The $\forall\exists$ theory of $\mathcal{D}(\leq, \vee, ')$ is undecidable

Richard A. Shore*
Department of Mathematics
Cornell University
Ithaca NY 14853

Theodore A. Slaman†
Department of Mathematics
University of California, Berkeley
Berkeley CA 94720

November 30, 2003

Abstract

We prove that the two quantifier theory of the Turing degrees with order, join and jump is undecidable.

1 Introduction

Our interest here is in the structure $\mathcal{D}$ of all the Turing degrees but much of the motivation and setting is shared by work on other degree structures as well. In particular, Miller, Nies and Shore [2004] provides an undecidability result for the r.e. degrees $\mathcal{R}$ at the two quantifier level with added function symbols as we do here for $\mathcal{D}$. To set the stage, we begin then with an adaptation of the introduction from that paper including statements of some of its results and so discuss $\mathcal{R}$, the r.e. degrees and $\mathcal{D}(\leq \mathbf{0}')$, the degrees below $\mathbf{0}'$ as well.

A major theme in the study of degree structures of all types has been the question of the decidability or undecidability of their theories. This is a natural and fundamental question that is an important goal in the analysis of these structures. It also serves as a guide and organizational principle for the development of construction techniques and algebraic information about the structures. A decision procedure implies (and requires) a full understanding and control of the first order properties of a structure. Undecidability results typically require and imply some measure of complexity and coding in the structure. Once a structure has been proven undecidable, it is natural to try to determine both the extent and source of the complexity. On the one hand, one wants to determine

*Partially supported by NSF Grant DMS-0100035. Thanks also to Andre Nies for some helpful conversations about this work.

†Partially supported by NSF Grant DMS-9988644.

the degree of the theory. On the other hand, one strives to find the dividing line between decidability and undecidability in terms of fragments of the theory. The first has frequently brought with it considerable information about second order properties such as definability and automorphisms. The second requires the most algebraic information and development of construction techniques.

For $\mathcal{D}$ and $\mathcal{D}(\leq \mathbf{0}')$ the results came fairly early. The first paper on the structure $\mathcal{D}$ of the Turing degrees as a whole, Kleene-Post [1954], developed the finite extension method (essentially Cohen forcing for one quantifier formulas of arithmetic) and proved that all finite partial orderings can be embedded in both $\mathcal{D}$ and $\mathcal{D}(\leq \mathbf{0}')$. As these structures are partial orderings, this suffices to show that the one quantifier ($\exists$) theories are decidable. (An existential sentence is true in either structure if and only if it is consistent with the theory of partial orders, or equivalently, if there is a partial order with a domain of size the number of variables in the formula which satisfies the formula.)

Once the embedding problem is settled, the next level of algebraic questions about the structures concern extension of embeddings. The first example here is density (or, from the other side minimal covers). A long development of construction techniques building on Spector's original construction [1956] of a minimal degree essentially by forcing with recursive trees lead to Lachlan's [1968] result that every countable distributive lattice is isomorphic to an initial segment of $\mathcal{D}$. This coding of distributive lattices is sufficient to get the undecidability of the theory as Lachlan [1968] notes. Combining these initial segment techniques with simultaneous control of the join and Spector's [1956] exact pair theorem, Simpson [1977] showed that the theory of $\mathcal{D}$ is recursively isomorphic to $Th^2(\mathbb{N})$, true second order arithmetic.

Finding the dividing line between decidability and undecidability required Lerman's [1971] result that every finite lattice (not just the distributive ones) is isomorphic to an initial segment of $\mathcal{D}$. On one hand, combining this with the finite extension method solved the extension of embedding problem in such a way that it gave the decidability of the two quantifier ($\forall\exists$) theory of $\mathcal{D}$ (Shore [1978] and Lerman [see 1983, Appendix A]). (By the extension of embedding problem we mean determining for which partial orders $\mathcal{X} \subseteq \mathcal{Y}$ does every embedding of $\mathcal{X}$ into $\mathcal{D}$ have an extension to one of $\mathcal{Y}$.) The ability to code all finite lattices also sufficed for Schmerl (see Lerman [1983, Appendix A]) to prove that the three quantifier ($\forall\exists\forall$) theory of $\mathcal{D}$ is undecidable.

A similar analysis of $\mathcal{D}(\leq 0')$ was then carried out. First came a significant elaboration of the construction techniques to get enough initial segments results below $0'$ to give undecidability (Epstein [1979] and Lerman). Lerman then proved the full analog that every finite (even recursive) lattice is isomorphic to an initial segment of $\mathcal{D}(\leq 0')$ (Lerman [1983, XII]). This immediately gives the undecidability of the three quantifier theory. Then these results were extended and combined with extension of embedding results below an arbitrary r.e. degree (Lerman and Shore [1988]) to get the decidability of the two quantifier theory. They were also used to show (Shore [1981]) that the theory of $\mathcal{D}(\leq 0')$ is recursively isomorphic to true first order arithmetic.

The road has been much longer for the analysis of the r.e. degrees, $\mathcal{R}$. It began with the finite injury (or $0'$) priority method of Friedberg [1957] and Muchnik [1956] that produced incomparable r.e. degrees and so an embedding of the simplest partial (nonlinear) order. This method sufficed to embed all finite (even countable) partial orderings (Sacks [1963]) and so decide the one quantifier theory of $\mathcal{R}$ in the same way that Kleene and Post's work decided that of $\mathcal{D}$ and $\mathcal{D}(\leq 0')$. As the r.e. degrees are dense (by the infinite injury (or $0''$) methods of Sacks [1964]), the next steps in the analysis could not follow the path laid out for $\mathcal{D}$. Many years of development of construction techniques and algebraic information ensued. Lachlan's monster (or $0'''$ injury) methods were eventually used by Harrington and Shelah [1982] to prove that $\mathcal{R}$ is undecidable. The degree of its theory, as by now one should expect, is also that of true first order arithmetic (Harrington and Slaman; Slaman and Woodin; Nies, Shore and Slaman [1998]).

This leaves us with determining the boundary line between decidability and undecidability for $\mathcal{R}$. Once again, a long hiatus and much work on other developments led to the undecidability of the three quantifier theory by Lempp, Nies and Slaman [1998]. The extension of embedding problem was solved by Slaman and Soare [2001] but the question of the decidability of the two quantifier theory of $\mathcal{R}$ remains open. A major obstacle is the lattice embedding problem of determining which finite lattices can be embedded in $\mathcal{R}$. Despite some forty years of effort by many researchers on both embedding and nonembedding results, this question is still unsolved. The best result to date is Lerman [2000] which shows that the question for an important class of lattices is of degree at most $0''$. Even if the lattice embedding problem is shown to be decidable, there are further difficulties related to Lachlan's [1966] nondiamond result that there is no embedding of the four element Boolean algebra into $\mathcal{R}$ that preserves both 0 and 1.

The situation for these three degree structures is summarized in the following table:

	$\mathcal{R}$	$\mathcal{D}$	$\mathcal{D}(\leq \mathbf{0}')$
$\exists(\leq)$	Dec	Dec	Dec
$\forall\exists(\leq)$	?	Dec	Dec
$\forall\exists\forall(\leq)$	Undec	Undec	Undec
$Th(\leq)$	$Th(N)$	$Th^2(N)$	$Th(N)$

Thus we remain a long way from the decidability of the two quantifier theory of $\mathcal{R}$. On the other hand, the methods used to prove undecidability of other degree structures, interpretation of theories with simple fragments known to be undecidable, cannot work for the two quantifier theory of $\mathcal{R}$ with just $\leq_T$, or even any extension by relation symbols, since the most we can code into this fragment is the validity (perhaps in all finite models) of an $\forall\exists$ sentence in a finite relational language but this problem is always decidable. (The point here is that, since the language is relational, any such sentence with n variables is satisfiable if and only if it is satisfiable in some structure of size at most n. As there are only finitely many such structures, this question is decidable. The basic result is classical

(Bernays and Schönfikel [1928] and Ramsey [1930]). Its application to interpretations in structures such as $\mathcal{R}$ is pointed out in Shore [1999, p. 179].)

The only hope for an undecidability result at the two quantifier level for $\mathcal{R}$ then is to add function symbols. One would then try to interpret some theory with function symbols or, more directly, to code register machines. (The coding of register machines is at the basis of much of the work on undecidability of various severely restricted quantification classes of formulas as in Börger, Grädel and Gurevich [1997].) This raises the natural question about the boundary between decidability and undecidability for all these degree structures: What happens when we add additional function symbols to the language?

In all these settings the most natural one to be considered is the join operator $\vee$. As the structures remain uniformly locally finite the arguments for the unlikeliness of interpretations of theories providing undecidability remain in place. (The closure of any finite set is finite with size bounded by a fixed recursive function of the cardinality of the starting set and so cannot, on its own, be used to generate the infinite (or at least unbounded) structures need for coding even register machines.) Indeed, for all the degrees, the $\forall\exists$ theory of this structure, $\mathcal{D}(\leq, \vee)$, is decidable by Jockusch and Slaman [1993].

The next thing to try in terms of the known structural work on $\mathcal{R}$ is the infimum operator $\wedge$. This has the advantage that finitely generated substructures can be infinite (Lerman, Shore and Soare [1984]). The obvious problem with this approach is that not every pair of r.e. degrees has an infimum and so $\wedge$ is not a total function on $\mathcal{R}$ as is required. We can, of course, consider total extensions of the partial infimum relation but would not want the undecidability to be an artifact of our (perhaps perverse) choice of extension. The solution of Miller, Nies and Shore is to prove undecidability in a sufficiently uniform way so that the proof is independent of the choice of extension.

Theorem 1.1 *(Miller, Nies and Shore [2004]) For any total extension $\wedge$ of the partial infimum relation on $\mathcal{R}$, the two quantifier ($\forall\exists$) theory of $\mathcal{R}(\leq, \vee, \wedge)$ is undecidable.*

They noted that the coding methods used for this result can be applied to both $\mathcal{D}$ and $\mathcal{D}(\leq \mathbf{0}')$ along with known initial segment constructions to get similar results.

Corollary 1.2 *(Miller, Nies and Shore [2004]) For any total extension $\wedge$ of the partial infimum relation on $\mathcal{D}$ ($\mathcal{D}(\leq 0')$), the two quantifier ($\forall\exists$) theory of $\mathcal{D}$ ($\mathcal{D}(\leq 0')$) with $\leq$, $\vee$ and $\wedge$ is undecidable.*

In the setting of the degrees as a whole, however, there is a second natural operator to consider adding to our language, the jump operator. The jump is definable in $\mathcal{D}$ by Shore and Slaman [1999] but the definition involves coding models of arithmetic and discussing automorphisms. Its complexity is very high and so sheds no light on the boundary between decidability and undecidability in $\mathcal{D}(\leq,')$, the degrees with jump which is our topic here.

Decidability results to date on the theory with the jump operator include the following:

Theorem 1.3 *(Jockusch and Soare, see Lerman [1983] III.4.21) The theory of $\mathcal{D}$ with just $'$ (no $\leq$) is decidable.*

Theorem 1.4 *(Hinman and Slaman [1991]): The $\exists$-theory of $\mathcal{D}(\leq,')$ is decidable.*

Theorem 1.5 *(Montalban [2003]): The $\exists$-theory of $\mathcal{D}(\leq,\vee,')$ is decidable.*

We show that Montalban's decidability result is the best possible (actually, our result was proven first so he provided the proof of sharpness for our result) and so solve problem IV.7 (attributed to Jockusch) of Arslanov and Lempp [1999]:

Theorem 1.6 *The $\forall\exists$-theory of $\mathcal{D}(\leq,\vee,')$ is undecidable.*

Our route to undecidability is via the coding of register machines as in Miller, Nies and Shore [2004]. We describe the machines and their coding in predicate logic in the next section. Once we see how they are interpreted in predicate logic it will be clear that our undecidability result would follow immediately from the existence of a Δ_0 formula with x free and additional free variables such that, as the additional variables range over the degrees, the sets defined by the formula range over all countable subsets of $\mathcal{D}$. This result is provided by our main technical theorem.

Theorem 1.7 *Given $\{\mathbf{y}_i|i\in\omega\}$ and $\{\mathbf{z}_j|j\in\omega\}$ disjoint countable sets of degrees, there are $\mathbf{g}$ and $\mathbf{h}$ such that $\forall i,j\in\omega$ $((\mathbf{y}_i\oplus\mathbf{g})'\leq_T(\mathbf{y}_i\oplus\mathbf{h})'$ & $(\mathbf{z}_j\oplus\mathbf{g})'\nleq_T(\mathbf{z}_j\oplus\mathbf{h})')$.*

Corollary 1.8 *If $\mathbf{C}$ is any countable set of degrees then it is uniformly Δ_0-definable in parameters, i.e. there is a single formula which defines every such $\mathbf{C}$ as the parameters vary.*

Proof. Choose any strict upper bound $\mathbf{z}$ for $\mathbf{C}$. Let $\{\mathbf{y}_i\}=\{\mathbf{x}<\mathbf{z}:\mathbf{x}\in\mathbf{C}\}$ and $\{\mathbf{z}_j\}=\{\mathbf{x}<\mathbf{z}:\mathbf{x}\notin\mathbf{C}\}$. Apply the theorem to get the degrees $\mathbf{g}$ and $\mathbf{h}$. $\mathbf{C}$ is then $\{\mathbf{x}<\mathbf{z}:(\mathbf{x}\vee\mathbf{g})'\leq_T(\mathbf{x}\vee\mathbf{h})'\}$ and so our desired formula is $x<w\,\&\,(x\vee u)'\leq_T(x\vee v)'$. $\square$

We prove this theorem in §3. In the final section we summarize the state of affairs and point out some new problems suggested by our view of these matters.

2 Coding Register Machines

In this section we will explain the algebraic aspects of our codings and derive the main theorem, assuming these codings can be interpreted in $\mathcal{R}$. The next section will provide the recursion theoretic arguments to show that the structures described here can be realized in the r.e. degrees. For completeness, we reprise (with some small simplification allowed by our construction here) the presentation of coding register machines in Miller, Nies and Shore [2004] beginning with a standard description of the k-register machines of Shepherdson and Sturgis [1963] and Minsky [1961] and their representation in predicate logic as in Nerode and Shore [1997, III.8] or Börger, Grädel and Gurevich [1997, 2.1].

A *k-register machine* consists of k many storage locations called registers. Each register contains a natural number. There are only two types of operations that these machines can perform in implementing a program. First, they can increase the content of any register by one and then proceed to the next instruction. Second, they can check if any given register contains the number 0 or not. If so, they go on to the next instruction. If not, they decrease the given register by one and can be told to proceed to any instruction in the program. Formally, we define register machine programs and their execution as follows:

A *k-register machine program* I is a finite sequence $I_1, \dots, I_t, I_{t+1}$ of instructions operating on a sequence of numbers $x_1, \dots, x_k$, where each instruction I_m, for $m \leq t$, is of one of the following two forms:

(i) $x_i := x_i + 1$ (replace x_i by $x_i + 1$)

(ii) If $x_i \neq 0$, then $x_i := x_i - 1$ and go to j. (If $x_i \neq 0$, replace it by $x_i - 1$ and proceed to instruction I_j.)

It is assumed that after executing some instruction I_m, the execution proceeds to I_{m+1}, the next instruction on the list, unless I_m directs otherwise. The execution of such a program proceeds in the obvious way on any input of values for $x_1, \dots, x_k$ (the initial content of the registers) to change the values of the x_i and progress through the list of instructions. The final instruction, I_{t+1}, is always a halt instruction. Thus, if I_{t+1} is ever reached, the execution terminates with the current values of the x_i. In general, we denote the assertion that an execution of the program I is at instruction I_m with values $n_1, \dots, n_k$ of the variables by $I_m(n_1, \dots, n_k)$.

The standard translation of a register machine M describes the action of M by a system of universal axioms in the language of one unary function s thought of as the successor function on $\mathbb{N}$. For technical reasons peculiar to our later coding, we want to use distinct domains D_i with least elements 0_i. In our application here, these sets will be of the form $\{\mathbf{q}_i^{(n)} | n \in \omega\}$ for arithmetically independent degrees $\mathbf{q}_i$. The successor operators s_i will all be the jump operator. For now, we describe the axioms needed in predicate logic with additional k-ary relations P_m corresponding to the instructions I_m.

For each instruction I_m, $1 \leq m \leq t$, include an axiom of the appropriate form:

(i) $P_m(x_1, \dots, x_k) \to P_{m+1}(x_1, \dots, x_{i-1}, s_i(x_i), x_{i+1}, \dots, x_k)$.

(ii) $P_m(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_k) \to P_{m+1}(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_k)$
$\wedge\ P_m(x_1, \dots, x_{i-1}, s_i(y), x_{i+1}, \dots x_k) \to P_j(x_1, \dots, x_{i-1}, y, x_{i+1}, \dots, x_k)$.

(Note that being a successor is equivalent to being nonzero, i.e. not equal to $\mathbf{q}_i$ which is the 0_i of D_i.)

Let $P(I)$ be the finite set of universal axioms corresponding in this translation to register program I. It is easy to prove that, program I halts on input $(n_1, \dots, n_k)$ if and only if the sentence $F_k(n_1, \dots, n_k) \equiv P_1(s^{n_1}(0), \dots s^{n_k}(0)) \to \exists x_1, \dots, \exists x_k[P_{t+1}(x_1, \dots, x_k)]$ is a logical consequence of $P(I)$. As it is a classical fact (Shepherdson and Sturgis [1963]; Minsky [1961]) that the halting problem for 2-register machine programs is r.e. complete, it suffices to code all such models with binary predicates to get undecidability.

As usual for interpretations, we now want to provide formulas $\Delta_i(\vec{q}, x)$, $\Pi_m(\vec{q}, x, y)$ defining, for each choice of parameters $\vec{q}$, sets D_i ($i = 1, 2$) and binary relations P_m on $D_1 \times D_2$ ($1 \le m \le t+1$). (Remember s_i will be interpreted as the jump operator.) We take q_1 and q_2 to be the interpretations of 0 in D_1 and D_2 respectively. We now interpret our formulas $P(I) \to F(n_1, \dots, n_k)$ in the usual way. We relativize the quantifiers to the appropriate domain, i.e. $\exists x_i(\dots)$ becomes $\exists x_i(\Delta_i(\vec{q}, x_i) \wedge \dots)$ and $\forall x_i(\dots)$ becomes $\forall x_i(\Delta_i(\vec{q}, x_i) \to \dots)$. We then replace occurrences of $s_i(x_i)$ by x_i' and ones of $P_m(x_1, x_2)$ by $\Pi_m(\vec{q}, x_1, x_2)$. We indicate this translation by $*$. We also need a correctness condition Θ that says that $q_i \in D_i$ and the jump is a function on the D_i: $\Delta_1(\vec{q}, q_1) \wedge \Delta_2(\vec{q}, q_2) \wedge \forall x_1(\Delta_1(\vec{q}, x_1) \to \Delta_1(\vec{q}, x_1')\ \wedge\ \forall x_2(\Delta_2(\vec{q}, x_2) \to \Delta_2(\vec{q}, x_2')$. The class of sentences of $\mathcal{R}(\le, \vee, \wedge)$ that we want will then be those of the form $\forall \vec{q}[\Theta \to (P(I)^* \to F_2^*)]$ where I ranges over programs for 2-register machines.

It is clear that to get these sentences to be $\forall\exists$ ones it is sufficient to get quantifier free definitions (Δ_i and Π_m) of the domains and relations (and the worst that would work would be equivalent Σ_1 and Π_1 definitions). For the undecidability it suffices, of course, for the D_i to include ones isomorphic to $\{\mathbf{q}_i^{(n)}\}$ and the relations to include all binary relations on the $D_1 \times D_2$ as the parameters vary. If we take $\mathbf{q}_1$ and $\mathbf{q}_2$ to be arithmetically independent, i.e. $\mathbf{q}_i^{(m)} \le_T \mathbf{q}_1^{(n_1)} \vee \mathbf{q}_2^{(n_2)} \Leftrightarrow m \le n_i$ then we can code any relation $R_m \subseteq D_1 \times D_2$ in a Δ_0 way from the D_j and one additional countable set of degrees $T_m = \{\mathbf{q}_1^{(n_1)} \vee \mathbf{q}_2^{(n_2)} : \langle n_1, n_2 \rangle \in R_m\}$ by $R_m(x_1, x_2) \Leftrightarrow x_1 \in D_1 \,\&\, x_2 \in D_2 \,\&\, x_1^{(n_1)} \vee x_2^{(n_2)} \in T_m$.

So to prove our $\forall\exists$ undecidability result for $\mathcal{D}(\le,')$ it suffices to be able to define arbitrary countable subsets of $\mathcal{D}$ by a Δ_0 formula in parameters. As we pointed out at the end of the introduction, Theorem 1.7 provides this Δ_0 formula.

3 The Coding Theorem

We now prove our main technical result that says that by joining arbitrary given degrees with degrees $\mathbf{g}$ and $\mathbf{h}$ of our choosing we can make the jumps of the results be comparable or not as desired.

Theorem 1.7 *Given* $\{\mathbf{y}_i | i \in \omega\}$ *and* $\{\mathbf{z}_j | j \in \omega\}$ *disjoint countable sets of degrees, there are* $\mathbf{g}$ *and* $\mathbf{h}$ *such that* $\forall i, j \in \omega\ ((\mathbf{y}_i \oplus \mathbf{g})' \leq_T (\mathbf{y}_i \oplus \mathbf{h})'\ \&\ (\mathbf{z}_j \oplus \mathbf{g})' \not\leq_T (\mathbf{z}_j \oplus \mathbf{h})')$.

We fix representatives Y_i and Z_j $(i, j \in \omega)$ of the degrees $\mathbf{y}_i$ and $\mathbf{z}_j$, respectively and construct sets G and H such that for every $i, j \in \omega$ the following requirements are satisfied:

- $C_i : (Y_i \oplus G)' \leq_T (Y_i \oplus H)'$.
- $D_j : (Z_j \oplus G)' \not\leq_T (Z_j \oplus H)'$.

Our construction will be a forcing argument. Our coding procedure for the C_i requirements relies on the fact that, for any A and B, $A' \in \Pi_2^B \Rightarrow A' \leq_T B'$ (Theorem 4.3 of Soare [1987, IV] relativized to B). Thus our plan for satisfying C_i is to make sure that, for almost every n, if (we force) $n \in (Y_i \oplus G)'$ then we immediately force some canonically chosen Π_2^0 fact (the $h(n)^{th}$ one for some recursive h) about $Y_i \oplus H$ to be true. On the other hand, if (we force) $n \notin (Y_i \oplus G)$ we want to immediately force this fact to be false. Thus h provides a one-one reduction from A' to the complete Π_2^B set and so shows that $A' \in \Pi_2^B$. We make a list of the corresponding subrequirements $C_{i,n}$ where h is some specific recursive function that we will define later:

- $C_{i,n} : n \in (Y_i \oplus G)' \Leftrightarrow h(n) \notin (Y_i \oplus H)''$.

To satisfy C_i we must satisfy $C_{i,n}$ for almost every n.

Our diagonalization strategy to satisfy the requirements D_j is based on the fact that if $A' \leq_T B'$ then $\Sigma_1^A \in \Delta_2^B$ and so $\Pi_2^A \in \Pi_2^B$. Thus there is a Δ_0 formula θ such that if $(Z_j \oplus G)' \leq_T (Z_j \oplus H)'$ then there is a recursive function f_k such that $n \notin (Z_j \oplus G)'' \Leftrightarrow \forall u \exists v \theta(f_k(n), u, v, Z_j, H)$. (Here we have listed all the recursive functions as f_k, $k \in \omega$.) Our plan is then for each k (i.e. for each recursive function) to choose an n and, at some stage s of the construction, meet the following subrequirement:

- $D_{j,k} : \exists n \neg [n \notin (Z_j \oplus G)'' \Leftrightarrow \forall u \exists v \theta(f_k(n), u, v, Z_j, H)]$.

We will accomplish this by arranging that either we force $\exists u \forall v \neg \theta(f_k(n), u, v, Z_j, H)$ at stage s while immediately forcing the $\Pi_2^{Z_j \oplus G}$ fact that $n \notin (Z_j \oplus G)''$ to be true or, at stage s we force $\forall u \exists v \theta(f_k(n), u, v, Z_j, H)$ while immediately making the $\Sigma_2^{Z_j \oplus G}$ fact

$n \in (Z_j \oplus G)''$ true. In either case we violate the biconditional equivalent to $(Z_j \oplus G)' \leq_T (Z_j \oplus H)$ and meeting $D_{j,k}$ for all k guarantees that $(Z_j \oplus G)' \not\leq_T (Z_j \oplus H)$ as required to satisfy D_j.

To give us immediate control (both positively and negatively) over a canonical list of Π_2 facts, we employ a variation of Kumabe-Slaman forcing (See Shore and Slaman [1999]). That forcing builds a (not necessarily recursive) Turing functional Φ and allows one to immediately guarantee that, for any set X, $\Phi(X)$ is a finite function (a canonical $\Sigma_2^{X \oplus \Phi}$ fact). We wish to extend this to allow such control over a recursive set of such facts and to also allow "immediate forcing" of their negations. The first goal is achieved by considering the action of $\Phi(X)$ on individual columns (of inputs) rather than all possible inputs. The second is met by adding on an additional second set of restraints on extensions that prevent the Σ_2 fact of interest from ever being immediately forced (for any witness). General genericity arguments will then guarantee that once we have done this, the Π_2 fact will immediately be forced.

We now supply the formal definitions of our forcing relation.

Definition 3.1 *A* Turing functional Φ *is a set of sequences (x, y, σ) (the axioms) such that x is a natural number (the input), y is either* 0 *or* 1 *(the output), and σ is a finite binary sequence (the use). Furthermore, for all x, for all y_1 and y_2, and for all compatible σ_1 and σ_2, if $(x, y_1, \sigma_1) \in \Phi$ and $(x, y_2, \sigma_2) \in \Phi$, then $y_1 = y_2$ and $\sigma_1 = \sigma_2$. (This says that only one axiom in Φ applies to a particular oracle on any particular input. It is technically useful.)*

We write $\Phi(x, \sigma) = y$ to indicate that there is an initial segment τ of σ, possibly equal to σ, such that $(x, y, \tau) \in \Phi$. If $X \subseteq \omega$, we write $\Phi(x, X) = y$ to indicate that there is an ℓ such that $\Phi(x, X \upharpoonright \ell) = y$, and write $\Phi(X)$ for the function evaluated in this way. (Note that the set of axioms need not be recursively enumerable so the functions computed are partial recursive in the functional Φ plus the input set X.)

Definition 3.2 *Conditions p in our forcing notion $\mathcal{P}$ are triples $\langle \Phi_p, \mathbf{X}_p, \mathbf{W}_p \rangle$ where Φ_p is a finite Turing functional and both $\mathbf{X}_p$ and $\mathbf{W}_p$ are finite collections of pairs $\langle X, n \rangle$ (or $\langle W, n \rangle$) consisting of one set and one natural number such that $\mathbf{X}_p \cap \mathbf{W}_p = \emptyset$. We say that q extends p, $q \leq p$, if*

1. $\Phi_p \subseteq \Phi_q \,\&([(x, y, \sigma) \in \Phi_p \,\&\, (x', y', \sigma') \in \Phi_q - \Phi_p] \rightarrow |\sigma| < |\sigma'|)$.
2. $\mathbf{X}_p \subseteq \mathbf{X}_q \,\&\, \mathbf{W}_p \subseteq \mathbf{W}_q$.
3. $(\forall x, y, \langle X, n \rangle \in \mathbf{X}_p)(\Phi_q(\langle n, x \rangle, X) = y \rightarrow \Phi_p(\langle n, x \rangle, X) = y)$.

If K is a filter on $\mathcal{P}$ then $\Phi_K = \cup\{\Phi_p | p \in K\}$ is a Turing functional. It is this functional that will be our generic object Φ and about which we will be able to speak in our forcing language. The second clause of (1) says that only longer axioms than the ones we already have can be added on by an extension. Clause (3) says that no new axioms can be added which apply to the oracle X on any input in column n.

One can now give a common definition (in the setting of forcing in arithmetic) of $p \Vdash \psi$ for $p \in \mathcal{P}$ and ψ a sentence of arithmetic with bounded as well as unbounded quantifiers and an added unary relation symbol Φ (for the generic Turing functional) as well as ones Y_i and Z_j for our fixed choice of representatives of the degrees mentioned in Theorem 1.7.

Definition 3.3 *We define $p \Vdash \psi$ by induction on formulas where we view $\forall x$ as an abbreviation for $\neg\exists x\neg$ and consider sentences with leading (unbounded) quantifiers ($\exists$ and $\forall$) and negation symbols followed by a bounded formula.*

1. If ψ is Δ_0 (i.e. has only bounded quantifiers) then $p \Vdash \psi$ if and only if $\psi(\Phi_K)$ is true for every filter K on $\mathcal{P}$ containing p. (Thus if ψ and ψ' are logically equivalent Δ_0 formulas then $p \Vdash \psi \Leftrightarrow p \Vdash \psi'$.)
2. $p \Vdash \exists n\psi(n)$ if and only if there is an $n \in \omega$ such that $p \Vdash \psi(n)$.
3. $p \Vdash \neg\psi$ if and only if no $q \le p$ forces ψ.

We say that a filter K on $\mathcal{P}$ is *1-generic* if for every Δ_0 formula $\psi(x)$ there is a $p \in K$ such that either $p \Vdash \exists x\psi(x)$ or $p \Vdash \neg\exists x\psi(x)$ (or equivalently, $p \Vdash \forall x\neg\psi(x)$).

Clause (3) of Definition 3.2 is the key to making our canonical Π_2 facts false as putting $\langle X, n\rangle$ into $\mathbf{X}_p$ immediately forces $\Phi(X) \upharpoonright \omega^{[n]}$ to be finite. The restriction that $\mathbf{X}_p \cap \mathbf{W}_p = \emptyset$ means that once a pair $\langle W, n\rangle$ has gone into $\mathbf{W}_p$ then it can never go into $\mathbf{X}_p$ and so, if Φ is sufficiently generic (even 1-generic) $\Phi(W)$ will be total on $\omega^{[n]}$. Thus we see that our canonical $\Pi_2^{X\oplus\Phi}$ facts can be taken to be that $\Phi(X)$ is total on $\omega^{[n]}$. Putting $\langle X, n\rangle$ into $\mathbf{X}_p$ will immediately force this sentence to be false while putting it into $\mathbf{W}_p$ will make it true as long as K is at least 1-generic. In the terminology of the subrequirements $C_{i,n}$ this corresponds to choosing a recursive h such that $h(n) \notin (X \oplus \Phi)'' \Leftrightarrow \forall x\exists y(\Phi(\langle n, x\rangle, X) = y)$ for every Φ and X.

We now list some of the basic facts about this forcing relation. Key among them are that forcing for bounded sentences is recursive (in the set parameters) and that, if p belongs to a 1-generic filter K, $p \Vdash \Theta(\Phi)$ where Θ is Σ_2 or Π_2 then $\Theta(\Phi_K)$ holds. These facts allow us to control the forcing relation when needed and guarantee that forcing the sentences relevant to our requirements makes them true of the functionals associated with the 1-genreic filters that we construct.

Remark 3.4 *If ψ is Δ_0 and K is a filter on $\mathcal{P}$ then we have the following facts:*

1. *$p \Vdash \exists u\psi(u, \Phi) \Rightarrow \exists q \le p\exists u[(\Phi_q, \emptyset, \emptyset) \Vdash \psi(u, \Phi)]$ since some initial segment Φ_q of any Φ_K with $p \in K$ suffices to guarantee the truth of the Δ_0 sentence. Of*

course, if $p \in K$ then $\exists x\psi(x, \Phi_K)$ is true. Conversely, if $\exists x\psi(x, \Phi_K)$ is true then $\exists p \in \mathcal{P}(p \Vdash \exists x\psi(x, \Phi)$ for the same reason and the fact that the definition of the forcing order guarantees that once a use σ is included in an axiom of Φ then no axioms (with codes) smaller than σ can ever be inserted.

2. *$(\Phi_q, \emptyset, \emptyset) \Vdash \psi(u, \Phi)$ is a uniformly recursive relation in q, u, ψ, Φ and the Y_i and Z_j appearing in ψ as the truth of $\psi(u, \Phi_K)$ depends only on initial segments of Φ_K (and the Y_i and Z_j). We abbreviate this relation as $\Phi_q \Vdash \psi(u, \Phi)$.*

3. *If $p \Vdash \forall u\psi(u, \Phi)$ then $[(\Phi_p, \mathbf{X}_p, \emptyset) \Vdash \forall u\psi(u, \Phi)]$ as otherwise there would be a $u \in \omega$ and $q \leq (\Phi_p, \mathbf{X}_p, \emptyset)$ such that $q \Vdash \neg\psi(u, \Phi)$ and so a $q' \leq q$ such that $(\Phi_{q'}, \emptyset, \emptyset) \Vdash \neg\psi(u, \Phi)$ but then $(\Phi_{q'}, \mathbf{X}_p, \mathbf{W}_p)$ extends p but also forces $\neg\psi(u, \Phi)$ for a contradiction.*

4. *If $p \Vdash \forall u\psi(u, \Phi)$ and $p \in K$ then $\forall u\psi(u, \Phi_K)$ as otherwise there would be a u such that $\neg\psi(u, \Phi_K)$ and so a $q \in K$ such that $q \Vdash \neg\psi(u, \Phi)$ contradicting the compatibility required by $p, q \in K$ and the definition of forcing. Conversely, if K is 1-generic and $\forall u\psi(u, \Phi_K)$ then $\exists p \in K(p \Vdash \forall u\psi(u, \Phi)$ as otherwise there would be a $q \in K$ such that $q \Vdash \exists u\neg\psi(u, \Phi)$ and so $\exists u\neg\psi(u, \Phi_K)$ would hold for a contradiction.*

5. *If $p \Vdash \exists x\forall y\psi(x, y, \Phi)$ and $p \in K$ then $\exists x\forall y\psi(x, y, \Phi_K)$ by the definition of forcing and (4). Similarly, if K is 1-generic, $p \in K$ and $p \Vdash \forall x\exists y\psi(x, y, \Phi)$ then $\forall x\exists y\psi(x, y, \Phi_K)$ as otherwise there would be an x such that $\neg\exists y\psi(x, y, \Phi_K)$, and so (by 4) a $q \in K$ such that $q \Vdash \neg\exists y\psi(x, y, \Phi)$. Again the compatibility of p and q gives us an r extending both for a contradiction (as $r \leq q$, $r \Vdash \exists x\neg\exists y\psi(x, y, \Phi)$ but as $r \leq p$, this contradicts the assumption that $p \Vdash \forall x\exists y\psi(x, y, \Phi)$.*

We will build G and H to be the Turing functionals corresponding to two 1-generic filters K and L for this forcing and to satisfy a specific list of other conditions by building two sequences p_s, q_s such that $\cup\Phi_{p_s} = \Phi_K = G$ and $\cup\Phi_{q_s} = \Phi_L = H$. The requirements that we have to satisfy in addition to those for 1-genericity are ones to guarantee that C_i and D_j are true of G and H.

As described above, our plans with our now known list of canonical Π_2 facts are as follows. To satisfy C_i we make sure that, for almost every n, if $n \in (Y_i \oplus G)'$ then we put $\langle Y_i, n\rangle$ into $\mathbf{W}_{q_m}$ for some m while if $n \notin (Y_i \oplus G)'$ we put it into $\mathbf{X}_{q_m}$ for some m. In the first case, this makes $h(n) \notin (Y_i \oplus H)''$, indeed $\operatorname{dom}\Phi_L(Y_i) \upharpoonright \omega^{[n]} = \omega^{[n]}$ for any 1-generic L containing q. In the second case, this makes $\operatorname{dom}\Phi_L(Y_i) \upharpoonright \omega^{[n]}$ finite and so $h(n) \in (Y_i \oplus H)''$. Thus, if for almost all n we meet the requirements $C_{i,n}$, then $(Y_i \oplus G)' \in \Pi_2^{Y_i \oplus H}$ and $(Y_i \oplus G)' \leq_T (Y_i \oplus H)'$ as required.

To satisfy D_j, our plan is, for each k, to choose an n such that $n \notin (Z_j \oplus G)'' \Leftrightarrow (G(Z_j) \upharpoonright \omega^{[n]}$ is total) and, at some stage s of the construction, meet the subrequirement $D_{j,k}$ that $\neg[n \notin (Z_j \oplus G)'' \Leftrightarrow \forall u\exists v\theta(f_k(n), u, v, Z_j, H)]$, i.e. $\neg[G(Z_j) \upharpoonright \omega^{[n]}$ is total $\Leftrightarrow$

$\forall u \exists v \theta(f_k(n), u, v, Z_j, H)]$ by arranging that either $q_s \Vdash \exists u \forall v \neg\theta(f_k(n), u, v, Z_j, \Phi)$ while $\langle Z_j, n\rangle \in \mathbf{W}_{p_s}$ which guarantees that $G(Z_j) \restriction \omega^{[n]}$ is total; or $q_s \Vdash \forall u \exists v \theta(f_k(n), u, v, Z_j, \Phi)$ while $\langle Z_j, n\rangle \in \mathbf{X}_{p_s}$ which guarantees that $G(Z_j) \restriction \omega^{[n]}$ is finite. In either case we violate the biconditional equivalent to $(Z_j \oplus G)' \leq_T (Z_j \oplus H)'$. Meeting $D_{j,k}$ for all k guarantees that $(Z_j \oplus G)' \not\leq_T (Z_j \oplus H)$ as required. We now give the details of the construction.

Construction: We make an ω-list of the subrequirements $C_{i,n}, D_{j,k}$ described above as well as others P_m to decide the one quantifier sentences $\exists w \psi_m$ about G and H. We begin at stage 0 with the empty conditions $p_0 = q_0 = \langle \emptyset, \emptyset, \emptyset\rangle$. Suppose we are at stage $s+1$ with p_s, q_s defined. We let t_i be the least s at which Y_i is mentioned in one of the conditions, i.e. $\langle Y_i, m\rangle \in \mathbf{X}_{p_s} \cup \mathbf{W}_{p_s} \cup \mathbf{X}_{q_s} \cup \mathbf{W}_{q_s}$ for some m. Our action depends on the requirement assigned to s, i.e. the s^{th} one on our list.

P_m : If there is a finite $\Phi' \supseteq \Phi_{p_s}$ (consistent with p_s, i.e. $(\Phi', \mathbf{X}_{p_s}, \mathbf{W}_{p_s}) \leq p_s$) and a w such that $\langle \Phi', \mathbf{X}_{p_s}, \mathbf{W}_{p_s}\rangle \Vdash \psi_m(w, \Phi)$ then we choose one and set $p_{s+1} = \langle \Phi', \mathbf{X}_{p_s}, \mathbf{W}_{p_s}\rangle$. If not we let $p_{s+1} = p_s$. We then do the same for H to define q_{s+1} from q_s.

$C_{i,n}$: If $\langle Y_i, n\rangle$ is already on either list in q_s let $p_{s+1} = p_s$ and $q_{s+1} = q_s$. Otherwise, ask if there is a finite $\Phi' \supseteq \Phi_{p_s}$ (consistent with p_s) such that $\langle \Phi', \mathbf{X}_{p_s}, \mathbf{W}_{p_s}\rangle \Vdash n \in (Y_i \oplus \Phi)'$. If so, choose one and let $p_{s+1} = \langle \Phi', \mathbf{X}_{p_s}, \mathbf{W}_{p_s}\rangle$ and $q_{s+1} = \langle q_s, \mathbf{X}_{q_s}, \mathbf{W}_{q_s} \cup \{\langle Y_i, n\rangle\}\rangle$. If not, let $p_{s+1} = p_s$ and $q_{s+1} = \langle \Phi_{q_s}, \mathbf{X}_{q_s} \cup \{\langle Y_i, n\rangle\}, \mathbf{W}_{q_s}\rangle$.

$D_{j,k}$: Choose an n such that $\langle Z_j, n\rangle$ does not appear on either list in p_s. Ask if there is a u and $q' \leq q_s$ (which by Remark 3.4 can be to assumed to have $\mathbf{W}_{q'} = \mathbf{W}_{q_s}$) such that $q' \Vdash \forall v \neg\theta(f_k(n), u, v, Z_j, \Phi)$ and a $p' \leq p_s$ (which again by Remark 3.4 can be assumed to have $\mathbf{W}_{p'} = \mathbf{W}_{p_s}$) so that for any m and any $t_i \leq s$ such that $Y_i \leq_T Z_j$ and $\langle Y_i, m\rangle \in \mathbf{X}_{q'} - \mathbf{X}_{q_{t_i}}$ we have that $p' \Vdash m \notin (Y_i \oplus \Phi)'$.

If not (Π_2 outcome), then we let $p_{s+1} = \langle \Phi_{p_s}, \mathbf{X}_{p_s} \cup \{\langle Z_j, n\rangle\}, \mathbf{W}_{p_s}\rangle$ and $q_{s+1} = q_s$.

If there are such u, q' and p' (Σ_2 outcome) then we claim (and will verify in Lemma 3.8 below) that we can choose such a q' with no $\langle Y_i, l\rangle \in \mathbf{X}_{q'} - \mathbf{X}_{q_s}$ with $Y_i \not\leq_T Z_j$ and a p' for which $\langle Z_j, n\rangle \notin \mathbf{X}_{p'}$. We choose such and set $p_{s+1} = \langle \Phi_{p'}, \mathbf{X}_{p'}, \mathbf{W}_{p'} \cup \{\langle Z_j, n\rangle\}\rangle$ and $q_{s+1} = q'$.

Verifications: It is clear that there are no difficulties carrying out these instructions when the requirement being considered is P_m or $C_{i,n}$. Moreover, it is also clear that, in the case of P_m, we have guaranteed that the Σ_1 formula is forced at $s+1$ if possible and otherwise no extension forces it. Thus K and L are 1-generic. Similarly, in the case of $C_{i,n}$, as long as from some stage t onward no actions other than for $C_{i,m}$ put any $\langle Y_i, m\rangle$ on any list in q_s without our forcing the corresponding outcome for $m \in (Y_i \oplus G)'$, then we satisfy C_i. In fact, our actions guarantee a bit more. If $\langle Y_i, m\rangle$ is put into $\mathbf{X}_{q_s}$ at $s \geq t_i$ (by $C_{i,m}$ or some $D_{j,k}$ acting for its Σ_2 outcome as described above) then $p_{s+1} \Vdash m \notin (Y_i \oplus \Phi)'$ as required; and no action other than for $C_{i,m}$ can put $\langle Y_i, m\rangle$ into $\mathbf{W}_{q_s}$ for $s \geq t_i$. Thus we are left with analyzing the way we decide the Π_2 questions for $D_{j,k}$ at $s_{j,k}$.

The simpler case is the Π_2 outcome. Here we only have to show that, for each u, when we consider the requirement P_m for the formula $\exists v\theta(f_k(n),u,v,Z_j,\Phi)$ for H at $s > s_{j,k}$ that we force the Σ_1 outcome. If not, then by our construction, $q_s \Vdash \forall v\neg\theta(f_k(n),u,v,Z_j,\Phi)$ and $q_s \leq q_{s_{j,k}}$. We now claim that if $\langle Y_i,m\rangle \in \mathbf{X}_{q_s} - \mathbf{X}_{q_{t_i}}$ then $p_s \Vdash m \notin (Y_i \oplus \Phi)'$ which would put us in the Σ_2 outcome for a contradiction. To verify the claim consider how $\langle Y_i,m\rangle$ could have entered $\mathbf{X}_{q_s}$. No action for P_m adds anything to either list. Action for $C_{h,l}$ adds at most $\langle Y_h,l\rangle$ to the $\mathbf{X}$ list but only when it already forces $l \notin (Y_h \oplus \Phi)'$ (for G) as required. Finally, by the claim in the Σ_2 case of the construction (Lemma 3.8 below) action for some $D_{h,l}$ at a stage t after t_i (and before s) puts $\langle Y_i,m\rangle$ into $\mathbf{X}_{q_s}$ only when it also makes $p_{t+1} \Vdash m \notin (Y_i \oplus \Phi)'$ as required.

The Σ_2 case requires a deeper analysis of the forcing relation along the lines of Shore and Slaman [1999] albeit somewhat more elaborate combinatorially. The idea here (as in Shore and Slaman [1999]) is that if there is a $q' \leq q$ such that $q' \Vdash \forall v\neg\theta(f_k(n),u,v,Z_j,\Phi)$ then, first, it can be taken to be of the form $(\Phi',\mathbf{X}',\mathbf{W}_q)$. Next, by the basic properties of forcing, no extension $q'' = (\Phi'',\mathbf{X}'',\mathbf{W}_q)$ of q' forces $\exists v\theta(f_k(n),u,v,Z_j,\Phi)$ so, if for some m, some $\Phi'' \supseteq \Phi'$ makes $\theta(f_k(n),u,m,Z_j,H)$ true then it must be incompatible with the conditions imposed by $\mathbf{X}'$, i.e. some axiom in $\Phi'' - \Phi'$ with input in column e applies to an X_i with $\langle X_i,e\rangle \in \mathbf{X}'$. We can form a tree of approximations to such an $\mathbf{X}'$ by considering those sequences $\langle\sigma_l,s_l\rangle$ such that any such Φ'' has an axiom compatible with one of the $\langle\sigma_l,s_l\rangle$. Now we must also associate with any such q' a condition $p' \leq p$ that forces $m \notin (Y_i \oplus \Phi)'$ for the i and m that we worry about in the definition of the Σ_2 outcome. This adds one more layer of approximations to produce p' from a path in the appropriate tree along with q'. We are thus led to the following definitions.

Definition 3.5 *Suppose we are given conditions p and q and an instance of one of our formulas $\forall v\neg\theta(a,u,v,Z,\Phi)$ where we are thinking about H as the interpretation of Φ and have written a for $f_k(n)$ and Z for Z_j. We say that i is* crucial (for p and q) *if $Y_i \leq_T Z$ and Y_i is on some list in p or q. Let t be larger than all numbers mentioned in p or q (either in one of the lists or one of the Turing functionals) and such that $U \restriction t \neq V \restriction t$ for any sets $U \neq V$ appearing on any list in p or q. For each $m \geq t$ and finite Turing functionals Γ and Θ such that $\langle\Gamma,\mathbf{X}_p,\mathbf{W}_p\rangle \leq p$ and $\langle\Theta,\mathbf{X}_q,\mathbf{W}_q\rangle \leq q$ we define the tree $T_m(\Gamma,\Theta,p,q) = T$ whose nodes are sequences $\langle\langle\sigma_l,s_l\rangle,\vec{\tau}_l\rangle_{l\leq m}$ (coded as numbers in some fixed way as are all finite sets and sequences) where each $\vec{\tau}_l$ is itself a sequence $\langle\tau_{l,e},t_{l,e}\rangle_{e\leq m}$ with the following properties:*

1. *$|\sigma_l|, |\tau_{l,e}| \geq m$ and all of these lengths are the same; $\langle W,s_l\rangle \in \mathbf{W}_q \Rightarrow \sigma_l \restriction m \nsubseteq W \restriction m$; $\langle W,t_{l,e}\rangle \in \mathbf{W}_p \Rightarrow \tau_{l,e} \restriction m \nsubseteq W \restriction m$.*

2. *If $\Theta' \supseteq \Theta$ with the code for Θ' less than the one for the node and $\Theta' \Vdash \theta(a,u,v,Z,\Phi)$ for some v also less than the code for the node, then $\Theta' - \Theta$ contains an axiom $(\langle s,z\rangle,y,\sigma)$ such that, for some $l \leq m$, $s = s_l$ and σ is compatible with σ_l.*

3. *If $\sigma_l \subseteq Y_i$ for some crucial i, $\Gamma' \supseteq \Gamma$ with the code for Γ' less than the one for the node and $\Gamma' \Vdash s_l \in (Y_i \oplus \Phi)'$ with a witness for convergence also less than the code*

for the node then $\Gamma' - \Gamma$ contains an axiom $(\langle t, z\rangle, y, \tau)$ such that, for some $e \leq m$, $t = t_{l,e}$ and τ is compatible with $\tau_{l,e}$.

We order the nodes of T in the expected way: $\langle\langle\sigma_l', s_l'\rangle, \vec{\tau}_l'\rangle \preceq_T \langle\langle\sigma_l, s_l\rangle, \vec{\tau}_l\rangle \Leftrightarrow$ $(\forall l \leq m)(\sigma_l' \subseteq \sigma_l \,\&\, s_l = s_l' \,\&\, (\forall e \leq m)(\tau_{l,e}' \subseteq \tau_{l,e} \,\&\, t_{l,e} = t_{l,e}'))$.

Lemma 3.6 *T is a finitely branching tree recursive in Z.*

Proof. T is obviously finitely branching since the σ_l and $\tau_{l,e}$ are binary sequences. As m is fixed the information needed in (1) is finite so it is a recursive condition. Since forcing a Δ_0 formula is recursive (in the set parameters), (2) is recursive in Z. The list of Y_i mentioned in p or q and recursive in Z is finite and Z can tell if σ_l is an initial segment of one of these Y_i while the rest of the condition is recursive given our bound on the witness for convergence in $s_l \in (Y_i \oplus \Phi)'$. $\Box$

Lemma 3.7 *If there are $q' = (\Theta, \mathbf{X}_{q'}, \mathbf{W}_q)$ and $p' = (\Gamma, \mathbf{X}_{p'}, \mathbf{W}_p)$ as required in the definition of the Σ_2 outcome for $D_{j,k}$ then for for some (large enough) m, $T_m(\Gamma, \Theta, p, q) = T$ has an infinite path.*

Proof. We claim that if $\mathbf{X}_{q'} = \{\langle S_l, s_l\rangle | l \leq m\}$ and $\mathbf{X}_{p'} = \{\langle T_e, t_e\rangle | e \leq m\}$ (we allow duplications to keep the indexing the same and $m \geq t$) then $\{\langle\langle S_l \restriction n, s_l\rangle_{l \leq m}, \vec{\tau}_{l,n}\rangle | n > m\}$ is a path in T where $\vec{\tau}_{l,n} = \langle T_e \restriction n, t_e\rangle_{e \leq m}$. Here we have chosen m larger than t and the cardinalities of $\mathbf{X}_{q'}$ and $\mathbf{X}_{p'}$ as well as to insure that if $U \neq V$ are mentioned in p' or q' then $U \restriction m \neq V \restriction m$. Note that if $\langle X, n\rangle \in \mathbf{X}_{q'} - \mathbf{X}_q$ and $\langle W, n\rangle \in \mathbf{W}_q$ then $X \neq W$ by the definition of the forcing ordering and so $X \restriction m \neq W \restriction m$ and similarly for p. To see that these nodes are on T check that each condition is satisfied for $\langle\langle S_l \restriction n, s_l\rangle_{l \leq m}, \vec{\tau}_{l,n}\rangle$:

1. The lengths are all larger than m by definition. The restrictions associated with $\mathbf{W}_q$ and $\mathbf{W}_p$ are satisfied by our choice of m and the note above.

2. If $\Theta' \supseteq \Theta$ and $\Theta' \Vdash \theta(a, u, v, Z, \Phi)$ for some v then Θ' is not consistent with q' and so $\Theta' - \Theta$ must contain an axiom $(\langle s, z\rangle, y, \sigma)$ such that there is an $l \leq m$ such that $s_l = s$ and $\sigma \subseteq S_l$. Clearly this σ is compatible with $S_l \restriction n$.

3. If $S_l \restriction n \subseteq Y_i$ and i is crucial (and so, in particular, $S_l = Y_i$ by our choice of m), $Y_i \leq_T Z$, $\Gamma' \supseteq \Gamma$ and $\Gamma' \Vdash s_l \in (Y_i \oplus \Phi)'$ then Γ' is not consistent with p' since by our requirements when we acted for $D_{j,k}$, $p' \Vdash s_l \notin (S_l \oplus \Phi)'$ and so $\Gamma' - \Gamma$ must contain an axiom $(\langle t, z\rangle, y, \tau)$ such that there is an $e \leq m$ such that $t = t_e$ and $\tau \subseteq T_e$. Clearly this τ is compatible with $T_e \restriction n$. $\Box$

Lemma 3.8 *If there is an infinite path $P = \{\langle\langle\sigma_{n,l}, s_l\rangle, \vec{\tau}_{n,l}\rangle\rangle_{l \leq m} | n \in \omega\}$ in some $T_m(\Gamma, \Theta, p, q)$ $= T$ then there are q' and p' as required by the claim in the implementation of the Σ_2 outcome for $D_{j,k}$.*

Proof. Any infinite path clearly produces sets $S_l = \cup\{\sigma_{n,l}|n \in \omega\}$ for $l \leq m$ and $R_{l,e} = \cup\{\tau_{n,l,e}|n \in \omega\}$ for $e \leq m$ if $S_l = Y_i$ some crucial i. As T is recursive in Z and has an infinite path we can choose one that does not compute any $Y_i \not\leq_T Z$ by Jockusch and Soare [1972] and so we may assume that no $S_l = Y_i$ for any crucial i. We let $q' = (\Theta, \mathbf{X}_q \cup \{\langle S_l, s_l\rangle | l \leq m\}, \mathbf{W}_q)$ which is clearly a condition since no $\langle S_l, s_l\rangle \in \mathbf{W}_q$ by clause (1) of the definition of T. We first argue that $q' \Vdash \forall v \neg\theta(a, u, v, Z, H)$. If not, then by Remark 3.4, there would be a $(\Theta', \mathbf{X}_{q'}, \mathbf{W}_{q'}) \leq q'$ and a v such that $\Theta' \Vdash \theta(a, u, v, Z, \Phi)$. By clause (2) of the definition of T there would then be an axiom $(\langle s, z\rangle, y, \sigma) \in \Theta' - \Theta$ and an $l \leq m$ such that $s = s_l$ and σ is compatible with $\sigma_{n,l} \subseteq S_l$ for infinitely many n and so $\sigma \subseteq S_l$. As this contradicts the definition of $(\Theta', \mathbf{X}_{q'}, \mathbf{W}_{q'}) \leq q'$, we have that $q' \Vdash \forall v \neg\theta(a, u, v, Z, \Phi)$ and there is no $\langle Y_i, h\rangle \in \mathbf{X}_{q'} - \mathbf{X}_q$ with $Y_i \leq_T Z$ as required.

We next argue that $\bar{p} = (\Gamma, \mathbf{X}_p \cup \{R_{l,e} | S_l = Y_i$ some crucial i and $e \leq m\}, \mathbf{W}_p) \Vdash s_l \notin (S_l \oplus \Phi)'$ for every $\langle S_l, s_l\rangle \in \mathbf{X}_{q'} - \mathbf{X}_{q_{t_i}}$ with $S_l = Y_i$ for some crucial i. Suppose $S_l = Y_i$ for some crucial i but $\bar{p} \nVdash s_l \notin (S_l \oplus \Phi)'$. In this case, there is a $\Gamma' \supseteq \Gamma$ consistent with $\bar{p}$ such that $\Gamma' \Vdash s_l \in (S_l \oplus \Phi)'$. By clause (3) of the definition of T, there must be a $(\langle t, z\rangle, y, \tau) \in \Gamma' - \Gamma$ such that, for some $e \leq m$, $t = t_{l,e}$ and τ is compatible with $\tau_{n,l,e}$ for infinitely many n. Thus $\tau \subseteq R_{l,e} \in \mathbf{X}_{\bar{p}}$ and so Γ' is not consistent with $\bar{p}$ for the desired contradiction.

Finally, we argue that we can find a $p' \leq p$ such that $\langle Z, n\rangle \notin \mathbf{X}_{p'}$ and $p' \Vdash s_l \notin (S_l \oplus \Phi)'$ for all the l's for which $S_l = Y_i$ for a crucial i and $\langle S_l, s_l\rangle \notin \mathbf{X}_{q_{t_i}}$ which will complete the proof. For each such l we consider the tree $T_{m,l}$ whose nodes are sequences $\langle \tau_e, t_e\rangle_{e \leq m}$ such that $(\forall e, e' \leq m)(|\tau_e| = |\tau_{e'}| > m \,\&\, (\langle W, t_e\rangle \in \mathbf{W}_p \rightarrow \tau_e \upharpoonright m \neq W \upharpoonright m))$ and $(\forall \Gamma' \supseteq \Gamma)$ (if the code for Γ' is less than the one for $\vec{\tau}$ and $\Gamma' \Vdash s_l \in (Y_i \oplus \Phi)'$ with a witness for convergence also less than the code for $\vec{\tau}$ then $\Gamma' - \Gamma$ contains an axiom $(\langle t, z\rangle, y, \tau)$ such that, for some $e \leq m$, $t = t_e$ and τ is compatible with τ_e). We order $T_{m.l}$ by $\langle \tau'_e, t'_e\rangle_{e \leq m} \preceq_{T_{m,l}} \langle \tau_e, t_e\rangle_{e \leq m} \Leftrightarrow (\forall e \leq m)(t'_e = t_e \,\&\, \tau'_e \subseteq \tau_e)$.

This tree is finitely branching and recursive in Y_i. Our original sequence $\langle \vec{\tau}_{n,l} | n \in \omega\rangle$ is a path in $T_{m,l}$ and so there is one that does not compute $Z \not\leq_T Y_i$. (Note here that as i is crucial $Y_i \leq_T Z$ but by the assumption of the Theorem they have different degree.) Let $\langle \vec{\tau}''_{n,l} | n \in \omega\rangle$ be such a path for each l as required and let $R'_{l,e} = \cup\{\tau'_{n,l,e} | n \in \omega\}$. As before we know that $p_l = (\Gamma, \mathbf{X}_p \cup \{\langle R_{l,e}, t'_{l,e}\rangle | e \leq m\}, \mathbf{W}_p) \Vdash s_l \notin (S_l \oplus \Phi)'$ while no $R'_{l,e}$ is Z. We can now get the required p' as $(\Gamma, \mathbf{X}_p \cup \{\langle R'_{l,e}, t'_{l,e}\rangle | S_l = Y_i$ for some crucial i and $\langle S_l, s_l\rangle \notin \mathbf{X}_{q_{t_i}} \,\&\, e \leq m\}, \mathbf{W}_p)$. As p' extends every p_l for l on our list, $p' \Vdash s_l \notin (S_l \oplus \Phi)'$ for all these l's while $\langle Z, n\rangle \notin \mathbf{X}_p$ by our choice of n and no $R_{l,e} = Z$, p' is as required to implement the Σ_2 outcome for $D_{j,k}$. $\square$

4 Conclusions and Questions

We summarize much of the current state of affairs about the dividing line between decidability and undecidability in the structures $\mathcal{R}$, $\mathcal{D}$ and $\mathcal{D}(\leq 0')$ in various languages in the following tables.

	$\mathcal{R}$	$\mathcal{D}$	$\mathcal{D}(\leq \mathbf{0}')$
$\exists(\leq, \vee)$	Dec	Dec	Dec
$\forall\exists(\leq, \vee)$	?	Dec	?
$\forall\exists\forall(\leq, \vee)$	Undec	Undec	Undec

	$\mathcal{R}$	$\mathcal{D}$	$\mathcal{D}(\leq \mathbf{0}')$
$\exists(\leq, \vee, \wedge)$	?	Dec	Dec
$\forall\exists(\leq, \vee, \wedge)$	Undec	Undec	Undec

	$\mathcal{D}$
$\exists(\leq, \vee, ')$	Dec
$\forall\exists(\leq, \vee, ')$	Undec

The first and third tables are straightforward. The second requires some explanation of the use of $\wedge$. We understand decidability in the $\exists$ line to mean that every finite lattice is embeddable, indeed this is true even if we preserve 0 and 1 (if it exists). (For $\mathcal{R}$ the question of which lattices are embeddable even without preserving 0 or 1 is, of course, still open.) Undecidability in the $\forall\exists$ line means that the theory is undecidable (via the same set of sentences) for every total extension of the partial infimum relation.

In addition to the long standing and well known open questions about $\mathcal{R}$, these tables suggest two other natural areas for investigation on the boundary between decidability and undecidability. The first is the $\forall\exists$ theory of $\mathcal{D}(\leq, ')$. This is a very rich theory that includes many interesting and difficult subproblems and a number of partial results. For example, it includes the $\exists$ theory of $\mathcal{D}(\leq, \vee, ')$ shown decidable by Montalban [2003]. It is also possible to define 0 at this level without any added complexity and so it is equivalent to the $\forall\exists$ theory of $\mathcal{D}(0, \leq, ')$ which includes $\exists$ theory of $\mathcal{D}(0, \leq, \vee, ')$. Now, for all the other decidability results in these tables which do not involve the jump operator, adding 0 to the language presents no serious extra difficulties. This is far from true once we allow the jump. Indeed even the $\exists$ theory of $\mathcal{D}(0, \leq, ')$ is a complex problem about which there are many difficult and interesting partial results. The crucial point is that the addition of $\mathbf{0}$ requires very fine control over the complexity of the degrees realizing some given $\exists$ statement in the language with just $\leq$ and $'$. The formula can now specify that witnesses lie in precise intervals with endpoints of the form $\mathbf{0}^{(n)}$. In contrast, the realizations in the decidability results for the $\exists$ theory of $\mathcal{D}(\leq, ')$ and $\mathcal{D}(\leq, \vee, ')$ (Hinman and Slaman [1991]; Montalban [2003]) produce witnesses which are not even hyperarithmetic.

Among the partial results on the $\exists$ theory with 0 we mention Lerman [1985] which proves the decidability of the $\exists$ theory of $\mathcal{D}$ with $\leq$ and additional predicates for the classes in the High/Low hierarchy. Montalban [2004] (answering a question of Lerman [1985]) proves the decidability of $\exists$ theory of $\mathcal{D}$ with $\leq, 0$ and additional predicates for the classes GL_n, GH_n and GI (for $n \geq 0$) in the generalized high/low hierarchies (which also falls within the $\exists$ theory of $\mathcal{D}(0, \leq, \vee, ')$). Lempp and Lerman [1996] prove the $\exists$

theory of $\mathcal{R}$ with $0, 1, \leq$ and predicates for $\mathbf{x}^{(n)} \leq_T \mathbf{y}^{(n)}$ for every n is decidable. And so, a fortiori, also the $\exists$ theory of $\mathcal{D}$ with $0, \leq$ and all of these predicates. Indeed, they can even add $\vee$ to the language as long as 1 ($0'$) is omitted). It seems plausible that their methods (which are very complex) may suffice to prove the decidability of the $\exists$ theory of $\mathcal{D}(0, \leq, ')$ and perhaps even with $\vee$ added to the language.

As for the full $\forall\exists$ theory of $\mathcal{D}(\leq, ')$, there are many difficult open problems along the road to decidability (including, for example, controlling the jumps of initial segments). On the other hand, the path to undecidability seems quite dark. It is not immediately blocked by our general comments in the introduction on proving undecidability of $\forall\exists$ theories since we do have a function symbol $'$ that generates ω-sequences. The problem is how to define them (and relations on them) in a Δ_0 way using only $\leq$ and $'$.

The new area suggested by these charts is the decision problem for the $\forall\exists$ theory of $\mathcal{D}(\leq \mathbf{0}')$ with $\leq$ and $\vee$. This problem turns out to be far from straightforward. It would seem that our only hope is to prove decidability. We have the required initial segment results for decidability from the proof of decidability without $\vee$ (Lerman and Shore [1988]). The extension of embeddings part, however, has, surprisingly, something of the flavor of the $\forall\exists$ theory of $\mathcal{R}$.

Montalban has shown that we cannot reduce the full $\forall\exists$ decision problem to the extension of embedding problem as has been done in all the other successful proofs of decidability at the $\forall\exists$ level. (The extension of embedding problem for a structure $\mathcal{M}$ asks for a characterization of the partial orders $\mathcal{X} \subseteq \mathcal{Y}$ such that every embedding $f : \mathcal{X} \to \mathcal{M}$ can be extended to one $g : \mathcal{Y} \to \mathcal{M}$. The decision problem for $\forall\exists$ sentences about one of the degree structures is equivalent to deciding all problems of the form "every embedding of $\mathcal{X}$ can be extended to an embedding of some $\mathcal{Y}_i$ from a specified list. A reduction argument shows that the list can always be taken to have only one element.)

Proposition 4.1 *(Montalban) For every $\mathbf{x}_1 < \mathbf{x}_2$ in $(\mathbf{0}, \mathbf{0}')$ there is either a $\mathbf{y}$ such that $\mathbf{0} < \mathbf{y} < \mathbf{x}_1$ or one such that $\mathbf{x}_1 < \mathbf{y} < \mathbf{1}$ and $\mathbf{x}_2 \vee \mathbf{y} = \mathbf{1}$ but neither disjunct holds for every $\mathbf{x}_1 < \mathbf{x}_2$ in $(\mathbf{0}, \mathbf{0}')$.*

Proof. If $\mathbf{x}_1 \notin \mathbf{L}_2$ then it is not minimal and so there is a nonrecursive $\mathbf{y} < \mathbf{x}_1$. On the other hand, if $\mathbf{x}_1 \in \mathbf{L}_2$ then $\mathbf{0}'$ is high over $\mathbf{x}_1$ and so by the join theorem for high degrees (Posner [1977] or see Lerman [1983 IV.9]) there is a $\mathbf{y}$ which joins $\mathbf{x}_2$ up to $\mathbf{0}'$. However, there are counterexamples to each disjunct.

Of course, any minimal degree $\mathbf{x}_1 < \mathbf{0}'$ and any $\mathbf{x}_2$ between $\mathbf{x}_1$ and $0'$ supply a counterexample to the first disjunct. On the other hand, if we consider the construction of Slaman and Steel [1989] of nonrecursive r.e. degrees $\mathbf{a} < \mathbf{b}$ such that no degree $\mathbf{c} < \mathbf{b}$ joins $\mathbf{a}$ to $\mathbf{b}$ and apply the pseudojump inversion theorem of Jockusch and Shore [1983], we get $\mathbf{x}_1 < \mathbf{x}_2 < \mathbf{0}'$ (replacing $\mathbf{0} < \mathbf{a} < \mathbf{b}$) such that no $\mathbf{y}$ below $\mathbf{0}'$ and above $\mathbf{x}_1$ joins $\mathbf{x}_2$ up to $\mathbf{0}'$ as required. $\square$

This is reminiscent of the nondiamond (and other) phenomenon in $\mathcal{R}$. (The nondiamond theorem of Lachlan [1966] says that for every pair of incomparable r.e. degrees $\mathbf{x}_1, \mathbf{x}_2$ there is either a $\mathbf{y}$ with $\mathbf{x}_1, \mathbf{x}_2 \leq_T \mathbf{y} <_\mathbf{T} \mathbf{0}'$ or one with $0 <_T \mathbf{y} \leq_\mathbf{T} \mathbf{x}_1, \mathbf{x}_2$. On the other hand, there are $\mathbf{x}_1, \mathbf{x}_2$ such that $\mathbf{x}_1 \vee \mathbf{x}_2 = \mathbf{0}'$ and ones such that $\mathbf{x}_1 \wedge \mathbf{x}_2 = \mathbf{0}$.) It suggests that there are many problems to consider here.

5 Bibliography

Arslanov, M. M. and Lempp, S. eds. [1999], *Recursion theory and Complexity, Proceedings of the Kazan '97 Workshop*, Walter de Gruyter, Berlin.

Bernays, P. and Schoenfikel, M. [1928], Zum Entscheidungsproblem der mathematischen Logik, *Math. Ann.* **99**, 342-372.

Börger, E., Grädel, E. and Gurevich, Y. [1997], *The Classical Decision Problem*, Springer, Berlin.

Epstein, R. L. [1979], *Degrees of Unsolvability: Structure and Theory*, *LNM* **759**, Springer-Verlag, Berlin.

Friedberg, R. M. [1957], Two recursively enumerable sets of incomparable degrees of unsolvability, *Proc. Nat. Ac. Sci.* **43**, 236-238.

Harrington, L. and Shelah, S. [1982], The undecidability of the recursively enumerable degrees (research announcement), *Bull. Am. Math. Soc.*, N. S. **6**, 79-80.

Hinman, P. G. and Slaman, T. A. [1991], Jump embeddings in the Turing degrees, *J. Symbolic Logic* **56**, 563–591.

Jockusch, C. G. Jr. and Slaman, T. A. [1993], On the Σ_2-theory of the upper semilattice of Turing degrees, *J. Symbolic Logic* **58**, 193–204.

Jockusch, C. G. Jr. and Shore, R. A. [1983], Pseudojump operators. I. The r.e. case. *Trans. Amer. Math. Soc.* **275**, 599–609.

Jockusch, C. G. Jr. and Soare, R. I. [1972], Π^0_1 classes and degrees of theories, *Trans. Amer. Math. Soc.* **173**, 33–56.

Kleene, S. C. and Post, E. L. [1954] The upper semi-lattice of degrees of recursive unsolvability, *Ann. Math.* (2) **59**, 379–407

Lachlan, A. H. [1966], The impossibility of finding relative complements for recursively enumerable degrees, *J. Symbolic Logic* **31**, 434-454.

Lachlan, A. H. [1968], Distributive initial segments of the degrees of unsolvability, Z. Math. Logik Grundlagen Math. **14**, 457-472.

Lempp, S. and Lerman, M. [1996], The decidability of the existential theory of the poset of recursively enumerable degrees with jump relations, *Adv. Math.* **120**, 1–142.

Lempp, S., Nies, A. and Slaman, T. A. [1998], The Π_3-theory of the computably enumerable Turing degrees is undecidable, *Trans. Am. Math. Soc.* **350**, 2719-2736.

Lerman, M. [1971], Initial segments of the degrees of unsolvability, *Ann. Math.* (2) **93**, 365-389.

Lerman, M. [1983], *Degrees of Unsolvability*, Springer-Verlag, Berlin.

Lerman [1985], On the ordering of classes in high/low hierarchies, *Recursion theory week (Oberwolfach, 1984)*, H.-D. Ebbinghaus, G. H. Müller and G. E. Sacks eds., *LNM* **1141**, Springer, Berlin, 260–270.

Lerman, M. [2000] A necessary and sufficient condition for embedding principally decomposable finite lattices into the computably enumerable degrees, *Annals of Pure and Applied Logic* **101**, 275–297.

Lerman, M. and Shore, R. A [1988], Decidability and invariant classes for degree structures, *Trans. Am. Math. Soc.* **310**, 669-692.

Lerman, M., Shore, R. A. and Soare, R. I. [1984], The elementary theory of the recursively enumerable degrees is not $\aleph_0$-categorical, *Adv. Math.* **53**, 301-320.

Miller, R. G., Nies, A. O. and Shore, R. A. [2004], The $\forall\exists$-theory of $\mathcal{R}(\leq, \vee, \wedge)$ is undecidable, *Trans. Am. Math. Soc.*, to appear.

Minsky, M. L. [1961], Recursive unsolvability of Post's problem of tag and other topics in the theory of Turing machines, *Ann. Math.* **74**, 437-454.

Montalban, A. [2003], Embedding jump upper semilattices into the Turing degrees, *J. Symbolic Logic* **68**, 989-1014.

Montalban, A. [2004], There is no ordering on the classes in the generalized high/low hierarchy, in preperation.

Muchnik, A. A. [1956], On the unsolvability of the problem of reducibility in the theory of algorithms, *Dokl. Akad. Nauk SSSR* N.S. **108**, 29-32.

Nerode, A. and Shore, R. A. [1997], *Logic for Applications*, Springer, New York, 2nd ed.

Nies, A., Shore, R. A. and Slaman, T. A. [1998], Interpretability and definability in the recursively enumerable degrees, *Proc. London Math. Soc.* (3) **77**, 241-291.

Posner [1977], *High Degrees*, Ph.D. Thesis, Univeristy of California, Berkeley.

Ramsey, F. P. [1930], On a problem of formal logic, *Proc. London Math. Soc.* (2) **30**, 338-384.

Sacks, G. E. [1963], Recursive enumerability and the jump operator, *Trans. Am. Math. Soc.* **108**, 223-239.

Sacks, G. E. [1964], The recursively enumerable degrees are dense, *Ann. Math. (2)* **80**, 300-312.

Shepherdson, J. and Sturgis, H. [1963], Computability of recursive functions, *J. Assoc. Comp. Mach* **10**, 217-255.

Shore, R. A. [1978], On the $\forall\exists$-sentences of α-recursion theory, in *Generalized Recursion Theory II*, J. E. Fenstad, R. O. Gandy and G. E. Sacks eds., *Studies in Logic and the Foundations of Mathematics* **94**, North-Holland, Amsterdam, 331-354.

Shore, R. A. [1981], The theory of the degrees below $0'$, *J. London Math. Soc.* **24** (1981), 1-14.

Shore, R. A. [1999], The recursively enumerable degrees, in *Handbook of Computability Theory, Studies in Logic and the Foundations of Mathematics* **140**, North-Holland, Amsterdam, 169–197.

Shore, R. A. and Slaman, T. A. [1999], Defining the Turing jump, *Math. Res. Lett.* **6**, 711–722.

Simpson, S. G. [1977], First order theory of the degrees of recursive unsolvability, *Ann. Math. (2)*, **105**, 121-139.

Slaman, T. A. and Soare, R. I. [2001], Extension of embeddings in the computably enumerable degrees, *Ann. Math.*, **153**, 1-43.

Slaman, T. A and Steel, J. R. [1989], Complementation in the Turing degrees, *J. Symbolic Logic* **54**, 160–176.

Soare, R. I. [1987], *Recursively Enumerable Sets and Degrees*, Springer-Verlag, Berlin.

Spector, C. [1956], On degrees of recursive unsolvability, *Ann. Math. (2)* **64**, 581-592.