, n), g(q(y))[t]\} + 1)$ after stage t . Again by the minimality of y , we have $g(q(y))[t] = g(q(y))[s]$. This means that $\psi(\max\{I(y, n), g(q(y))[t]\} + 1) \leq \delta_s(x)$, 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 $I(x, m)$ after stage s . $\square$

Theorem 5.2.4. *There exists a c.e. set A such that $A \geq_T^{\omega\text{-c.e.}} \emptyset'$ and A is not even $(\leq_{wtt}, \Delta_2^0)$ -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 : If Ψ_e^A is total and $\lim_s g_e(k, s)$ exists for every k then there is k such that φ_k is total and $\varphi_k(x) > \Psi_e^A(x)$ for some $x > \lim_s g_e(k, s)$.

Here Ψ_e^A is the e^{th} partial *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 $N_{e,s}$ 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 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 $\delta_s(x) < \max_x \psi_e(z_e)$ for some least $x > N_e$, 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 $N_e < x$, and hence $e < x$. (Another possibility is due to the coding of $\emptyset'$, when some $y \leq x$ enters $\emptyset'$, but this happens at most $x+1$ times). If $\delta(x)$ is increased when acting for R_e , then the same R_e can do this again only if either 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 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, $N_{e,s}$ is eventually stable. Since $\delta(x)$ will eventually settle down for all $x \leq N_e$, 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 5.2.5. *Each bounded high c.e. set is $(\leq_{wtt}, \omega\text{-c.e.})$ -dominant.*

Proof. Let A be c.e. and bounded high. In Theorem 4.3.1 it was shown that this is equivalent to $\emptyset^{2b} \leq_T^{\omega\text{-c.e.}} A$. Proposition 4.2.1 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 $F(\langle e, x \rangle, 0) = 0$ 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 $F(\langle e, x \rangle, s+1) = 1 - F(\langle e, x \rangle, s)$), 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 an 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 . $\square$

Theorem 5.2.6. *There exists a $(\leq_{wtt}, \omega\text{-c.e.})$ -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:

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

These requirements are sufficient by Proposition 4.2.1 and Theorem 4.3.1: As $\emptyset^{2b}$ is 1-complete amongst the ω^2 -c.e. sets, the N -requirements ensure that $A \not\leq_T^{\omega\text{-c.e.}} \emptyset^{2b}$, and therefore A is not bounded high.

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 .

Without loss of generality we assume that $\text{dom}(\varphi_e)$ is an initial segment of ω at every stage. We arrange the requirements $N_0 < P_0 < N_1 < \dots$. 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 $s > 2s^-$, $\max \text{dom}(\varphi_e) > s^-$ and for every $k \leq s^-$ where N_k has acted before, we want $\max \text{dom}(\varphi_e) > \psi_k(k, s)$. 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 $I_x = [2^x, 2^{x+1})$ 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 $g(e)[s] < x \leq \max \text{dom}(\varphi_e)$ 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 $\Psi_e^A = F$ when ρ_e is 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 $\max \text{dom}(\varphi_k) > \psi_e(e, s_0)$. 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 now 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 5.2.7. *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 $I_x = [2^x, 2^{x+1})$ 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 , as 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 as 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 down 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 $F(z_e) \neq \Psi_e^A(z_e)$. $\square$

As a corollary to Theorem 4.3.1 and Proposition 5.2.7 we obtain [2, Theorem 3.2] that the analogue of the jump theorem fails for the bounded jump:

Corollary 5.2.8 (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 5.2.7. By Theorem 4.3.1, $B^b \geq_1 A^b$. $\square$

Theorem 5.2.9. *There exists a $(\leq_{wtt}, \Delta_2^0)$ -dominant c.e. set A such that $A \not\leq_T^{\omega\text{-c.e.}} \emptyset'$.*

Proof. The proof is very similar to the proof of Theorem 5.2.6. We follow the proof of Theorem 5.2.6 closely. We shall build a c.e. set A , an ω -c.e. set F , a 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 5.2.6, 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 using a single follower, N_e will require $e + 1$ many followers. These will be $\langle e, d_0 \rangle, \langle e, d_1 \rangle, \dots, \langle e, d_e \rangle$, where $d_0 = 0$ and $d_{i+1} = \rho_e(\langle e, d_i \rangle)$ for all $i < e$.

We arrange the requirements $N_0 < P_0 < N_1 < \dots$. 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 $s > 2s^-$, $\max \text{dom}(\varphi_e) > s^-$ and for every $k \leq s^-$ where N_k has acted before, we want $\max \text{dom}(\varphi_e) > \psi_k(\langle k, d_k \rangle, s)$. 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 $I_x = [2^x, 2^{x+1})$ 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 $g(e)[s] < x \leq \max \text{dom}(\varphi_e)$ 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_2^0 . 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 $x > g(e)$. $\square$

Bibliography

- [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] U. Andrews, S. Lempp, J. Miller, and K.M. Ng. Universal computably enumerable equivalence relations. *Journal of Symbolic Logic*, 79(1):60 – 88, 2014.
- [4] C. Bernardi and A. Sorbi. Classifying positive equivalence relations. *Journal of Symbolic Logic*, 48(3):529–538, 1983.
- [5] W. Calvert, D. Cummins, J.F. Knight, and S. Miller. Comparing classes of finite structures. *Algebra and Logic*, 43(6):374–392, 2004.
- [6] W. Calvert and J.F. Knight. Classification from a computable viewpoint. *Bulletin of Symbolic Logic*, 12(2):191–218, 2006.
- [7] B. Cooper. *Computability Theory*. Chapman and Hall/CRC, 2004.
- [8] S. Coskey, J.D. Hamkins, and R. Miller. The hierarchy of equivalence relations on the natural numbers under computable reducibility. *Computability*, 1(1):15–38, 2012.
- [9] B. Csima, R. Downey, and K.M. Ng. Limits on jump inversion for strong reducibilities. *Journal of Symbolic Logic*, 76(4):1287–1296, 2011.

- [10] R. Downey and N. Greenberg. A hierarchy of computably enumerable degrees. Submitted.
- [11] R. Downey and N. Greenberg. A transfinite hierarchy of lowness notions in the computably enumerable degrees, unifying classes, and natural definability. Submitted.
- [12] R. Downey, N. Greenberg, and R. Weber. Totally ω -computably enumerable degrees and bounding critical triples. *Journal of Mathematical Logic*, 7:145–171, 2007.
- [13] 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.
- [14] 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.
- [15] Yu. L. Ershov. Theory of numberings. 1977.
- [16] E.B. Fokina and S.-D. Friedman. Equivalence relations on classes of computable structures. In *In Proceedings of the 5th Conference on Computability in Europe: Mathematical Theory and Computational Practice*, Springer-Verlag.
- [17] R. Friedberg. A criterion for completeness of degrees of unsolvability. *Journal of Symbolic Logic*, 22:159–160, 1958.
- [18] R. Friedberg. Three theorems on recursive enumeration: I. Decomposition, II. Maximal Set, III. Enumeration without duplication. *Journal of Symbolic Logic*, 23:309–316, 1958.
- [19] H. Friedman and L. Stanley. A Borel reducibility theory for classes of countable structures. *Journal of Symbolic Logic*, 54(3):894–914, 1989.
- [20] S. Gao and P. Gerdes. Computably enumerable equivalence relations. *Studia Logica*, 67(1):27–59, 2001.

- [21] Kenneth Harris. *The complexity of classical theorems on saturated models*. PhD thesis, University of Chicago, 2007.
- [22] E. Iakovski, R. Miller, K.M. Ng, and A. Nies. Complexity of equivalence relations and preorders from computability theory. *Journal of Symbolic Logic*, 79(3):859, 2014.
- [23] S. Kleene. On notation for ordinal numbers. *Journal of Symbolic Logic*, 3:150–155, 1938.
- [24] S. Kleene and E. Post. The uppersemilattice of degrees of recursive unsolvability. *Annals of Mathematics*, 59:379–407, 1954.
- [25] A. H. Lachlan. Initial segments of one-one degrees. *Pacific Journal of Mathematics*, 29:351–366, 1969.
- [26] R. Miller and K.M. Ng. Finitary reducibilities on equivalence relations. *Journal of Symbolic Logic*, 81(4):1225–1254, 2016.
- [27] K.M. Ng, F. Stephan, Y. Yang, and L. Yu. The computational aspects of hyperimmune-free degrees. *Proceedings of the 12th Asian Logic Conference*, pages 271–284, 2012.
- [28] G. Sacks. *Degrees of Unsolvability*. Princeton University Press, Princeton, N.J., 1963.
- [29] J. Shoenfield. On degrees of unsolvability. *Annals of Mathematics*, 69:644–653, 1959.
- [30] 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.
- [31] P. R. Young. Notes on the structure of recursively enumerable sets. *Notices of the American Mathematical Society*, 10:586, 1963.