Complexity Theoretic Model Theory and Algebra

Douglas Cenzer
Department of Mathematics
University of Florida Gainesville, Fl 32611
e-mail: cenzer@math.ufl.edu
Jeffrey B. Remmel*
Department of Mathematics
University of California at San Diego
La Jolla, CA 92093
e-mail: jremmel@ucsd.edu

November 19, 1997

1 Introduction

In this paper, we will survey some recent results on complexity theoretic model theory and algebra. Essentially there are two major themes in this work. The first, which we call complexity theoretic model theory, deals with model existence questions. For example, given a recursive model $\mathcal{A}$, is there there a polynomial time (exponential time, polynomial space, etc.) model $\mathcal{B}$ which is isomorphic to $\mathcal{A}$. The second theme, which we call complexity theoretic algebra, fixes a given polynomial time structure and explores the properties of that structure. For example, we can ask whether every polynomial time ideal of a given polynomial time representation of the free Boolean algebra can be extended to a maximal polynomial time ideal. In both cases, one uses the rich theory of recursive model theory and algebra as a reference but looks at resource bounded versions of the results in those areas.

It turns out that not only are there a number of contrasts between results in recursive model theory and algebra and complexity theoretic model theory and algebra, but some new and interesting phenomena occur in the study of complexity theoretic model theory and algebra. That is, there are results in recursive model theory and algebra for which the natural complexity theoretic analogue is true but requires a more delicate proof which incorporates the resource bounds. There are also results in recursive model theory and algebra for

*Dept. of Commerce Agreement 70-NANB5H1164 and NSF grant DMS-9306427.

which the natural complexity theoretic analogue is false because the proof of the recursive result uses the unbounded resources allowed in recursive constructions in a crucial way. However, there are a number of interesting new phenomena which arise due to the fact that not all infinite polynomial time sets are polynomial time isomorphic or due to the fact that complexity theoretic results do not relativize as is the case for most recursion theoretic results. For example, in recursive model theory any two infinite recursive sets are recursively isomorphic, so that one can restrict one's attention to models whose universe is the set of natural numbers. It is not the case that any two infinite polynomial time sets are polynomial time isomorphic so that the choice of a particular universe, say the tally representation of the natural numbers versus the binary representation of the natural numbers, makes a difference.

Also, it is well known that the question of whether $P = NP$ is oracle dependent. That is, Baker, Gill and Solovay [4] proved that there are recursive oracles X and Y such that $P^X = NP^X$ and $P^Y \neq NP^Y$. We shall see that some of the natural complexity theoretic analogues of results in recursive algebra are oracle dependent as well.

There are several other areas of complexity theoretic model theory and algebra which will not be covered in this survey. There is the work of Friedman and Ko (see for example, [27], [42], and [43]) on polynomial time analysis, where complexity theoretic versions of various theorems of analysis are studied. Some of these results are oracle dependent and some are shown to be equivalent to $P = NP$. There is the work of Crossley, Nerode and Remmel on p-time equivalence types and p-time isols, as developed in [56], [57], [22] and [23]. We will present some results from [56] on p-time equivalence types in section 4. There is the work of Khoussainov and Nerode [41] on automatic, or automata presentable structures, which is a further restriction of polynomial time structures.

We will start with a survey of complexity theoretic model theory. In section 2, we will provide a general introduction to complexity theoretic model theory. In section 3, we shall give our basic complexity theoretic definitions and establish notation. In section 4, we will give a series of lemmas which are useful for building models with standard universes such as the binary representation of the natural numbers, $Bin(\omega)$, and the tally representation of the natural numbers, $Tal(\omega)$. In section 5 we provide a survey of the main existence theorems for feasible models. In section 6, we survey various feasible categoricity results. In section 7, we give an introduction to complexity theoretic algebra. Then in section 8, we focus on the structure of the binary and tally representation of an infinite dimensional vector space over a polynomial time field. In section 9, we look at the semilattice of NP ideals of the binary and tally representation of the free Boolean algebra. Finally in section 10, we give conclusions as well as some directions for further work.

2 Complexity Theoretic Model Theory

Complexity theoretic or feasible model theory is the study of resource-bounded structures and isomorphisms and their relation to computable structures and computable isomorphisms. The focus of complexity theoretic model theory in this paper is very different from classical complexity theory. A primary focus in classical complexity theory has been to determine the complexity of certain classes of finite models encoded as a decision problem. That is, one is interested in classifying decision problems as being in P, NP, $PSPACE$, etc. A typical example is the graph-coloring problem, where it is known that the family of finite graphs which can be 3-colored is NP-complete.

Complexity theoretic model theory is more concerned with infinite models whose universe, functions, and relations are in some well known complexity class such as polynomial time, exponential time, polynomial space, etc. Thus if one studies graph colorings from this point of view, one would study the complexity of graph colorings in an infinite polynomial time graph as was done by Cenzer and Remmel in [13]. However complexity theoretic model theory has been more concerned with the complexity of the model itself. Thus one can pick any complexity class and ask questions about what structures can be represented by models in that complexity class. By far, the complexity class that has received the most attention is polynomial time. The basic questions that have been consider are to classify which recursive models are isomorphic or recursively isomorphic to a polynomial time model.

To establish some notation, let $\omega = \{0, 1, \ldots\}$ denote the set of natural numbers. Let $[,]$ denote the usual quadratic-time pairing function $[m, n] = m + \frac{1}{2}(m+n)(m+n+1)$, which maps $\omega \times \omega$ onto ω. Let $\phi_{e,n}$ denote the n-ary partial function on $(\{0,1\}^*)^n$ computed by the e-th Turing machine. Then we say that a structure

$$\mathcal{A} = (A, \{R_i^{\mathcal{A}}\}_{i \in S}, \{f_i^{\mathcal{A}}\}_{i \in T}, \{c_i^{\mathcal{A}}\}_{i \in U}),$$

(where the universe A of $\mathcal{A}$ is a subset of $\{0,1\}^*$) is *recursive* if A is a recursive subset of $\{0,1\}^*$, S, T, and U are initial segments of ω, the set of relations $\{R_i^{\mathcal{A}}\}_{i \in S}$ is uniformly recursive in the sense that there is a recursive function G such that for all $i \in S$, $G(i) = [n_i, e_i]$ where $R_i^{\mathcal{A}}$ is an n_i-ary relation and ϕ_{e_i,n_i} computes the characteristic function of $R_i^{\mathcal{A}}$, the set of functions $\{f_i^{\mathcal{A}}\}_{i \in T}$ is uniformly recursive in the sense that there is a recursive function F such that for all $i \in T$, $F(i) = [n_i, e_i]$ where $f_i^{\mathcal{A}}$ is an n_i-ary function and ϕ_{e_i,n_i} restricted to A^{n_i} computes $f_i^{\mathcal{A}}$, and there is a recursive function interpreting the constant symbols in the sense that there is a recursive function H such that for all $i \in U$, $H(i) = c_i^{\mathcal{A}}$. Note that if $\mathcal{A}$ is a recursive structure, then the atomic diagram of $\mathcal{A}$ is recursive.

We say that a recursive structure $\mathcal{A} = (A, \{R_i^{\mathcal{A}}\}_{i \in S}, \{f_i^{\mathcal{A}}\}_{i \in T}, \{c_i^{\mathcal{A}}\}_{i \in U})$, is *polynomial time* if A is a polynomial time subset of $\{0,1\}^*$ and the set of relations $\{R_i^{\mathcal{A}}\}_{i \in S}$ and the set of functions $\{f_i^{\mathcal{A}}\}_{i \in T}$ are uniformly polynomial time in the sense that, in addition to the functions G and F defined above, there are recursive functions G' and F' such that for $i \in S$, $G'(i) = m_i$ where for all $(x_1, \ldots, x_{n_i})$ in $(\{0,1\}^*)^{n_i}$, it takes at most $(max\{2, |x_1|, \ldots, |x_{n_i}|\})^{m_i}$ steps to compute $\phi_{e_i,n_i}(x_1, \ldots, x_{n_i})$ and for all $i \in T$, $F'(i) = q_i$ where for all $(x_1, \ldots, x_{n_i})$ in $(\{0,1\}^*)^{n_i}$, it takes at most $(max\{2, |x_1|, \ldots, |x_{n_i}|\})^{q_i}$ steps to compute $\phi_{e_i,n_i}(x_1, \ldots, x_{n_i})$. Note that if $\mathcal{A}$ is a polynomial time structure with infinitely many relation symbols or with infinitely many function symbols, then our definition of a polynomial time structure does not ensure that the atomic diagram of $\mathcal{A}$ is polynomial time. Thus we say $\mathcal{A}$ is *uniformly polynomial time* if the atomic diagram of $\mathcal{A}$ is polynomial time. Note that the fact that $\mathcal{A}$ is uniformly polynomial time implies, among other things, that the sequence of run times $\{x^{m_i} : i \in S\}$ and $\{x^{q_i} : i \in T\}$ are bounded by some fixed polynomial. Of course, if $\mathcal{A}$ is a structure over a finite language, then $\mathcal{A}$ is a polynomial time structure if and only if $\mathcal{A}$ is a uniformly polynomial time structure. Similar definitions may be given for other resource-bounded classes.

There are two basic types of questions which have been studied in polynomial time model theory. First, as discussed above, there is the basic existence problem, i.e. whether a given infinite recursive structure $\mathcal{A}$ isomorphic or recursively isomorphic to a polynomial time model. For example, the authors showed in [10] (p. 24) that every recursive relational structure is recursively isomorphic to a polynomial time model and that the standard model of arithmetic $(\omega, +, -, \cdot, <, 2^x)$ with addition, subtraction, multiplication, order and the 1-place exponential function is isomorphic to a polynomial time model. The fundamental effective completeness theorem says that any decidable theory has a decidable model. It follows that any decidable relational theory has a polynomial time model. However, one is naturally led to ask more refined existence questions in complexity theoretic algebra than one asks in recursive algebra. That is, since all infinite recursive sets are recursively isomorphic, it is easy to see that any infinite recursive structure is recursively isomorphic to a recursive structure whose universe is ω. It is certainly not the case that any two infinite polynomial-time sets are polynomial-time isomorphic. For example, the tally representation of the natural numbers is not polynomial time isomorphic to the binary representation of the natural numbers. Hence it no longer the case that any infinite polynomial-time structure can be identified with a polynomial-time structure whose universe is $\{0,1\}^*$. Thus a more refined existence questions is to take a fixed universe, such as the tally representation of the natural numbers or the binary representation of the natural numbers, and ask if a recursive model is isomorphic or recursively isomorphic to a polynomial time model with that given universe.

Here are two examples which illustrate both the negative and positive outcomes to the simplest existence type question, i.e., whether a given recursive model is isomorphic to a polynomial time model.

Example 2.1 *Let* $\mathcal{A} = (A, 0, S, R)$ *where* $A = \{1\}^*$ *(that is, the set of natural* *numbers in unary representation), S is the successor function, (that is, $S(1^n) = 1^{n+1}$), and R is a unary relation, (that is, a subset of $\{1\}^*$). Now if $\mathcal{A}$ is isomorphic to a polynomial-time structure $\mathcal{B} = (B, 0^B, S^B, R^B)$, then we can test for membership in R as follows. Given 1^n, compute $(S^B)^n(0^B) = y_n$ and then test whether y_n is in R^B. Now if we assume that we can compute $S^B(x)$ in $|x|^k$ steps for $|x| \geq 2$, then it takes at most $\Sigma_{i=1}^n |0^B|^{k^i} \leq |0^B|^{k^n+1}$ steps to compute y_n. Next we may assume that testing whether $x \in R^B$ takes $|x|^r$ steps if $|x| \geq 2$, so that it takes at most $|0^B|^{r(k^n+1)}$ steps to test whether 0^n is in R. This means that R is a doubly-exponential-time set. Thus if we start with any recursive structure $\mathcal{A} = (A, 0, S, R)$ where R is a recursive set but is not doubly exponential-time, then $\mathcal{A}$ is not even isomorphic, much less recursively isomorphic, to a polynomial-time structure.*

Despite this example, there are lots of recursive structures which are recursively isomorphic to polynomial-time structures.

Example 2.2 *Let $\mathcal{A} = (A, f)$, where $A = \{1\}^*$ and f is a unary function. We say that 1^m and 1^n are in the same f-orbit if, for some $k \geq 0$, either $f^k(1^m) = 1^n$ or $f^k(1^n) = 1^m$. If f is length-increasing, then it is clear that each f-orbit is isomorphic to (A,S). Now let f and g be any two recursive length-increasing functions from $\{1\}^*$ into $\{1\}^*$. Then the structures (A,f) and (A,g) are recursively isomorphic if and only if they have the same number of orbits. Thus, for example, we can let $f(1^n) = 1^{a(n)}$ where a is Ackermann's function and still be guaranteed that (A,f) is recursively isomorphic to a polynomial-time structure.*

Next consider the more restricted kind of existence question, i.e. whether a given recursive model is isomorphic or recursively isomorphic to a polynomial time model which has a standard universe such as the binary representation of the natural numbers, $Bin(\omega)$, or the tally representation of the natural numbers, $Tal(\omega) = \{1^n : n \in \omega\}$. Grigorieff [32] proved that every recursive linear ordering is isomorphic to a linear time linear ordering which has universe $Bin(\omega)$. However Grigorieff's result can not be improved to the result that every recursive linear ordering is **recursively isomorphic** to a linear time linear ordering over $Bin(\omega)$. That is, Cenzer and Remmel [10] (p. 25) showed that for any infinite polynomial time set $A \subseteq \{0,1\}^*$, there exists a recursive copy of the linear ordering $\omega + \omega^*$ which is not recursively isomorphic to any polynomial time linear ordering which has universe A. Here $\omega + \omega^*$ is the ordering obtained by taking a copy of $\omega = \{0, 1, 2, \ldots\}$ under the usual ordering followed by a copy of the negative integers under the usual ordering.

The general problem of determining which recursive models are isomorphic or recursively isomorphic to feasible models has been studied by the authors in [10], [11], and [14]. For example, it was shown in [11] (pp. 343-348) that any recursive torsion Abelian group G is isomorphic to a polynomial time group A

and that if the orders of the elements of G are bounded, then A may be taken to have a standard universe, i.e. either $Bin(\omega)$ or $Tal(\omega)$. It was also shown in [11] (p. 357) that there exists a recursive torsion Abelian group which is not even isomorphic to any polynomial time (or any primitive recursive) group with a standard universe. Feasible linear orderings were studied by Grigorieff [32], by Cenzer and Remmel [10], and by Remmel [68, 69]. Feasible vector spaces were studied by Nerode and Remmel in [53] and [55]. Feasible Boolean algebras were studied by Cenzer and Remmel in [10] and by Nerode and Remmel in [54]. Feasible permutation structures and feasible Abelian groups were studied by Cenzer and Remmel in [11] and [14]. By a *permutation structure* $\mathcal{A} = (A, f)$, we mean a set A together with a unary function f which maps A one-to-one and onto A. Similarly an *equivalence structure* $\mathcal{A} = (A, R^{\mathcal{A}})$ consists of a set A together with an equivalence relation.

The second basic type of problem studied in polynomial time model theory is the problem of feasible categoricity. Here we say that a recursive model $\mathcal{A}$ is *recursively categorical* if any other recursive model isomorphic to $\mathcal{A}$ is in fact recursively isomorphic to $\mathcal{A}$. The notion of recursive categoricity was first defined by Mal'cev [49] and is referred to in the Russian literature as *autostability*. Recursively categorical structures have been widely studied in the literature of recursive algebra and recursive model theory.

The recursively categorical structures for various theories have been classified, including Boolean algebras independently by Goncharov [30] and LaRoche [44], Abelian groups by Smith [74] and linear orderings independently by Dzgoev [31] and Remmel [66]. For example, Remmel showed in [66] that a recursive linear ordering $L = (D, <)$ is recursively categorical if and only if L has only finitely many successivities, where a pair $a < b$ is a successivity if there is no c with $a < c < b$.

Defining a natural analogue of feasible categoricity is complicated by the fact that unlike the case of infinite recursive models, where any two infinite recursive universes are recursively isomorphic, it is not the case that any two polynomial time universes are polynomial time isomorphic. It turns out to be more natural to define polynomial categorical structures with respect to a fixed universe. Thus we say that a p-time structure $\mathcal{A}$ with universe $D \subseteq \{0,1\}^*$ is *p-time categorical with respect to* D if every p-time structure $\mathcal{B}$ with universe D which is isomorphic to $\mathcal{A}$ is necessarily p-time isomorphic to $\mathcal{A}$, i.e. there exist polynomial time functions f, g such that f restricted to D is an isomorphism from $\mathcal{A}$ onto $\mathcal{B}$ and g restricted to D is an isomorphism from $\mathcal{B}$ onto $\mathcal{A}$.

Remmel showed in [66] that there are no p-time categorical linear orderings with respect to the standard universes $Bin(\omega)$ and $Tal(\omega)$. There are two parts to this strongly negative statement. For any p-time linear ordering L with universe B (either $Bin(\omega)$ or $Tal(\omega)$), there is a p-time linear ordering L' with universe B which is not primitive recursively isomorphic to L. Furthermore, if L is not recursively categorical, then L' is not even recursively isomorphic to L. Similar results will be shown for other structures. The problem of feasible categoricity for permutation structures and torsion Abelian groups was studied by Cenzer and Remmel in [14]. Here there are some limited positive results. In particular, a permutation structure (A, f) such that all orbits of f have the same finite size is p-time categorical over $Tal(\omega)$. There are also structures $\mathcal{A}$ which are not p-time categorical over B, but such that any p-time structure $\mathcal{D}$ with universe B which is isomorphic to $\mathcal{A}$ must be exponential time isomorphic or double exponential time isomorphic to $\mathcal{A}$. More generally, we can define a larger notion of feasibility, *q-time* or iterated exponential time computability, and show that there are many natural structures which are q-time categorical over $Bin(\omega)$ and $Tal(\omega)$.

General semantic conditions for when a decidable model is recursively categorical were given by Nurtazin [60] and Goncharov [29] and similar results were found by Ash and Nerode [2] for models in which one can effectively decide all Σ_1 formulas. These methods are based on the existence of a so-called *Scott family* of formulas. We discuss in section 6 various notions from [15] of a feasible Scott family of formulas for a feasible model and show that any two families which possess a common Scott family and have the same universe B are feasibly isomorphic. Structures considered in [15] include linear orderings, permutation structures, Abelian groups and equivalence structures.

3 Preliminaries

In this section, we will give the basic definitions from complexity theory which will be needed for the rest of the paper.

Let Σ be a finite alphabet. Then Σ^* denotes the set of finite strings of letters from Σ and Σ^ω denotes the set of infinite strings of letters from Σ where $\omega = \{0, 1, 2, \ldots\}$ is the set of natural numbers. For any natural number $n \neq 0$, $tal(n) = 1^n$ is the tally representation of n and $bin(n) = i_0 i_1 \ldots i_e \in \{0,1\}^*$ is the (reverse) binary representation of n if $n = i_0 + 2 \cdot i_1 + \cdots + 2^e \cdot i_e$ and $i_e \neq 0$. In general, the k-ary representation $b_k(n) = i_0 i_1 \ldots i_e$ if $n = i_0 + i_1 \cdot k + \cdots i_e \cdot k^e$ and $i_e \neq 0$. We let $tal(0) = bin(0) = b_k(0) = 0$. Then we let $Tal(\omega) = \{tal(n) : n \in \omega\}$, $Bin(\omega) = \{bin(n) : n \in \omega\}$ and, for each $k \geq 3$, $B_k(\omega) = \{b_k(n) : n \in \omega\}$. Occasionally, we will want to say that $B_2(\omega) = Bin(\omega)$ and that $B_1(\omega) = Tal(\omega)$.

For a string $\sigma = (\sigma(0), \sigma(1), \ldots, \sigma(n-1))$, $|\sigma|$ denotes the length n of σ. The empty string has length 0 and will be denoted by $\emptyset$. A constant string σ of length n will be denoted by k^n. For $m < |\sigma|$, $\sigma\lceil m$ is the string $(\sigma(0), \ldots, \sigma(m-1))$; σ is an *initial segment* of τ (written $\sigma \prec \tau$) if $\sigma = \tau\lceil m$ for some m. The *concatenation* $\sigma^\frown\tau$ (or sometimes just $\sigma\tau$) is defined by

$$\sigma^\frown\tau = (\sigma(0), \sigma(1), \ldots, \sigma(m-1), \tau(0), \tau(1), \ldots, \tau(n-1)),$$

where $|\sigma| = m$ and $|\tau| = n$; in particular we write $\sigma^\frown a$ for $\sigma^\frown(a)$ and $a^\frown\sigma$ for $(a)^\frown\sigma$.

Our basic computation model is the standard multitape Turing machine of Hopcroft and Ullman [5]. Note that there are different heads on each tape and that the heads are allowed to move independently. This implies that a string σ can be copied in linear time. An oracle machine is a multitape Turing machine M with a distinguished work tape, a query tape, and three distinguished states QUERY, YES, and NO. At some step of a computation on an input string σ, M may transfer into the state QUERY. In state QUERY, M transfers into the state YES if the string currently appearing on the query tape is in an oracle set A. Otherwise, M transfers into the state NO. In either case, the query tape is instantly erased. The set of strings accepted by M relative to the oracle set A is $L(M,A) = \{\sigma|$ there is an accepting computation of M on input σ when the oracle set is $A\}$. If $A = \emptyset$, we write $L(M)$ instead of $L(M,\emptyset)$.

Let *t(n)* be a function on natural numbers. A Turing machine M is said to be *t(n) time bounded* if each computation of M on inputs of length n where $n \geq 2$ requires at most $t(n)$ steps. A function $f(x)$ on strings is said to be in $DTIME(t)$ if there is a $t(n)$-time bounded deterministic Turing machine M which computes $f(x)$. For a function f of several variables, we let the length of $(x_1,\ldots,x_n)$ be $|x_1| + \cdots + |x_n|$. A set of strings or a relation on strings is in $DTIME(t)$ if its characteristic function is in $DTIME(t)$. We let

$R = \bigcup_c \{DTIME(n+c) : c \geq 0\}$,
$LIN = \bigcup_c \{DTIME(cn) : c \geq 0\}$,
$P = \bigcup_i \{DTIME(n^i) : i \geq 0\}$,
$DEXT = \bigcup_{c\geq 0}\{DTIME(2^{c\cdot n})\}$, and
$DOUBEXT = \bigcup_{c\geq 0}\{DTIME(2^{2^{c\cdot n}})\}$,
$EXPTIME = \bigcup_{c\geq 0}\{DTIME(2^{n^c})\}$, and in general,
$DEX(S) = \bigcup_{t(n)\in S} DTIME(2^{t(n)})\}$.

A function $f(x)$ on strings is said to be in $NTIME(t)$ if there is a $t(n)$-time bounded nondeterministic Turing machine M which computes $f(x)$. A set of strings or a relation on strings is in $NTIME(t)$ if its characteristic function is in $NTIME(t)$. We let

$NP = \bigcup_i \{NTIME(n^i) : i \geq 0\}$,
$NEXT = \bigcup_{c\geq 0}\{NTIME(2^{c\cdot n})\}$,
$NEXPTIME = \bigcup_{c\geq 0}\{NTIME(2^{n^c})\}$,
$DOUBNEXT = \bigcup_{c\geq 0}\{NTIME(2^{2^{c\cdot n}})\}$ and in general,
$NEX(S) = \bigcup_{t(n)\in S} NTIME(2^{t(n)})\}$.

We fix enumerations $\{P_i\}_{i\in N}$ and $\{N_i\}_{i\in N}$ of the polynomial time bounded deterministic oracle Turing machines and the polynomial time bounded nondeterministic oracle Turing machines respectively. We may assume that $p_i(n) = max(2,n)^i$ is a strict upper bound on the length of any computation by P_i or N_i with any oracle X on inputs of length n. P_i^X and N_i^X denote the oracle Turing machine using oracle X and in an abuse of notation we shall denote $L(P_i, X)$

by simply P_i^X and $L(N_i, X)$ by N_i^X. This given, $P^X = \{P_i^X : i \in N\}$ and $NP^X = \{N_i^X : i \in N\}$.

For A, $B \subset \Sigma^*$, we shall write $A \leq_m^P B$ if there is a polynomial-time function f such that for all $x \in \Sigma^*$, $x \in A$ iff $f(x) \in B$. We shall write $A \leq_T^P B$ if A is polynomial time Turing reducible to B. For r equal to m or T, we write $A \equiv_r^P B$ if $A \leq_r^P B$ and $B \leq_r^P A$ and we write $A \,|_r^P B$ if not $A \leq_r^P B$ and not $B \leq_r^P A$.

We define the standard notions of feasibility as follows. We say that a function $f(x)$ is *quasi-real-time* if $f(x) \in R$. (This is slightly more general than the usual notion of real-time as computable by a Turing machine which simply reads the input one symbol at a time from left to right (or right to left) and simultaneously leaves the output in its place on the tape. In particular, a real-time function is always in $DTIME(n)$.) The function $f(x)$ is *linear time* if $f(x) \in L$, *polynomial time* if $f(x) \in P$, *nondeterministic polynomial time* if $f(x) \in NP$, *exponential time* if $f(x) \in DEXT$, *nondeterministic exponential time* if $f(x) \in NEXT$, and is *double exponential time* if $f(x) \in DOUBEXT$. We say that $f(x)$ is exponentially feasible if $f(x) \in DEX(T)$ for a notion T of feasibility. In particular, if $f(x) \in DEX(DOUBEXT)$, then $f(x)$ is said to be *triple exponential time.*

The smallest class including P and closed under DEX can be defined by iterating DEX. That is, let

$P^0 = P$, $P^{n+1} = DEX(P)$ for each n, and $Q = \bigcup_{n<\omega} P^n$.

A function $f(x) \in Q$ is said to be *iterated exponential time* or *q-time*. The iterated exponential functions $E_n(x)$ can be defined recursively by $E_0(x) = x$ and $E_{n+1}(x) = 2^{E_n(x)}$ for all n and x. It is easy to see that $x^r \leq E_r(x)$ for all $r > 0$, from which it follows that $Q = \bigcup_{m<\omega} DEX^m(DTIME(n))$.

We observe that the classes R, LIN, P, NP, and Q are all closed under composition, whereas the other classes defined above are not. In addition, $Tal(\omega)$ and $Bin(\omega)$ are q-time isomorphic.

Observe that for a function $f(x_1, \ldots, x_k)$ of several variables, the above definitions are equivalent, if we declare the size of the input $(x_1, \ldots, x_k)$ to be the maximum of the sizes $|x_1|, \ldots, |x_k|$ since we allow multiple tapes. This occasionally simplifies the computation of the complexity of various functions.

We refer the reader to Odifreddi [61] for the basic definitions of recursion theory. Let $\phi_{i,n}$ be the partial recursive function of n variables computed by the i^{th} Turing machine M_i. If $n = 1$, we will write ϕ_i instead of $\phi_{i,1}$. Given a string $\sigma \in \{0,1\}^*$, we write $\phi_i^s(\sigma) \downarrow$ if M_i gives an output in s or fewer steps when started on input string σ. Thus the function ϕ_i^s is uniformly polynomial time. We write $\phi_e(\sigma) \downarrow$ if $(\exists s)(\phi_e^s(\sigma) \downarrow)$ and $\phi_e(\sigma) \uparrow$ if not $\phi_e(\sigma) \downarrow$.

The notion of a p-time structure was defined in section 2. We need a few refinements of that definition.

Definition 3.1 *(i) A p-time function f is* **honest p-time** *if there is a polynomial function q such that for all $x_1, \ldots, x_n$,*

$$y = f(x_1, \ldots, x_n) \rightarrow (\forall i \leq n)(|x_i| \leq q(|y|)).$$

(ii) A p-time structure $\mathcal{A}$ is **honest p-time** *if all of its functions are honest p-time.*

(iii) A structure $\mathcal{A}$ has **honest witnesses** *if for any quantifier-free formula $\phi(y, x_1, \ldots, x_n)$, there is a polynomial q such that for any $a_1, \ldots, a_n \in A$, if $\mathcal{A} \models (\exists y)\phi(y, a_1, \ldots, a_n)$, then there is a $z \in A$ with $|z| \leq q(|a_1| + \cdots + |a_n|)$ such that $\mathcal{A} \models \phi(z, a_1, \ldots, a_n)$.*

Note that for an honest p-time function mapping $Tal(\omega)$ into $Tal(\omega)$, Nerode and Remmel showed in [56] that f^{-1} is also honest p-time.

For a group, we will distinguish two types of computability. The structure of a group $\mathcal{G}$ is determined by the binary operation which we will denote by the addition sign $+^G$, since we are interested in Abelian groups. We let e^G denote the additive identity of $\mathcal{G}$. However, the inverse operation, denoted by inv^G, may also be included as an inherent part of the group. Thus we have the following distinction.

Definition 3.2 *A group $\mathcal{G}$ is* **Γ-computable** *if $(G, +^G, e^G)$ is Γ-computable, and is* **fully Γ-computable** *if $(G, +^G, inv^G, e^G)$ is Γ-computable.*

It is easy to see that any recursive group is also fully recursive, since $inv^G(a)$ can be computed as the least member b of G such that $a +^G b = e^G$, where the elements of G are ordered first by length and then lexicographically for elements of the same length.

On the other hand, the fully p-time groups make up a proper subclass of the p-time groups, as shown by Proposition 1.1 of [11].

Definition 3.3 *For any complexity class Γ and any structures*

$$\mathcal{A} = (A, \{R_i^A\}_{i \in S}, \{f_i^A\}_{i \in T}, \{c_i\}_{i \in U}),$$

and

$$\mathcal{B} = (B, \{R_i^B\}_{i \in S}, \{f_i^B\}_{i \in T}, \{c_i\}_{i \in U}),$$

we say that $\mathcal{A}$ and $\mathcal{B}$ are **Γ-isomorphic** *if there is an isomorphism f from $\mathcal{A}$ onto $\mathcal{B}$ and Γ-computable functions F and G such that $f = F\lceil A$ (the restriction of F to A) and $f^{-1} = G\lceil B$.*

4 Polynomial time sets and isomorphisms

In this section we shall give a number of useful lemmas about the relations between the various standard universes that we will consider in our study of feasible structures. The most basic standard universe is the set Σ^* where Σ is a finite alphabet and in particular where $\Sigma = \{0, 1\}$, $\{1\}$, or $\{0\}$. Other standard universes include the set $Tal(\omega)$ of tally representations of natural numbers, the set $Bin(\omega)$ of binary representations of natural numbers, and, for any k, the set $B_k(\omega)$ of k-ary representations of natural numbers. In recursion theory, all of these sets are recursively isomorphic and therefore interchangeable. For our purposes, we must consider carefully which of these isomorphisms are polynomial time or even polynomial time in one direction.

First we need to explicitly define a polynomial time pairing function. For any finite alphabet Σ, there is a natural embedding ρ of Σ^* into $Bin(\omega)$ given as follows. We may suppose that $\Sigma \subset \{0, 1, 2, \ldots, n\}$ for some n. Let $\rho(\emptyset) = 0$ and, for $\sigma = (i_1, \ldots, i_k)$, let

$$\rho(\sigma) = 0^{i_1} {}^\frown 1 {}^\frown 0^{i_2} {}^\frown 1 {}^\frown \cdots {}^\frown 0^{i_k} {}^\frown 1.$$

The function ρ is actually an isomorphism from ω^* onto $Bin(\omega)$ and has an inverse ρ^{-1}. It is also clear that, for each n, the set $\rho(\{0, 1, \ldots, n\}^*)$ is linear time (uniformly in n). Thus we can normally assume that an arbitrary structure has universe a subset of $Bin(\omega)$.

The coding function $\langle \sigma_1, \sigma_2, \ldots, \sigma_k \rangle_k$ for $\sigma_1, \ldots, \sigma_k \in \{0, 1\}^*$ is now defined by

$$\langle \sigma_1, \sigma_2, \ldots, \sigma_k \rangle_k = \rho(\sigma_1 {}^\frown 2 {}^\frown \sigma_2 {}^\frown 2 {}^\frown \ldots {}^\frown \sigma_{k-1} {}^\frown 2 {}^\frown \sigma_k).$$

Let $Q_k = \{\langle \sigma_1, \sigma_2, \ldots, \sigma_k \rangle_k : \sigma_i \in \{0, 1\}^* \text{ for each i}\}$. For $i = 1, \ldots k$, the projection functions π_i^k from Q_k onto $\{0, 1\}^*$ are implicitly defined by the equation

$$\sigma = \langle \pi_1^k(\sigma), \pi_2^k(\sigma), \ldots, \pi_k^k(\sigma) \rangle_k.$$

The subscript k will normally be omitted. It is easy to see that the sets Q_k and $B_k(\omega)$ are all linear time and that the functions π_i^k, and $\langle \quad , \ldots , \quad \rangle_k$ are all computable in linear time.

Given two subsets A and B of $\{0, 1\}^*$, define

$$A \otimes B = \{\langle a, b \rangle : a \in A, b \in B\}$$

and

$$A \oplus B = \{\langle 0, a \rangle : a \in A\} \cup \{\langle 1, b \rangle : b \in B\}.$$

It is clear that if A and B are p-time, then both $A \oplus B$ and $A \otimes B$ will also be p-time.

Now, for each $k \geq 2$, a natural number in (reverse) k-ary form is simply a string $\sigma \in \{0, 1, \ldots, k-1\}^*$ which is either 0 or else ends with an element of

$\{1, \ldots, k-1\}$. Thus the set $B_k(\omega)$ of k-ary representations of natural numbers is a linear time subset of $\{0, 1, \ldots, k-1\}^*$. $Tal(\omega) = \{0\} \cup \{1\}^*1$ is of course a linear time subset of $\{0, 1\}^*$, but is not polynomial time isomorphic to the whole set. (This will follow immediately from Lemma 4.6 below.) For a string $\sigma = b_k(n) \in B_k(\omega)$ with $n > 0$, let $\mu_k(\sigma)$ be the unary representation, 1^n, of n and let $\mu_k(0) = 0$. We now state a sequence of lemmas which will be useful for our main results. Most of these lemmas are proved in [11] or [14].

Lemma 4.1 **(a)** *Suppose that $\mathcal{A}$ is a polynomial time structure and that ϕ is a polynomial time set isomorphism from A onto a set B. Then $\mathcal{B}$ is a polynomial time structure, where the functions and relations on B are defined to make ϕ an isomorphism of the structures.*

(b) *Suppose that $\mathcal{A}$ is an EXPTIME structure and that ϕ is a polynomial time set isomorphism from A onto a set B. Then $\mathcal{B}$ is an EXPTIME structure, where the functions and relations on B are defined to make ϕ an isomorphism of the structures.*

(c) *Suppose that $\mathcal{A}$ is a q-time time structure and that ϕ is a q-time time set isomorphism from A onto a set B. Then $\mathcal{B}$ is a q-time time structure, where the functions and relations on B are defined to make ϕ an isomorphism of the structures.*

Proof: We sketch the proof for the p-time case. The other cases are similar. To simplify the proof, let us suppose that A has one function f^A and one relation R^A. Observe first that B is a polynomial time set, since $b \in B \iff \phi^{-1}(b) \in A$. The function f^B is polynomial time, since $f^B(b_1, \ldots, b_n) = \phi(f^A(\phi^{-1}(b_1), \ldots, \phi^{-1}(b_n)))$. The relation R^B is polynomial time, since $R^B(b_1, \ldots, b_n) \iff R^A(\phi^{-1}(b_1), \ldots, \phi^{-1}(b_n))$. □

Next we state two lemmas which relate tally representation of a structure to its binary representation and, more generally, to its k-ary representation. Part (b) of the first lemma is an improvement of Lemma 2.2 of [11] where the computation was bounded in polynomial time.

Lemma 4.2 *For each $k > 1$,*

(a) *$Bin(\omega)$ is linear time isomorphic to $\{0, 1\}^*$.*

(b) *There is a linear time function p such that, for all n, both the computation of $\mu_k(b_k(n)) = 1^n$ and the inverse computation of $\mu_k^{-1}(1^n) = b_k(n)$ can be computed in time $p(n)$;*

(c) *For each $n > 0$ and $\sigma = b_k(n)$, $k^{|\sigma|-1} < n + 1 \leq k^{|\sigma|}$*

Proof: We sketch the proof of (b) for $k = 2$. The basic computation in either direction consists of enumerating the binary numbers from 1 to n on one tape while either writing or reading n 1's on the other tape. The enumeration of the binary numbers is done by repeatedly adding 1 by the usual algorithm, which consists of replacing 1's with 0's while looking for the first 0, and then replacing the first 0 with a 1. If we define $h(n)$ to be the total number of symbols written by this procedure for the numbers from $k = 1$ up to $k = 2^n - 1$, then we observe that $h(1) = 1$ and that, in general, $h(n+1) = n+1+2h(n)$. This is because the numbers from 2^n up to $2^{n+1} - 1$ are obtained by writing 2^n, which takes $n+1$ steps and then essentially writing the numbers from 1 up to $2^n - 1$ again, while leaving the final 1 on the end of each. Now it is easy to see by induction that $h(n) \leq 2^{n+1} - n - 2$ for each n. Counting a slightly smaller number of steps for returning to the beginning of the string, we see that all of the binary numbers from 1 up to $k = 2^n - 1$ may be written in total time $\leq 2h(n) \leq 2^{n+1} - 2 = 4k$. For any number k with $2^{n-1} < k \leq 2^n$, the binary numbers from 1 to k may be written in total time $\leq 2^{n+1} - 2 \leq 8k$. □

For any subset M of ω, let $tal(M) = \{tal(n) : n \in M\}$, let $bin(M) = \{bin(n) : n \in M\}$, and for any finite $k > 1$, let $b_k(M) = \{b_k(n) : n \in M\}$.

Lemma 4.3 *For any finite $k > 1$, any $M \subset \omega$ and any oracle X:*

(a) $tal(M) \in P^X \iff b_k(M) \in DEXT^X$;

(b) $tal(M) \in NP^X \iff b_k(M) \in NEXT^X$.

(c) $b_k(M) \in P^X \rightarrow tal(M) \in LIN^X$.

Proof: We give the proof for $k = 2$, where $b_k(M) = bin(M)$. The proof of (b) is the same as the proof of (a).

(a) Suppose first that $tal(M) \in P^X$. Then there is a procedure with oracle X which tests whether $tal(n) \in tal(M)$ in time $\leq n^c$ for some fixed c and all $n \geq 2$. To test whether $\sigma = bin(n) \in bin(M)$, we first compute $tal(n)$, which requires time $\leq 8n$ by the proof of Lemma 4.2. Then we test whether $tal(n) \in tal(M)$, which requires $\leq n^c$ steps by assumption. Now by part (c) of Lemma 4.2, $n^c < (2^{|\sigma|})^c = 2^{c|\sigma|}$ so that $bin(M) \in DEXT^X$.

Next suppose that $bin(M) \in DEXT^X$. Then there is a procedure with oracle X which tests whether $bin(n) \in bin(M)$ in time $\leq 2^{c|bin(n)|}$ for some fixed c and all n. To test whether $tal(n) \in tal(M)$, we first compute $bin(n)$, which requires time $\leq 8n$ by the proof of Lemma 4.2. Then we test whether $bin(n) \in bin(M)$, which requires $\leq 2^{c|bin(n)|}$ steps by assumption. Now by part (c) of Lemma 4.2, $2^{c|bin(n)|} = (2^{|bin(n)|})^c \leq (2n)^c \leq min(2,n)^{2c}$. Hence $tal(M) \in P^X$.

(c) Suppose that $bin(M) \in P^X$, so that we can test $\sigma = bin(n) \in bin(M)$ in time $\leq |\sigma|^c$. Now let $|\sigma| = r$, so that $2^{r-1} \leq n < 2^r$. Then as in part (a) above, we see that we can test $tal(n) \in tal(M)$ in time $\leq r^c$. Now it is clear that for sufficiently large r, $r^c \leq 2^{r-1} \leq n$. Thus $tal(M) \in LIN^X$. □

We note that the assumption that $tal(M) \in LIN^X$ actually implies that we can test $\sigma \in Bin(M)$ in time $\leq c2^{|\sigma|}$ for some fixed c and almost all σ.

For any structure $\mathcal{M}$ with universe $M \subseteq \omega$, let $tal(\mathcal{M})$ be the tally representation of $\mathcal{M}$ with universe $tal(\omega)$ and relations and functions defined so that the mapping taking n to $tal(n)$ is an isomorphism from $\mathcal{M}$ onto $tal(\mathcal{M})$; $bin(\mathcal{M})$ and $b_k(\mathcal{M})$ are similarly defined. Lemma 4.3 is easily extended to tally and binary representations of relational structures $\mathcal{M}$ and one direction extends to structures with functions.

Lemma 4.4 *Let $k \geq 2$, let $\mathcal{M}$ be a structure with universe $M \subseteq \omega$ and let $\mathcal{A} = tal(\mathcal{M})$ and $\mathcal{B} = b_k(\mathcal{M})$. Then*

(a) *If $\mathcal{A}$ is p-time, then $\mathcal{B}$ is exponential time.*

(b) *If $\mathcal{B}$ is exponential time, then $\mathcal{A}$ is EXPTIME.*

(c) *If $\mathcal{B}$ is exponential time and, for all functions $f^{\mathcal{M}}$, $f^{\mathcal{M}}(n_1, \ldots, n_k) \leq 2^{c(n_1+\cdots+n_k)}$ for some fixed constant c and all but finitely many k-tuples, then $tal(\mathcal{M})$ is exponential time.*

(d) *If $\mathcal{B}$ is exponential time and, for all functions $f^{\mathcal{M}}$, $f^{\mathcal{M}}(n_1, \ldots, n_k) \leq (n_1 + \ldots + n_k)^c$ for some fixed constant c and all but finitely many k-tuples, then $\mathcal{A}$ is p-time.*

(e) *If $\mathcal{B}$ is polynomial time and, for all functions $f^{\mathcal{M}}$, $f^{\mathcal{M}}(n_1, \ldots, n_k) \leq c(n_1 + \ldots + n_k)$ for some fixed constant c and all but finitely many k-tuples, then $\mathcal{A}$ is linear time.*

Proof: We sketch the proofs for $k = 2$.

(a) Suppose that $\mathcal{A}$ is p-time. It follows from Lemma 4.3 that $\mathcal{B}$ has a exponential time universe and it is easy to see that the relations of $\mathcal{B}$ are also exponential time. For simplicity, suppose that $f^{\mathcal{M}}$ is a unary function. (The general proof can be found on p. 320 of [11].) Suppose $tal(m) = f^{\mathcal{A}}(tal(n))$. By assumption, $tal(m)$ may be computed from $tal(n)$ in time $\leq n^c$ for some fixed c and all $n \geq 2$, so that $m \leq n^c$. To compute $f^B(bin(n))$, we first compute $tal(n)$ from $bin(n)$, which takes exponential time by Lemma 4.2. Then we compute $tal(m) = f^{\mathcal{A}}(tal(n))$ in $\mathcal{A}$, which takes time $\leq n^c \leq 2^{c|bin(n)|}$. Finally, we must compute $bin(m)$ from $tal(m)$. This final computation takes time $\leq 8m \leq 8n^c \leq 8(2^{|bin(n)|})^c$. Thus $\mathcal{B}$ is exponential time.

For parts (b), (c) and (d), suppose that $\mathcal{B}$ is exponential time. Then we easily see that the universe of $\mathcal{A}$ and the relations of $\mathcal{A}$ are polynomial time. For simplicity, let $f^{\mathcal{M}}$ be a unary function. The procedure for computing $f^A(tal(n))$ has three parts as above. First, compute $bin(n)$ from $tal(n)$, which takes time $\leq 8n$ by Lemma 4.2. Next, compute $bin(m) = f^{\mathcal{B}}(bin(n))$ which takes time $\leq 2^{c|bin(n)|} \leq (2n)^c$. The final and most time-consuming part of the computation is to compute $tal(m)$ from $bin(m)$, which takes time $\leq 8m$. In general, we only know that $|bin(m)| \leq 2^{c|bin(n)|} \leq (2n)^c$ so that $m \leq 2^{(2n)^c}$ and hence we can only conclude that $\mathcal{A}$ is EXPTIME. For part (c), we have $m \leq 2^{cn}$ and hence $\mathcal{A}$ is exponential time. For part (d), we have $m \leq n^c$ so that $\mathcal{A}$ is p-time.

(e) It follows easily from the hypothesis and Lemma 4.2 that the universe of $\mathcal{A}$ and the relations of $\mathcal{A}$ are linear time. Now let $tal(m) = f^{\mathcal{A}}(tal(n)$ and observe that by the hypothesis, $m \leq nc$. The computation of $bin(n)$ from $tal(n)$ takes time $\leq 8n$ by Lemma 4.2. Since $\mathcal{B}$ is polynomial time, the computation of $bin(m) = f^{\mathcal{B}}(bin(n))$ takes time $\leq |bin(n)|^d$ for some fixed d. It follows as in the proof of Lemma 4.2(c) that this computation can almost always be done in time $\leq n$. Finally, the computation of $tal(m)$ from $bin(m)$ takes time $\leq 8m \leq 8nc$. Thus $\mathcal{A}$ is a linear time structure. □

Note that the hypothesis needed for part (d) follows from the assumption that, for some fixed constant c and all but finitely many k-tuples,

$$|f^{\mathcal{B}}(\sigma_1, \ldots, \sigma_k)| \leq c(|\sigma_1| + \ldots + |\sigma_k|).$$

Nerode and Remmel define in [56] the notion of a *p-time equivalence type (PET)* by saying that two subsets A and B of $Tal(\omega)$ are p-time equivalent if there is a partial 1:1 honest p-time function f with domain including A such that $f(A) = B$. It is natural to extend this notion to p-time subsets of $Bin(\omega)$ by defining two sets A and B to be *p-time isomorphic* if there is a 1:1 p-time function mapping A onto B whose inverse is also p-time. (We may assume that f and f^{-1} have domain $Bin(\omega)$ since A and B assumed to be p-time.)

It follows from Theorem 3 of [56] that for any p-time subset A of $Tal(\omega)$, there are infinitely many p-time subsets of $Tal(\omega)$ which are recursively isomorphic to A but not p-time isomorphic to A. We can now characterize those subsets of $\{1\}^*$ which are polynomial time isomorphic to $Tal(\omega)$ and put conditions on those subsets of $\{0,1\}^*$ which are polynomial time isomorphic to $Tal(\omega)$. The following was proved in [11].

Lemma 4.5 **(a)** *Let A be a p-time subset of $Bin(\omega)$ which is polynomial time isomorphic to $Tal(\omega)$ and let $a_0, a_1, \ldots$ list the elements of A in the standard ordering, first by length and then lexicographically. Then for some j,k and all $n \geq 3$, $n \leq |a_n|^j$ and $|a_n| \leq n^k$.*

(b) *Let A be a p-time subset of $Tal(\omega)$ and let $a_0, a_1, \ldots$ list the elements of A in the standard ordering, Then the following are equivalent.*

1. *A is p-time isomorphic to $Tal(\omega)$.*
2. *For some k and all $n \geq 2$, $|a_n| \leq n^k$.*
3. *The canonical map taking 1^n to a_n is p-time.*

Lemma 4.6 *For any infinite set M of natural numbers, $tal(M) = \{tal(n) : n \in M\}$ and $bin(M) = \{bin(n) : n \in M\}$ are not p-time isomorphic.*

Lemma 4.7 *Let $B_k(\omega)$ be the set of k-ary representations of natural numbers. Then*

(a) *The addition, subtraction, multiplication and division (with remainder) functions from $B_k(\omega) \otimes B_k(\omega)$ to $B_k(\omega)$, the order relation on $B_k(\omega)$ and the length function from $B_k(\omega)$ to $B_k(\omega)$ are all p-time. (As usual, $m - n$ is set to 0 if $m < n$.)*

(b) *$Bin(\omega) \setminus \{1\}^*$ is p-time isomorphic to $Bin(\omega)$.*

(c) *For each $k \geq 2$ and for A equal to either the set $B_k(\omega)$ or the set $\{0, 1, \ldots, k-1\}^*$, there is a polynomial time isomorphism ϕ from A to $Bin(\omega)$ and a constant c such that, for all but finitely many $a \in A$, $|a| \leq |\phi(a)| \leq c|a|$.*

We will frequently want to combine structures using disjoint unions and direct sums. Nerode and Remmel proved in [56] that there exist p-time subsets A, B and C of $Tal(\omega)$ such that A is not p-time isomorphic to B but $A \oplus C$ is p-time isomorphic to $B \oplus C$ and similarly there exist subsets X, Y and Z of $Tal(\omega)$ such that X is not p-time isomorphic to Y but $X \otimes Z$ is p-time isomorphic to $Y \otimes Z$. This result of Nerode and Remmel also easily follows from our next results due to Cenzer and Remmel [11].

Lemma 4.8 **(a)** *Let A be a p-time subset of $Tal(\omega)$. Then $A \oplus Tal(\omega)$ is p-time isomorphic to $Tal(\omega)$ and $A \oplus Bin(\omega)$ is p-time isomorphic to $Bin(\omega)$.*

(b) *Let A be a nonempty p-time subset of $Tal(\omega)$. Then $A \otimes Tal(\omega)$ is p-time isomorphic to $Tal(\omega)$ and $A \otimes Bin(\omega)$ is p-time isomorphic to $Bin(\omega)$.*

Proof: (a) First observe that $A \oplus Tal(\omega)$ is p-time isomorphic to the set $C = \{2a : a \in A\} \cup \{2n + 1 : n \in Tal(\omega)\}$ by the obvious isomorphism. Now let $c_0, c_1, \ldots$ enumerate C in increasing order. Since C contains every odd number, it is clear that $c_n \leq 2n+1$. It follows from Lemma 4.5 that C is p-time isomorphic to $Tal(\omega)$.

Next, $A \oplus Bin(\omega)$ is certainly p-time isomorphic to $(A \oplus Tal(\omega)) \oplus (Bin(\omega) \setminus Tal(\omega))$. Now by the preceding discussion, $A \oplus Tal(\omega)$ is p-time isomorphic to $Tal(\omega)$. It follows that $A \oplus Bin(\omega)$ is p-time isomorphic to $Tal(\omega) \oplus (Bin(\omega) \setminus Tal(\omega))$ which is clearly p-time isomorphic to $Bin(\omega)$.

(b) If A has only one element, this is obvious. If A has at least two elements, let a be one of them. Then $A \times Tal(\omega)$ is p-time isomorphic to $(\{a\} \times Tal(\omega)) \oplus ((A \setminus \{a\}) \times Tal(\omega))$. Now the first part of this sum is obviously p-time isomorphic to $Tal(\omega)$ and the second part is p-time isomorphic to some p-time subset of $Tal(\omega)$. It now follows from part (a) that the sum is p-time isomorphic to $Tal(\omega)$.

For the case of $Bin(\omega)$, again if A consists of single element, the result is trivial. Otherwise, let a be the least element of A and consider the following mapping. If $b \in A$ and $b \neq a$, then let $f(\langle b, \sigma \rangle) = \sigma^\frown 10^b 1$. The idea is to define

f so that the image of $\{a\} \times Bin(\omega)$ under f is $Bin(\omega) \setminus f((A \setminus \{a\}) \times Bin(\omega))$ by letting $f(\langle a, bin(n)\rangle)$ be the n-th element of $Bin(\omega) \setminus f((A \setminus \{a\}) \times Bin(\omega))$. Observe that for any $\tau = bin(q) \in Bin(\omega)$, which is not of the form $\sigma^\frown 10^b1$ for some $b \in A \setminus \{a\}$, we can find in polynomial time in $|\tau|$ all strings of the form $1^{k_b}{}^\frown 10^b1$ such that

1. $b > 0$,
2. $1^b \in A$ and
3. under the usual ordering on $Bin(\omega)$, $1^{k_b}{}^\frown 10^b1 < \tau$ but $1^{k_b+1}{}^\frown 10^b1 \not< \tau$.

That is, since A is a polynomial time subset of $Tal(\omega)$, we can test each element of the 1^b with $0 < b \le |\tau|$ for membership in A in b^k steps for some fixed k and hence find all strings of the form 1^b where $q \ge b > 0$ and $1^b \in A$ in at most $\sum_{j=1}^{|\tau|} j^k \le (|\tau|)^{k+1}$ steps. Thus in polynomial time in $|\tau|$, we can find all the strings $1^{k_{b_1}}{}^\frown 10^{b_1}1, \ldots, 1^{k_{b_p}}{}^\frown 10^{b_p}1$ satisfying properties (1)-(3) above. This mean that exactly $\sum_{i=1}^{p} 2^{k_i}$ elements of $Bin(\omega)$ which are less than or equal to τ are in the image of $f((A\setminus\{a\}) \times Bin(\omega))$. Thus we let $f(\langle a, bin(q - \sum_{i=1}^{p} 2^{k_i}\rangle) = \tau$. It follows that f^{-1} is polynomial time. To see that f is polynomial time, note that $f(\langle a, bin(n)) \le bin(n)^\frown 11$ so a similar computation starting with $\tau = bin(n)^\frown 11$ will allow us to find the n-th element of $Bin(\omega)\setminus f((A\setminus\{a\})\times Bin(\omega))$ in polynomial time in $|bin(n)|$. □

It follows easily from the lemmas above that any p-time relational structure $\mathcal{A}$ is recursively isomorphic to a p-time structure $\mathcal{B}$ such that $\mathcal{A}$ and $\mathcal{B}$ are not p-time isomorphic. (We assume that $\mathcal{A}$ has universe $Bin(A) = \{bin(a) : a \in A\}$ for some $A \subseteq \omega$ and let $\mathcal{B}$ have universe $Tal(A) = \{tal(a) : a \in A\}$.) Now these structures are actually exponential-time isomorphic. However, there is a stronger result.

Lemma 4.9 *For any p-time set $A = \{bin(a_0) < bin(a_1) < \cdots\}$, there is a set $M = M(A) = \{bin(m_0) < bin(m_1) < \cdots\}$ such that M is p-time and the map which takes $bin(m_i)$ to $bin(a_i)$ is p-time, but there is no primitive recursive map from A into M which maps at most k elements of A to any element of M' where k is any fixed finite number. Furthermore, M may be taken to be a subset of $Tal(\omega)$.*

Proof: Let ϕ_e be the e-th primitive recursive function mapping $Bin(\omega)$ into $Bin(\omega)$ and, for each e, let t_e be the total time required to test all numbers up to a_e for membership in A and to compute $\phi_i(bin(a_j))$ for all $j, i \le a_e$; clearly $t_e < t_{e+1}$. For each e, let $m_e = 2^{t_e}$, so that $bin(m_e) = 0^{t_e}1$ and $|bin(m_e)| = t_e + 1$. It follows that $\phi_e(bin(a_i)) < bin(m_i)$ for all $i \le e$, since by convention it takes at least k steps to compute an output of length k. Let $A^* = \{bin(m_e) : e < \omega\}$.

Here is the p-time algorithm for testing whether $x \in A^*$. First check to see that $x = 0^t1$ for some n. Then start to test $bin(0), bin(1), \ldots$ for membership in

A. As soon as we find that $bin(n)$ is e-th member of A so that $bin(n) = bin(a_e)$, then compute in order $\phi_e(a_0), \cdots, \phi_e(a_{e-1})$ and $\phi_0(a_e), \cdots, \phi_e(a_e)$ and then return to testing whether $bin(n+1), bin(n+2), \cdots$ are in A. If the total number of steps reaches t exactly when the computation of some $\phi_e(a_e)$ has just been completed, then $t = t_e$ so that $x = bin(m_e)$ belongs to A^*. Otherwise, $x \notin A^*$. This argument also shows that the map which takes $bin(m_e)$ to $bin(a_e)$ is p-time. Finally, let $M = M(A) = \{bin(m_{i^2}) : i < \omega\}$. Then M is also p-time since we can clearly check in polynomial time whether i is a square if we discover that $x = bin(m_e)$ at the end of the computation. The map taking $bin(m_{i^2})$ to $bin(a_i)$ is p-time since we can test all numbers $\leq a_{i^2}$ for membership in A in time $\leq |bin(m_{i^2})|$ and therefore determine a_i from $bin(m_{i^2})$.

Suppose now by way of contradiction that ϕ_e were a map from A into M' which were at most k to 1. Since each primitive recursive function has infinitely many indices, we may assume that $e \geq k$. Then

$$\{\phi_e(bin(a_0)), \ldots, \phi_e(bin(a_{e^2}))\}$$

must contain at least e distinct elements, so that at least one of them is $\geq bin(m_{e^2})$, which contradicts the observation above that $\phi_e(bin(a_i)) < bin(m_e)$ for all $i \leq e$ and thus establishes the result.

Since $Bin(\omega)$ and $Tal(\omega)$ are primitive recursively isomorphic via the standard map μ_2, it follows that we could replace M in this argument with the set $M^* = \{tal(m_0), tal(m_1), \ldots\}$. □

Remark 1: The only properties of the set of primitive recursive functions needed for this result is that the primitive recursive functions is a class of total recursive functions which can be effectively listed. Thus in the statement of Theorem 4.9 we can replace the primitive recursive functions by any class of total recursive functions which can be effectively listed.

Remark 2: Letting $k = 1$, we see that there is no primitive recursive embedding of A into $M(A)$. Furthermore, there is no primitive recursive embedding ϕ of a cofinite subset C of A into $M(A)$, since if $|A \setminus C| = k - 1$, then we could extend ϕ to a map from A into $M(A)$ which is at most k to 1 by mapping all elements of $A \setminus C$ to some fixed element of $M(A)$.

5 The Existence of Feasible Structures

In this section, we shall survey a number of model existence results for polynomial time structures. In particular, we will consider four existence questions for any class **C** of structures.

- Is every recursive structure in **C** isomorphic to a polynomial time structure?
- Is every recursive structure in **C** recursively isomorphic to a polynomial time structure?

- Is every recursive structure in **C** isomorphic to a polynomial time structure with a specified universe such as the binary or tally representation of the natural numbers?
- Is every recursive structure in **C** recursively isomorphic to a polynomial time structure with a specified universe such as the binary or tally representation of the natural numbers?

The fundamental result for relational structures is due to Grigorieff [32] and (for structures with infinitely many relations) to Cenzer-Remmel [10]. Recall that a structure with no function is said to be relational.

Theorem 5.1 *Every relational structure is recursively isomorphic to a real time structure with universe a subset of $Bin(\omega)$ and to a linear time structure with universe a subset of $Tal(\omega)$.*

Sketch of Proof: We may assume without loss of generality that the structure $\mathcal{A}$ has universe ω. The element a is represented in the binary real time model by $\psi(a) = 1^{a+1}010^t$, where t is the time required to compute whether $R_j(x_1, \ldots, x_{t(j)})$ for all $j \leq a$ and all tuples $(x_1, \ldots, x_{t(j)})$ from $\{0, \ldots, a\}^{t(j)}$.

The set $B = \{\psi(a) : a \in \omega\}$ is real time by the following algorithm.

> Given $bin(n)$, start to read $bin(n)$ from left to right. If at any time we discover that $bin(n)$ is not of the form $1^{a+1}010^t$ for some a and t, then $bin(n) \notin B$. Otherwise, having read the first 0 in $bin(n)$ so that we have found a such that $bin(n) = 1^{a+1}010^t$, start the computation which tests for all $j \leq a$ and for all tuples $(x_1, \ldots, x_{t(j)})$ from $\{0, \ldots, a\}^{t(j)}$, whether $R_j(x_1, \ldots, x_{t(j)})$. If the total computation finishes in exactly t steps, then $bin(n) \in B$. If the computation either finishes in fewer steps, or has not finished by t steps, then $bin(n) \notin B$.

Each relation R_j is real time, by the following algorithm.

> Given $(bin(n_0), \ldots, bin(n_{t(j)}) \in B^{t(j)}$, first compute a_k and t_k for each $k \leq j$ so that $bin(n_k) = 1^{a_k+1}010^{t_k}$. Then let $t = max\{t_0, \ldots, t_{t(j)}\}$. Now by the construction we can test whether $R_j(a_0, \ldots, a_{t(j)})$ in time $\leq t$.

The tally representation of $\mathcal{A}$ has universe $\{tal(n) : bin(n) \in B\}$ and is linear time by Lemma 4.4(e). □

Note that Theorem 5.1 allows us to conclude that if G is a recursive graph, i.e. $G = (V, E)$ where V, the vertex set of G, is a recursive set of natural numbers and the edge relation E is also recursive, then G is recursively isomorphic to a polynomial time graph G'. However if G has a recursive k-coloring, then to conclude that G is recursively isomorphic to a polynomial time graph with a polynomial time k-coloring requires a stronger version of Theorem 5.1. We will present an improved version of Theorem 5.1 due to Cenzer and Remmel [17] which is their primary tool in their analysis of polynomial time combinatorial structures and Π_1^0-classes. The improved version of the theorem presented applies to structures with two distinct types of objects, the first type being the normal universe of the structure, and with functions which map the first type into the second type. The type of example that we have in mind is a function from the vertices of a graph into the natural numbers which computes the degree of a vertex or the color assigned to a vertex. The universe of the graph is now expanded by adding a p-time set which represents the natural numbers and the degree function or coloring now becomes part of the structure. Naturally, the new objects are not vertices and therefore are not joined to any other objects by edges.

Theorem 5.2 *Let*

$$\mathcal{C} = (C, A, B, \{R_i^{\mathcal{C}}\}_{i \in S}, \{f_i^{\mathcal{C}}\}_{i \in T},),$$

be a recursive structure such that

(i) *A and B are disjoint subsets of C with $C = A \cup B$ and B is a polynomial time set;*

(ii) *there is a recursive isomorphism from $Bin(\omega)$ onto a subset of $Bin(\omega) \setminus B$ with a p-time inverse;*

(iii) *for each $i \in T$, f_i maps C into B;*

(iv) *for each $i \in S$, the relation R_i is independent of B, that is, for any $(x_1, \ldots, x_n) \in C^n$, where $n = s(i)$, any $j \leq n$ such that $x_i \in B$, and any $b \in B$, $R_i^{\mathcal{C}}(x_1, \ldots, x_n)$ holds if and only if $R_i^{\mathcal{C}}(x_1, \ldots, x_{j-1}, b, x_{j+1}, \ldots, x_n)$ holds;*

(v) *for each $i \in T$, the function f_i is independent of B, that is, for any $(x_1, \ldots, x_n) \in C^n$, where $n = t(i)$, any $j \leq n$ such that $x_i \in B$, and any $b \in B$, $f_i^{\mathcal{C}}(x_1, \ldots, x_n) = f_i^{\mathcal{C}}(x_1, \ldots, x_{j-1}, b, x_{j+1}, \ldots, x_n)$.*

Then there is a recursive isomorphism ϕ of $\mathcal{C}$ onto a p-time structure $\mathcal{M}$ such that $\phi(b) = b$ for all $b \in B$.

5.1 Structures with Functions

In contrast to purely relational structures, recursive structures with functions need not be effectively isomorphic to feasible structures. The following results are Theorems 3.1, 3.2 and 3.3 of [10].

Theorem 5.3 *Let $\mathcal{L}_0$ be the language with exactly one function symbol f which is unary.*

(a) *There is a recursive structure $\mathcal{A} = (A, f^{\mathcal{A}})$ which is not recursively isomorphic to any primitive recursive structure.*

(b) *There is an exponential time structure $\mathcal{D} = (D, f^{\mathcal{D}})$ which is not recursively isomorphic to any polynomial time structure.*

Proof: (a) Let $(A_0, f_0), (A_1, f_1), \ldots$ be an effective list of all primitive recursive structures over $\mathcal{L}_0$ and let $\phi_0, \phi_1, \ldots$ be a list of all one-to-one partial recursive functions. We must meet the following set of requirements in our construction of $\mathcal{A}$.

$$R_{i,j}: \quad \phi_j \text{ is not a recursive isomorphism from } \mathcal{A} \text{ to } (A_i, f_i).$$

To meet the requirements $R_{i,j}$, recursively partition $\{0,1\}^*$ into infinitely many disjoint infinite recursive sets $S_{i,j}$. We then define $\mathcal{A}$ so that $A = \bigcup_{i,j} S_{i,j} = \{0,1\}^*$ and for all i, j, $f^{\mathcal{A}}$ maps $S_{i,j}$ into $S_{i,j}$.

We now fix i, j and then we define $f = f^{\mathcal{A}}$ on $S_{i,j}$ in stages. We let $a_0, a_1, \ldots$ be some effective listing of $S_{i,j}$. At stage s, we shall define $f(a_s)$. We start by defining $f(a_0) = a_1$ at stage 0. At stage $s+1$, compute $\phi_j^s(a_0)$. If $\phi_j^s(a_0) \uparrow$ or if $\phi_j^{s-1}(a_0) \downarrow$, then we define $f(a_s) = a_{s+1}$. Otherwise, that is, if $\phi_j^s(a_0) \downarrow$ but $\phi_j^{s-1}(a_0) \uparrow$, let $x = \phi_j^s(a_0)$ and do the following. Compute the sequence $x, f_i(x), f_i(f_i(x)), \ldots, f_i^{(s+1)}(x)$, where here $f_i^{(k)}$ denotes f composed with itself k times. Note that if ϕ_j were an isomorphism, then it must be the case that $\phi_j(a_k) = f_i^{(k)}(x)$ for all k. Thus if ϕ_j were an isomorphism, then it must be the case that

$$f(a_s) = a_0 \iff f_i^{(s+1)}(x) = x.$$

Thus if $f_i^{(s+1)}(x) = x$, then we define $f(a_s) = a_{s+1}$. If $f_i^{(s+1)}(x) \neq x$, then we define $f(a_s) = a_0$. Note that in either case, we will have ensured that ϕ_j cannot be an isomorphism from $\mathcal{A}$ onto (A_i, f_i).

(b) The proof of part (a) must be modified in several ways. First, let $(E_0, f_0), (E_1, f_1), \ldots$ be an effective list of all p-time structures whose universe is contained in $\{0,1\}^8$ over $\mathcal{L}_0$. Let $\phi_0, \phi_1, \ldots$ be a list of all one-to-one partial recursive functions which map $\{0,1\}^*$ into $\{0,1\}^*$. We shall build our structure (D, f^D) so that $D \subseteq Tal(\omega)$ and we meet the following set of requirements.

$$R_{i,j}: \quad \phi_j \text{ is not a recursive isomorphism from } \mathcal{D} \text{ to } (E_i, f_i).$$

To meet the requirements $R_{i,j}$, we construct D as a disjoint polynomial-time union of infinite p-time sets $D = \bigcup_{i,j} T_{i,j}$. Define the function $\psi : Tal(\omega) \times Tal(\omega) \times Tal(\omega) \to Tal(\omega)$ by $\psi(0, tal(i), tal(j)) = tal(2[i,j]) + 3)$ and $\psi(tal(n+1), tal(i), tal(j)) = tal(2^p)$ if $\psi(tal(n), tal(i), tal(j)) = tal(p)$.

Note that $tal(x) = tal(2[i,j] + 3)$ can be computed from input $(tal(i), tal(j))$ in time $a \cdot x$ for some fixed constant a and that the computation of $tal(y) = tal(2^x)$ from input $tal(x)$ can be computed in time at most $b \cdot y$ for some fixed constant $b \geq a$. Thus the computation of $tal(z) = \psi(tal(n), tal(i), tal(j))$ from input $(tal(n), tal(i), tal(j))$ takes at most the following number of steps:

$$b[i,j] + bx + b2^x + b2^{2^x} + \cdots + bz < b(1 + 2 + \cdots + z) < bz^2.$$

For each i, j, we let $T_{i,j} = \{\psi(n, i, j) : n < \omega\}$. Then we can test whether $tal(z) = \psi(tal(n), tal(i), tal(j))$, perform the computation of

$$\psi(0, tal(i), tal(j)), \psi(1, tal(i), tal(j)), \ldots, \psi(tal(n), tal(i), tal(j))$$

for bz^2 steps and see if the computation converges to z by that time. It follows that the sets $T_{i,j}$ are uniformly p-time and that D is also p-time, since

$$tal(z) \in T_{i,j} \iff (\exists n < z)\ tal(z) = \psi(tal(n), tal(i), tal(j))$$

and

$$tal(z) \in D \iff (\exists i, j < z)\ tal(z) \in T_{i,j}.$$

We now fix i, j and define $f = f^{\mathcal{D}}$ on $T_{i,j} = \{tal(a_0), tal(a_1), \ldots\}$, where $tal(a_n) = \psi(tal(n), tal(i), tal(j))$. For each m, perform the following series of computations for at most 2^{a_m} steps.

(1) Start to compute $\phi_j(tal(a_0))$. If this converges in less than 2^{a_m} steps, let $b_0 = \phi_j(tal(a_0))$.

(2) Check that $b_0 \in E_i$.

(3) Compute the sequence $b_1 = f_i(b_0), b_2 = f_i(b_1), \ldots, b_{m+1} = f_i(b_m)$.

Let s be the least m such that the computations can be successfully completed in at most 2^{a_m} steps. Assuming the existence of $b_0 = \phi_j(a_0)$, we can show that such an s must exist. That is, it takes some constant amount c_0 of time to compute b_0. Since f_i is p-time, $f_i(y)$ can be computed in time bounded by $|y|^k$ for some fixed integer $k > 1$ and any $y \in \{0,1\}^*$ with $|y| > 1$. Let c_1 be the time required to compute $f_i(\emptyset)$, $f_i(0)$ and $f_i(1)$, if needed, and let $c = c_0 + c_1$. Then to compute the sequence $b_0 = \phi_j(a_0), b_1 = f_i(b_0), \ldots, b_{m+1} = f_i(b_m)$ takes time at most

$$t(m) = c + |b_0|^k + (|b_0|^k)^k + \cdots \quad = c + |b_0|^k + |b_0|^{k^2} + \cdots + |b_0|^{k^m}.$$

We need to show that this sequence is eventually dominated by the sequence $a_0, a_1 = 2^{a_0}, \ldots, a_m = 2^{a_{m-1}}$. We may assume without loss of generality that $|b_0| > 1$. Now if m is large enough so that both c and m are $< |b_0|^{k^m}$, then

$$t(m) < (m+1)|b_0|^{k^m} \leq |b_0|^{2k^m}.$$

Now let m be large enough so that $|b_0|^2 < 2^{2^m}$, $k < 2^m$, and $m^2 + m < 2^m$. Then $k^m < 2^{m^2}$ and $t(m) < 2^{2^m \cdot 2^{m^2}} = 2^{2^{m^2+m}} < 2^{2^{2^m}} = exp_3(m)$. To show that the latter is dominated by a_m, note first that $a_0 \geq 3$ and that, for any m, $a_m \geq m+3$; it follows that $a_{m+3} \geq exp_3(m+3)$.

The definition of f now proceeds in stages, as in part (a). Let s be the least m such that the computations described above can be successfully completed. Now for $t \neq s$, we let $f(a_t) = a_{t+1}$. To compute $f(a_s)$, we let $x = \phi_j(a_0)$ and compute $f_i^{(s+1)}(x)$. Then if $f_i^{(s+1)}(x) = x$, we define $f(a_s) = a_{s+1}$. If $f_i^{(s+1)}(x) \neq x$, then we define $f(a_s) = a_0$. Note that in either case, we will have ensured that ϕ_j cannot be an isomorphism from $\mathcal{D}$ onto (E_i, f_i).

It remains to be seen that the computation of f can be done in exponential time. Given $tal(x) \in D$, we first compute the unique triple (n,i,j) such that $tal(x) = \psi(tal(n), tal(i), tal(j))$. This can be done in polynomial time since n, i and j are all less than x. Next we perform the computations (1), (2) and (3) for $m = 0, 1, \ldots, n$ in turn. This can be done in exponential time since each series of computations is bounded by time 2^{a_m}. The remainder of the computation of $f(x)$ takes little time. We look for the least n, if any, such that the n-th series of computations has been successfully completed. If $m = n$, then we check to see if $f_i^{(s+1)}(x) = x$ and let $f(tal(a_m)) = tal(2^{a_m})$ if so; otherwise, $f(tal(a_m)) = tal(a_0)$. □

We note here that the functions in the previous theorem may be taken to be permutations of the sets A and D. Then next two results show that we can diagonalize over Δ_2^0 isomorphisms if the underlying language has at least two unary function symbol or at least one n-ary function symbol with $n \geq 2$.

Theorem 5.4 *Let $\mathcal{L}_0$ be the language with exactly two function symbols f and g which are unary.*

(a) *There is a recursive structure $\mathcal{A} = (A, f^{\mathcal{A}}, g^{\mathcal{A}})$ which is not Δ_2^0 isomorphic to any primitive recursive structure.*

(b) *There is an exponential time structure $\mathcal{D} = (D, f^{\mathcal{D}}, g^{\mathcal{D}})$ which is not Δ_2^0 isomorphic to any polynomial time structure.*

Theorem 5.5 *Let $\mathcal{L}_0$ be the language with exactly one function symbol h which is binary.*

(a) *There is a recursive structure $\mathcal{A} = (A, h^{\mathcal{A}})$ which is not Δ_2^0 isomorphic to any primitive recursive structure.*

(b) *There is an exponential time structure $\mathcal{D} = (D, h^{\mathcal{D}})$ which is not Δ_2^0 isomorphic to any polynomial time structure.*

Two natural types of structures with functions will be considered below in more detail. These are permutation structures (A, f^A), where f^A is a permutation of the set A, and Abelian groups.

Next we state an unpublished theorem due to H. Freidman and J. Remmel which characterizes when structures which are finitely generated are isomorphic to polynomial time structures.

Definition 5.6 (a) *We say that*
$\mathcal{A} = (A, \{R_i^A\}_{i=1,\ldots,k}, \{f_i^A\}_{i=1,\ldots,n}, \{c_i^A\}_{i=1,\ldots,m})$ *is finitely generated if* A *equals the closure of* $\{c_i^A\}_{i=1,\ldots m}$ *under the set of functions* $\{f_i^A\}_{i=1,\ldots,n}$.

(b) *A finitely generated structure* $\mathcal{A}$ *as above has a double exponential time decision procedure if*

1. $A \subseteq \{0,1\}^*$ *is double exponential time.*
2. *There is an algorithm which given any two terms* $t_1(c_1^A, \ldots, c_m^A)$ *and* $t_2(c_1^A, \ldots, c_m^A)$ *in the free term algebra generated by* $c_1^A, \ldots, c_m^A$ *and* $f_1^A, \ldots, f_n^A$, *decides in* $2^{2^{dk_1}}$ *steps for some constant* k_1 *if* $t_1(c_1^A, \ldots, c_m^A) = t_2(c_1^A, \ldots, c_m^A)$ *where* d *is equal to the maximum of the depth of* $t_1(c_1^A, \ldots, c_m^A)$ *and the depth of* $t_2(c_1^A, \ldots, c_m^A)$.
3. *There is an algorithm which given a relation* R_i^A *and terms* $t_1(c_1^A, \ldots, c_m^A), \ldots, t_p(c_1^A, \ldots, c_m^A)$ *in the free term algebra generated by* $c_1^A, \ldots, c_m^A$ *and* $f_1^A, \ldots, f_n^A$, *decides in* $2^{2^{dk_2}}$ *steps for some constant* k_2 *if* $R_i^A(t_1(c_1^A, \ldots, c_m^A), \ldots, t_p(c_1^A, \ldots, c_m^A))$ *holds where* d *is equal to the maximum of the depths of* $t_j(c_1^A, \ldots, c_m^A)$ *for* $j = 1, \ldots, p$.

Theorem 5.7 *Let* $\mathcal{A} = (A, \{R_i^A\}_{i=1,\ldots,k}, \{f_i^A\}_{i=1,\ldots,n}, \{c_i^A\}_{i=1,\ldots,m})$ *be a finitely generated structure. Then* $\mathcal{A}$ *is isomorphic to a polynomial time model iff* $\mathcal{A}$ *has a double exponential time decision procedure.*

5.2 Linear Orderings

There are several results on the existence of p-time linear orderings.

Theorem 5.8 *(Grigorieff 1989) Every recursive linear ordering* $\mathcal{L}$ *is isomorphic to a real time linear ordering* $\mathcal{L}' = (Bin(\omega), <_{\mathcal{L}'})$.

Sketch of Proof: We sketch a proof showing that $\mathcal{L}$ is isomorphic to a p-time ordering. There are two cases. The first case is where $\mathcal{L}$ has either a recursive increasing sequence

$$S = (s_0 <_{\mathcal{L}} s_1 <_{\mathcal{L}} s_2 <_{\mathcal{L}} \ldots)$$

such that S is cofinal or S has a limit or $\mathcal{L}$ has a recursive decreasing sequence

$$D = (d_0 >_{\mathcal{L}} d_1 >_{\mathcal{L}} d_2 >_{\mathcal{L}} \ldots)$$

such that D is cofinal or D has a limit.

In this case, we have a p-time copy of either S or D with universe $Bin(\omega)$ and we can make a p-time copy of $\mathcal{L} \setminus \mathcal{S}$ (or $\mathcal{L} \setminus D$) with universe a subset of $Tal(\omega)$ by Theorem 5.1. Then we can apply Lemma 4.8 to combine the two orderings into one p-time ordering with universe $Bin(\omega)$ which is recursively isomorphic to $\mathcal{L}$.

The second case is where no such sequences exist. In this case $\mathcal{L}$ is isomorphic to $\omega + \mathbb{Z} \cdot \lambda + \omega^*$ for some recursive ordinal λ. Here $\mathbb{Z}$ is the order type of the integers. Then there are two subcases. First if λ has a first or last element or has a pair of elements $x <_L y$ such y is an immediate successor of x, then $\mathcal{L}$ contains an explicit recursive copy of $D = \omega + \omega^*$ which is isomorphic to a p-time linear ordering with universe $Bin(\omega)$. We can make a p-time copy of $\mathcal{L} \setminus D$ with universe a subset of $Tal(\omega)$ by Theorem 5.1. Then we can apply Lemma 4.8 to combine the two orderings into one p-time ordering with universe $Bin(\omega)$. The only other subcase is that λ is a dense linear ordering without endpoints so that λ is isomorphic to the rationals $\mathbb{Q}$. But in this case it is easy to construct a p-time linear ordering with universe $Bin(\omega)$ which is isomorphic to $\mathcal{L}$. □

The natural question is whether the isomorphism in the previous theorem is effective. It should be observed that the proof is not uniform. Indeed in case 2, we only constructed an isomorphic copy and not a recursively isomorphic copy of $\mathcal{L}$. The next result due to Remmel is Theorem 2.2 of [68] which shows that in case 2 the isomorphism cannot be replaced by a recursive isomorphism.

Theorem 5.9 *Let $A \subseteq \{0,1\}^*$ be any infinite p-time set and let $\mathcal{L}$ be a recursive linear ordering which is isomorphic to $\omega + \mathbb{Z} \cdot \lambda + \omega^*$ for some linear ordering λ. Then there exists a recursive linear ordering $\mathcal{K}$ which is isomorphic to $\mathcal{L}$ but which is not recursively isomorphic to any p-time linear ordering whose universe is A.*

Sketch of Proof:

We will not give the full proof as it requires an infinite injury priority argument. However we will give the proof in the case where the $\mathcal{L}$ is isomorphic to $\omega + \omega^*$ since in that case, a simple finite injury priority argument suffices.

Recall that ϕ_i is the i-th partial recursive function and let $R_0, R_1, \ldots$ be an effective list of all polynomial time binary relations on $\{0,1\}^*$. For simplicity, we let $\langle A, R_e \rangle$ denote the structure with universe A and relation R which is the restriction of R_e to $A \times A$.

Let $\tau_0, \tau_1, \ldots$ be an effective enumeration of A in the usual order (first by length and then by lexicographic order.)

We shall construct our desired recursive linear ordering $\mathcal{L}$ in stages. Let $\sigma_0, \sigma_1, \ldots$ be an effective listing of $\{0,1\}^*$. At any given stage s, we shall specify two sequences $a_0^s, a_1^s, \ldots, a_{n_s}^s$ and $b_0^s, b_1^s, \ldots, b_{n_s}^s$ for some $n_s \geq s$ such that $B_s = \{\sigma_0, \ldots, \sigma_{2n_s+1}\} = \{a_0^s, b_0^s, a_1^s, b_1^s, \ldots, a_{n_s}^s, b_{n_s}^s\}$. Moreover, at stage s we shall define the ordering $< \, = \, <_{\mathcal{L}}$ on $B_s \times B_s$ so that

$$a_0^s < a_1^s < \cdots < a_{n_s}^s < b_{n_s}^s < b_{n_s-1}^s < \cdots < b_0^s.$$

Our construction will ensure that for all i, $\lim_s a_i^s = a_i$ and $\lim_s b_i^s = b_i$ exist. Moreover, our construction will ensure that $\{0,1\}^* = \{a_0, b_0, a_1, b_1, \ldots\}$ and that

(a) for all i, $a_i < a_{i+1}$ and $b_{i+1} < b_i$ and

(b) for all i and j, $a_i < b_j$.

Thus $(\{0,1\}^*, <_{\mathcal{L}})$ will have order type $\omega + \omega^*$. To ensure that $\mathcal{L}$ is not recursively isomorphic to (A, R_e) for any e, we shall meet the following set of requirements

$P_{[e,k]}$: There exist n and m such that one of the following four conditions holds.

(i) $\phi_k(a_n) \uparrow$ or $\phi_k(a_n) = x \notin A$.

(ii) $\phi_k(b_m) \uparrow$ or $\phi_k(b_m) = x \notin A$.

(iii) $\phi_k(a_n) = x \in A$ and there exist $n+1$ elements $v_0, \ldots, v_n$ of A such that $(v_i, x) \in R_e$ for $i = 0, \ldots, n$.

(iv) $\phi_k(b_m) = y \in A$ and there exist $m+2$ elements $w_0, \ldots, w_{m+1}$ of A such that $(w_i, y) \notin R_e$ for $i = 0, \ldots, m+1$.

We write $(w_i, y) \notin R_e$ rather than $(y, w_i) \in R_e$ in clause (iv) to allow for the possibility that R_e is not actually a linear ordering.

It is easy to see that if requirement $P_{[e,k]}$ is satisfied, then ϕ_e is not a recursive isomorphism from $\mathcal{L} = (\{0,1\}^*, <_{\mathcal{L}})$ onto (A, R_e). Thus meeting all the requirements $P_{[e,k]}$ ensures that $\mathcal{L}$ is not recursively isomorphic to any p-time linear ordering with universe A.

Our basic strategy for meeting a requirement P_z, where $z = [e,k]$, is as follows. Let us assume that $s > z$ is a stage large enough so that requirements $P_0, \ldots, P_{z-1}$ no longer require action at any stage $t \geq s$. Then at stage s, we consider a_z^s. Our construction will then ensure that $a_j^s = a_j^t$ for all $j \leq z$ and $t \geq s$ unless there is a stage $u \geq s$ such that $\phi_k^u(a_z^s) \downarrow$. Of course if there is no such u, then $a_z^s = a_z$ and a_z will witness that requirement P_z is satisfied (by virtue of clause (i)).

Now if there is such a stage u, then let $x = \phi_k^u(a_z^s)$. If $x \notin A$, then again we will simply ensure $a_z^s = a_z$ so that once again a_z will witness that requirement P_z is satisfied. If $x \in A$, then we will compare x to the first $4n_{u-1} + 4$ elements of A (in the fixed order $\tau_0, \tau_1, \ldots$ prescribed above) with respect to the binary relation R_e. Note that since A and R_e are polynomial time, we can effectively make these $4n_{u-1} + 4$ comparisons. There are two possibilities.

(i) There are $h = 2n_{u-1} + 2$ of these elements v of A such that $(v, x) \in R_e$ – denote these elements by $v_0, \ldots, v_{h-1}$.

(ii) There are $h = 2n_{u-1} + 3$ of these elements w of A such that $(w, x) \notin R_e$ – denote these elements by $w_0, \ldots, w_{h-1}$.

In case (i), we will simply ensure $a^s_z = a_z$. But then a_z is preceded by exactly z elements in $\mathcal{L}$, where $z \leq n_{u-1}$, whereas $x = \phi_k(a_z)$ is preceded by at least $2n_{u-1} + 2$ elements in $\langle A, R_e\rangle$. Thus ϕ_k is not an isomorphism from $\mathcal{L}$ onto (A, R_e).

In case (ii), we will switch $a^s_z = a^{u-1}_z$ from the ω side of $\mathcal{L}$ to the ω^* side of $\mathcal{L}$. That is, we shall let $n_u = 2n_{u-1} - z + 1$ and let $b^u_{n_{u-1}+i} = a_{n_{u-1}-i+1}$ for $i = 1, \ldots, n_u - n_{u-1}$. We also let $b^u_i = b^{u-1}_i$ for $i \leq n_{u-1}$. Then our construction will ensure that for all $t \geq u$, $b^t_{n_u} = b^u_{n_u} = a^s_z$. Thus in this case, there will be precisely $n_u + 1$ elements w (namely $b_0, \ldots, b_{n_u}$) such that $(w, b_{n_u}) \notin <_{\mathcal{L}}$. However, in (A, R_e) there are at least $2n_{u-1} + 3$ elements x such that $(x, \phi_k(b_{n_u})) \notin R_e$. But $n_u + 1 = 2n_{u-1} + 2 - z < 2n_{u-1} + 3$, so that ϕ_u cannot be an isomorphism from $\mathcal{L}$ onto (A, R_e). Our construction will ensure that a^s_z can switch from the ω side to the ω^*-side of $\mathcal{L}$ only for the sake of requirements $P_0, \ldots, P_z$. The usual priority argument will then show that a^s_z "switches sides" for at most finitely many s.

We shall employ a set of movable markers Γ_e to help us keep track of which requirements we have acted on. The idea is that if we have taken an action as described above which ensures a^u_z will witness that requirement P_z is satisfied, then we will place a Γ_z marker on a^u_z. Thus at any given stage s, either Γ_z is *inactive*, i.e., Γ_z does not rest on any element at stage s, or Γ_z is *active*, i.e., Γ_z rests on some element $x \in \{a^s_0, b^s_0, \ldots, a^s_{n_s}, b^s_{n_s}\}$. If Γ_z is active, we let $\Gamma_z(s) = x$, where x is the element on which Γ_z is placed.

CONSTRUCTION.

<u>Stage 0</u>.

Let $a^0_0 = \sigma_0$, $b^0_0 = \sigma_1$, and declare $a^0_0 < b^0_0$. We let Γ_z be inactive for all z at stage 0.

<u>Stage $s+1$</u>. Assume we have defined $n = n_s$, $a^s_0, b^s_0, \ldots, a^s_n, b^s_n$ so that $n \geq s$ and

$$\{a^s_0, b^s_0, \ldots, a^s_n, b^s_n\} = \{\sigma_0, \ldots, \sigma_{2n+1}\} = B_s.$$

Moreover, assume we have defined a linear order $< = <_{\mathcal{L}}$ on $B_s \times B_s$ so that

$$a^s_0 < a^s_1 < \cdots < a^s_n < b^s_n < b^s_{n-1} < \cdots < b^s_0.$$

Look for a $p \leq s$ such that Γ_p is inactive at stage s and $\phi^s_k(a^s_p)\downarrow$, where $p = [e, k]$.

If there is no such p, then for all z, let Γ_z be inactive at stage $s+1$ if and only if Γ_z is inactive at stage s. If Γ_z is active, let $\Gamma_z(s+1) = \Gamma_z(s)$. In addition, let $a^{s+1}_i = a^s_i$ and $b^{s+1}_i = b^s_i$ for all $i \leq n = n_s$. Finally, let $n_{s+1} = n + 1$, $\sigma_{2n+2} = a^{s+1}_{n+1}$ and $\sigma_{2n+3} = b^{s+1}_{n+1}$ and extend our definition of $< = <_{\mathcal{L}}$ to $B_{s+1} \times B_{s+1}$ by declaring

$$a^{s+1}_0 < \cdots < a^{s+1}_{n+1} < b^{s+1}_{n+1} < \cdots < b^{s+1}_0.$$

If there is such a p, let $p = p(s+1) = [e(s+1), k(s+1)] = [e, k]$ be the least such p and $x = x(s+1) = \phi_k(a_p^s)$. If $x(s+1) \notin A$, then proceed exactly as in the case where $p(s+1)$ is not defined, except declare Γ_p active and let $\Gamma_p(s+1) = a_p^{s+1}$. If $x(s+1) \in A$, then find the first $4n_s + 4$ elements of A. Now compare these elements to x with respect to R_e. We will then be in one of two cases.

Case 1. There are h $= 2n_s + 2$ elements $v_0, \ldots, v_{h-1}$ among the first $4n_s + 4$ elements of A such that $(v_i, x) \in R_e$ for $i = 0, 1, \ldots, h-1$. In this case, we proceed exactly as in the case where $x(s+1) \notin A$.

Case 2. Otherwise, there must be $h+1$ elements $w_0, \ldots, w_{h+1}$, among the first $4n_s + 4$ elements of A, such that $(w_i, x) \notin R_e$ for all i. Then we let $a_i^{s+1} = a_i^s$ for $i < p$ and $b_j^{s+1} = b_j^s$ for all $j \leq n = n_s$. Set $n_{s+1} = 2n+1-p$. Let $b_{n+i}^{s+1} = a_{n-i+1}^s$ for $i = 1, \ldots, n+1-p$ and let $a_{p+j}^{s+1} = \sigma_{2n+2+j}$ for $j = 0, \ldots, 2n+1-2p$. Activate the Γ_p marker and place it on $b_{n_{s+1}}^{s+1} = a_p^s$. Remove any markers Γ_z that were on elements among $a_p^s, \ldots, a_n^s$ and make them inactive. Any marker Γ_z which was active at stage s where $\Gamma_z(s) \in \{a_0^s, \ldots, a_{p-1}^s, b_0^s, \ldots, b_n^s\}$ is still active at stage $s+1$ and $\Gamma_z(s+1) = \Gamma_z(s)$. Markers Γ_z where $z \neq p$ which were inactive at stage s remain inactive at stage $s+1$. Finally, extend $< = <_{\mathcal{L}}$ to $B_{s+1} \times B_{s+1}$ by declaring

$$a_0^{s+1} < \cdots < a_{n_{s+1}}^{s+1} < b_{n_{s+1}}^{s+1} < \cdots < b_0^{s+1}.$$

This complete our construction. Because A is a polynomial time set and each R_e is a polynomial time relation, it easily follows that each stage is completely effective. The following facts are easily proved by induction.

(1) For all s, $n_s \geq s$.

(2) For all s, $\{a_0^s, b_0^s, \ldots, a_{n_s}^s, b_{n_s}^s\} = \{\sigma_0, \ldots, \sigma_{2n_s+1}\}$.

(3) Our definition of $<_{\mathcal{L}}$ is consistent, that is, if $i, j \leq 2n_s + 1$ and stage s, we declare $\sigma_i <_{\mathcal{L}} \sigma_j$, then for all $t \geq s$, we declare $\sigma_i <_{\mathcal{L}} \sigma_j$ at stage t.

Note that these facts imply that $\mathcal{L} = (\{0,1\}^*, <^{\mathcal{L}})$ is a recursive linear ordering, because to decide if $\sigma_i < \sigma_j$, we simply go to stage $s = max\{i, j\}$ and then $\sigma_i < \sigma_j$ if and only if at stage s, we declare $\sigma_i < \sigma_j$.

Next we prove two lemmas which will complete the proof that $\mathcal{L}$ has the desired properties.

Lemma 5.10 *For each z, $\lim_s a_z^s = a_z$ and $\lim_s b_z^s = b_z$ exist and there is a stage t_z such that either Γ_z is inactive at stage s for all $s \geq t_z$ or Γ_z is active at stage s and $\Gamma_z(s) = \Gamma_z(t_z)$ for all $s \geq t_z$.*

Proof: We proceed by induction on $z = [e, k]$. By induction, we can assume that there is a stage $u > z$ large enough so that

(i) $a_j^s = a_j^u$ and $b_j^s = b_j^u$ for all $j < z$ and $s \geq u$ and

(ii) for each $j < z$, either Γ_j is inactive at stage s for all $s > u$ or for all $s \geq u$, Γ_j is active at stage s and $\Gamma_j(s) = \Gamma_j(u)$.

Note that our construction ensures that a Γ_j marker can be removed from an element at stage s only if $\Gamma_j(s-1) = a_k^s$ for some k and we take action to meet a requirement $\Gamma_{p(s)}$ at stage s where $p(s) < k$. Similarly, the only way $a_k^s \neq a_k^{s+1}$ is if $p(s+1) \leq k$ and we act according to Case 2 at stage $s+1$. Moreover, our construction ensures that if $j \leq n_s$, then $b_j^t = b_j^s$ for all $s \geq t$. It follows that $\lim_s b_z^s = b_z^u$. Now if Γ_z is active at stage s, then our choice of u ensures $p(s) > z$ for all $s > u$ so that $\lim_s a_z^s = a_z^u$ and Γ_z is active for all $s > u$. If Γ_z is not active at stage u, then either

(i) $\phi_k^s(a_z^u) \uparrow$ for all $s \geq u$, in which case, for all $s \geq u$, Γ_z is inactive at stage s, $p(s) > z$ and $a_z^s = a_z^u$, or

(ii) There is an $s > u$ such that $\phi_k^s(a_z^u) \downarrow$.

In case (ii), let t be the least $s > u$ such that $\phi_k^t(a_z^u) \downarrow$. Then our choice of u ensures that, for all $u \leq s \leq t$, $p(s) > z$ and Γ_z is inactive at stage s, so that $p(t+1)$ will be defined and $p(t+1) = z$. But then at stage $t+1$, Γ_z becomes active and is placed on either a_z^{t+1} or $b_{n_{t+1}}^{t+1}$. If Γ_z is placed on a_z^{t+1}, then $a_z^{t+1} = a_z^t$ and Γ_z will never be removed from a_z^{t+1}. This is because Γ_z can be removed from a_z^{t+1} only if $p(s) < z$ for some $s > t+1$, which is ruled out by our choice of u. If Γ_z is placed on $b_{n_{t+1}}^{t+1}$, then again Γ_z can never be removed from $b_{n_{t+1}}^{t+1}$. Thus in either case Γ_z will remain active for all $s \geq t+1$. But this means $p(s) > z$ for all $s \geq t+1$, so that $a_z^s = a_z^{s+1}$ for all $s \geq t+1$. $\square$

Lemma 5.11 *Requirement P_p is satisfied for all p.*

Proof: Let $p = [e,k]$ and let s_p be a stage such that $s_p > p$ and

(i) $(\forall s \geq s_p)(\forall j \leq p)[a_j^s = a_j^{s_p}$ and $b_j^s = b_j^{s_p}]$ and

(ii) $s_p \geq max\{t_o, \ldots, t_p\}$, where t_z is a stage such that either

- **(a)** for all $s \geq t_z$, Γ_z is active at s, or
- **(b)** for all $s \geq t_z$, Γ_z is inactive at s.

It then easily follows from our construction that if Γ_p is inactive at stage s_p, then $\phi_k^s(a_p^s) \uparrow$ and $a_p^s = a_p^{s_p}$ for all $s \geq s_p$. Thus $\phi_k(a_p) \uparrow$, where $a_p = \lim_s(a_p^s)$. Hence, the requirement P_p is automatically satisfied. If Γ_p is active at stage s_p, then there are two possibilities. The first is that $\Gamma_p(s) = a_p^s = a_p$, in which case our construction guarantees that either $\phi_k^s(a_p) \notin A$ or $\phi_k^s(a_p) \in A$ but there are at least $p+1$ elements $v_0, \ldots, v_p \in A$ such that $(v_i, \phi_k(a_p)) \in R_e$ for $i = 0, \ldots, p$.

The second possibility is that $\Gamma_p(s) = b^s_m = b_m$ for some $m \leq n_{s_p}$. In this case, our construction ensures that $\phi^s_k(b^s_m) \in A$ and there are at least $m+2$ elements $w_0, \ldots, w_{m+1} \in A$ such that $(w_i, \phi_k(b_m)) \notin R_e$ for $i = 0, \ldots, m$. Thus in any case, P_p is satisfied. □

It now follows from Lemma 5.10 that $a_i <_{\mathcal{L}} a_{i+1}$ and $b_{i+1} <_{\mathcal{L}} b_i$ for all i and that $a_i <_{\mathcal{L}} b_j$ for all i and j, so that $\mathcal{L}$ is isomorphic to $\omega + \omega^*$. By our remarks preceding the construction of $\mathcal{L}$, it follows that meeting all the requirements P_p ensures that $\mathcal{L}$ is not recursively isomorphic to any polynomial time linear ordering over A.

In the general case, if λ is a recursive linear ordering, then a finite injury priority argument very similar to the one given above for $\omega + \omega^*$ will prove the theorem. However, the only thing that we can conclude from the fact that $\omega + \mathbb{Z} \cdot \eta + \omega^*$ is a recursive linear ordering is that λ is a Π^0_2 linear ordering and in that case, a more complicated infinite injury priority argument is required. □

We note that in the special case of $\omega + \omega^*$, one can make K have universe $Tal(\omega)$ when $A = Bin(\omega)$ (This is Theorem 2.2 of [69]).

Cenzer and Remmel [16] gave a general condition which implies that a relational structure is recursively isomorphic to a p-time structure with universe $Bin(\omega)$. This condition can be thought of as a generalization of the argument in case 1 of Gregorieff's Theorem on linear orderings.

Definition 5.12 *Let $\mathcal{A}$ be a recursive substructure with universe A of the recursive relational structure $\mathcal{M}$ with universe M. Then $\mathcal{A}$ is said to be a highly recursive relatively indiscernible binary substructure of $\mathcal{M}$ if*

(1) There is a recursive map f from $\mathcal{A}$ to $Bin(\omega)$ which induces a p-time model $\widetilde{\mathcal{M}}$. (Let $a_i = f^{-1}(bin(i))$.)

(2) For any m-ary relation $R(x_1, \ldots, x_m)$ of $\mathcal{M}$, any fixed sequence $b_1, \ldots, b_k \in M \setminus A$ with $k \leq m$ and any fixed sequence $1 \leq I_1 < \ldots < I_k \leq m$, let $R^{b_1,\ldots,b_k}_{I_1,\ldots,I_k}$ denote the $(m-k)$-ary relation on $M - A$ which results by substituting b_{I_j} for x_{I_j} for $j = 1, \ldots, k$.

Then, for any such R, $1 \leq I_1 < \ldots < I_k \leq m$ and $b_1, \ldots, b_k \in M \setminus A$, either $R^{b_1,\ldots,b_k}_{I_1,\ldots,I_k}$ holds for all but finitely many elements in $(M-A)^{m-k}$, or $\neg R^{b_1,\ldots,b_k}_{I_1,\ldots,I_k}$ holds for all but finitely many elements in $(M-A)^{m-k}$.

Furthermore, there is a uniform effective algorithm which, given an index for R and sequences $b_1, \ldots, b_k \in M \setminus A$ and $I_1, \ldots, I_k$, will compute whether $R^{b_1,\ldots,b_k}_{I_1,\ldots,I_k}$ holds for all but finitely many elements in $(M-A)^{m-k}$ or $\neg R^{b_1,\ldots,b_k}_{I_1,\ldots,I_k}$ holds for all but finitely many elements in $(M-A)^{m-k}$, along with with a complete list of the finitely many sequences of $(M-A)^{m-k}$ which are exceptions.

Theorem 5.13 *Suppose that $\mathcal{A}$ is a highly recursive relatively indiscernible binary substructure of the relational structure $\mathcal{M}$. Then $\mathcal{M}$ is recursively isomorphic to a p-time structure with universe $Bin(\omega)$ (and therefore also recursively isomorphic to a p-time structure with universe $Tal(\omega)$).*

Proof: There is no loss in generality in assuming the universe of $\mathcal{M}$ is the set of natural numbers ω. Let $S_0, S_1, \ldots$ be an effective list of all relations of $\mathcal{A}$ and assume that there is a recursive function F such that S_i is an $F(i)$-ary relation. Let $\mathcal{A}$ with universe $A = \{a_0, a_1, \ldots\}$ be isomorphic to $\widetilde{\mathcal{M}}$ as described above and let $\omega \setminus A = \{b_0 < b_1 < \ldots\}$. For any $x \in \omega \setminus A$, let $t(x)$ be the total time needed to search all elements $y \leq x$ and determine if $y \in A$ and to compute the common value of all relations of the form $R^{b_1,\ldots,b_k}_{I_1,\ldots,I_k}$ and the finite list of exceptions where $R = S_i$ for some $i \leq x$ and $b_1, \ldots, b_k$ are elements of $M - A$ which are less than or equal to x. Let $\phi(x) = 1^{[x,t(x)]}$ and let $P = \{\phi(x) : x \in M \setminus A\}$. It is clear that P is a p-time subset of $Tal(\omega)$.

We define a polynomial time model $\mathcal{D}$ with universe $Bin(\omega)$ by defining the relation $S^{\mathcal{D}}_q$ for $\mathcal{D}$ as follows. Given an element $bin(i)$, search the strings of the form 1^k for $k \leq |bin(i)|$ and determine whether each such string is in P. If $bin(i) \in P$, then let $r(bin(i)) = bin(n)$ where $bin(i) = 1^{[b_n,t(b_n)]}$. If $bin(i) \notin P$, then let $r(bin(i)) = bin(m)$ where $bin(i)$ is the m-th element of $Bin(\omega) \setminus P$. Note that because P is a polynomial time set, there is a fixed polynomial p such that we can compute whether $x \in P$ in $p(|x|)$ steps. It thus takes at most $\sum_{j=0}^{|bin(i)|} p(j)$ steps to search the strings of the form 1^k for $k \leq |bin(i)|$ for membership in P so that the function r is polynomial time. Then $S^{\mathcal{D}}_q(s_1, \ldots, s_{F(q)})$ holds if either

(A) no $s_i \in P$ and $S_q(r(s_1), \ldots, r(s_{F(q)}))$ holds in $\widetilde{\mathcal{M}}$, or

(B) there is some s_i in P and $S_q(Z_1, \ldots, Z_{F(q)})$ holds in $\mathcal{M}$ where $Z_j = b_k$, if $s_j \in P$ and $r(s_j) = bin(k)$, and $Z_j = a_k$, if $s_j \notin P$ and $r(s_j) = bin(k)$.

Note that in case (A) we can compute whether $S^{\mathcal{D}}_q(s_1, \ldots, s_{F(q)})$ holds in time polynomial in $|s_1| + \cdots + |s_{F(q)}|$ since $\widetilde{\mathcal{M}}$ is a polynomial time model. In case (B), let n be the maximum value such that there is an $s_j \in P$ with $j \leq F(q)$ and $r(s_j) = bin(n)$. If $n \geq q$, then $s_j = 1^{[b_n,t(b_n)]}$ and in $t(b_n)$ steps we can compute whether $R_q(s_1, \ldots, s_{F(q)})$ holds by the definition of t. Thus in case (B), the only cases in which we can not directly compute in linear time whether $S^{\mathcal{D}}_q(s_1, \ldots, s_{F(q)})$ holds is if $n \leq q$. However it is easy to see that our assumptions ensure that it takes only a finite amount of information to determine whether $S^{\mathcal{D}}_q(s_1, \ldots, s_{F(q)})$ holds in all such cases. Thus $S^{\mathcal{D}}_q$ is a polynomial time relation.

Finally it is easy to see that our definitions ensure that we have defined the map g where $g(bin(i)) = b_n$, if $bin(i) \in P$ and $r(bin(i)) = bin(n)$, and $g(bin(i)) = a_n$, if $bin(i) \notin P$ and $r(bin(i)) = bin(n)$, is an isomorphism from $\mathcal{D}$ onto $\mathcal{M}$.

The model $\mathcal{C}$ with universe $Tal(\omega)$ simply has relations $R^{\mathcal{C}}$ defined by

$$R^{\mathcal{C}}(tal(i_1, \ldots, tal(i_n)) \iff R^{\mathcal{D}}(bin(i_1), \ldots, bin(i_n)).$$

The relation $R^{\mathcal{C}}$ is p-time since $bin(i)$ can be computed from $tal(i)$ in polynomial time and $|bin(i)| \leq |tal(i)|$. $\square$

For a simple example, consider a well-ordering $\mathcal{M} = (M, <^M)$ of type $> \omega$ and let $\mathcal{A} = (A, <^M)$ be the first ω elements of $\mathcal{M}$. A is a recursive set since

$x \in A \iff x <^M w$ where w is the ω-th element of $\mathcal{M}$. $\mathcal{A}$ is certainly recursively isomorphic to the standard ordering on $Bin(\omega)$ and is indiscernible since for all $a \in A$ and all $x \in M \setminus A$, $a <^M x$. Thus by Theorem 5.13, $\mathcal{M}$ is recursively isomorphic to a p-time model with standard universe $Bin(\omega)$. This is a special case of Theorem 5.9 above.

5.3 Boolean Algebras

There are some cases of structures with function symbols where we can get some positive results. For example, every recursive Boolean algebra is recursively isomorphic to a polynomial time Boolean algebra.

Definition 5.14 (i) *The language $\mathcal{L}$ of Boolean algebras consists of two binary function symbols $\wedge$ (meet) and $\vee$ (join), one unary function symbol $\neg$ (complement) and two constant symbols 0 (zero) and 1 (unity). A Boolean algebra $\mathcal{B}$ is a structure $(B, \wedge^B, \vee^B, \neg^B, 0^B, 1^B)$ for this language which satisfies the usual axioms.*

(ii) *Given a linear ordering $\mathcal{M} = (\mathcal{M}, <)$ with a first element, the interval algebra $Intalg(\mathcal{M})$ is the Boolean algebra of subsets of M generated by the left-closed, right-open intervals of M, [a,b) $= \{x : a \leq x < b\}$.*

The partial ordering $\leq^B$ of a Boolean algebra $\mathcal{B}$ is defined by $a \leq^B b$ if and only if $b = a \vee c$ for some $c \neq 0$. An element a of a Boolean algebra is said to be an *atom* if whenever $b \leq a$, either $b = 0$ or $b = a$. The Boolean algebra $\mathcal{A}$ is said to be *atomic* if for any $b \neq 0$, there exists some atom a with $a \leq b$. A element $x \in \mathcal{B}$ is *atomless* if x is not the zero of $\mathcal{B}$ and there is no atom a of $\mathcal{B}$ such that $a <^B x$. $\mathcal{B}$ is said to be *non-atomic* if $\mathcal{B}$ contains an atomless element.

The following is Lemma 2.5 of [10].

Lemma 5.15 *For any p-time linear ordering $\mathcal{L}$ with a first element, the interval algebra $Intalg(\mathcal{L})$ is a p-time Boolean algebra.*

Sketch of Proof: The nonzero elements of $Intalg(\mathcal{L})$ are given the natural representation $[a_1, a_2) \cup [a_3, a_4) \ldots \cup [a_{2n-1}, a_{2n})$, where $a_1 <^L a_2 <^L \ldots <^L a_{2n-1}$ and either $a_{2n} = \infty$ or $a_{2n} \in L$ and $a_{2n-1} < a_{2n}$. $\square$

This lemma is used to prove the following result from [10].

Theorem 5.16 *Every recursive Boolean algebra $\mathcal{B}$ is recursively isomorphic to a p-time Boolean algebra.*

Sketch of Proof: First observe that the classical proof that every countable Boolean algebra is isomorphic to the interval algebra of a countable linear ordering is effective. (See Remmel [67].) Thus every recursive Boolean algebra is recursively isomorphic to Intalg($\mathcal{M}$) where $\mathcal{M}$ is a recursive linear ordering.

However by Theorem 5.8, $\mathcal{M}$ is recursively isomorphic to a polynomial time linear ordering $\mathcal{P}$. The interval algebra of $\mathcal{P}$ is thus recursively isomorphic to $\mathcal{B}$ and is polynomial time by Lemma 5.15.

□

The next two theorems are unpublished results due to Cenzer and Remmel.

Theorem 5.17 *Every infinite* **non-atomic** *recursive Boolean algebra is recursively isomorphic to a p-time Boolean algebra (a) with universe $Bin(\omega)$; (b) with universe $Tal(\omega)$.*

Theorem 5.18 *Let $A \subseteq \{0,1\}^*$ be an infinite p-time set and let B be an infinite* **atomic** *recursive Boolean algebra. Then there is a recursive Boolean algebra $\mathcal{D}$ which is isomorphic to $\mathcal{B}$ but is not recursively isomorphic to any p-time Boolean algebra with universe A.*

5.4 Graphs

Next we give two applications of Theorem 5.13 to recursive graphs due to Cenzer and Remmel in [16].

Definition 5.19 **(i)** *A graph $G = (V, E)$ is* **locally finite** *(respectively* **locally cofinite***) if for every $v \in V$, the set $NB(v)$ of neighbors of v is finite (resp. cofinite).*

(ii) *A graph $G = (V, E)$ is* **locally finite/cofinite** *if for every $v \in V$, either the set $NB(v)$ of neighbors of v is finite or $V \setminus NB(v)$ is finite.*

(iii) *A locally finite/cofinite recursive graph G is* **highly recursive** *if there are algorithms for deciding whether a given $v \in V$ has finite degree and for computing $NB(v)$ is the degree is finite and $V \setminus NB(v)$ if not.*

Theorem 5.20 *Every infinite highly recursive locally finite/cofinite graph G is recursively isomorphic to a p-time graph H with universe $Bin(\omega)$.*

Sketch of Proof: If there are infinitely many vertices of G whose degree is finite, then we can construct a recursive subset of such vertices U such that G restricted to U is the empty graph. Moreover we can construct U so that it will be the highly recursive relatively indiscernible binary substructure required to apply Theorem 5.13. If there are infinitely many vertices of finite co-degree. We can construct a a complete subgraph C which will be the highly recursive relatively indiscernible binary substructure needed for Theorem 5.13. □

Theorem 5.21 *Every infinite recursive graph G which is either locally finite or locally cofinite, is recursively isomorphic to a p-time graph H with universe $Bin(\omega)$.*

Sketch of Proof: Suppose that all vertices have finite co-degree. Then we again pick out a complete subgraph $C = (\{v_0, v_1, \ldots\}, E)$, now with the additional property that v_i is not joined to any vertex from $\{0, 1, \ldots, i-1\}$. $\Box$

The next result shows that the hypotheses of the two preceding theorems are needed.

Theorem 5.22 *Let A be any infinite polynomial time subset of $\{0,1\}^*$. Then there is a recursive graph, having every vertex of either finite degree or finite co-degree, which is not recursively isomorphic to any p-time graph with universe A.*

5.5 Trees

A connected graph G with no cycles is said to be a *tree*. The vertices of T are called *nodes*. We will assume that any tree T has a designated *root* ϵ. Then any node v of T can be reached from the root by a *path*, that is, a sequence of edges $(\epsilon, v_1), (v_1, v_2), \ldots, (v_{n-1}, v)$. We say that v is a *successor* of v_{n-1}. It is clear that the successor relation is recursive, since the path from the root to a node v may be computed from v in a uniform fashion. The partial ordering $u \prec^T v$, which says that there is a path from the root to v which passes through u, is also recursive. On the other hand, if T is a p-time tree, then this computation of the path from ϵ to v might not be in polynomial time in $|v|$, so that the successor relation and the relation $\prec^T$ need not be p-time. Thus we say that T is *fully* p-time if both the successor relation and the relation $\prec^T$ are p-time. Similar notions may be defined for any bounded resource class.

T is said to be *highly recursive* if there is a recursive function which computes from any node v a list of successors of v. The corresponding notion of a highly feasible tree (or more generally, of a highly feasible graph) is more difficult to formulate. Several inequivalent notions are studied in [12, 13, 16]. In particular, a graph is said to be *locally p-time*, if there is a p-time function which computes from any node v a list of successors of v and is said to be *highly p-time* if there is a p-time function which computes from a vertex v and a tally number $tal(n)$ a list of all vertices at distance n from v.

The following results are Theorems 9, 10 and 12 of [16].

Theorem 5.23 *Any infinite recursive tree T is recursively isomorphic to a p-time tree with universe $Bin(\omega)$.*

Sketch of Proof: There are two cases. First, every node of T may have only finitely many successors. In this case, Theorem 5.20 may be applied. Second, some node v of T may have an infinite set A of successors. In this case, the set A plays the role of the highly recursive relatively indiscernible binary substructure, so that Theorem 5.13 may be applied. $\Box$

Theorem 5.24 *There is a highly recursive tree T which is not recursively isomorphic to any fully primitive recursive tree P with a standard universe $Bin(\omega)$.*

Theorem 5.25 *There is a highly recursive binary tree which is not isomorphic as a directed graph to any highly primitive recursive tree with universe $Bin(\omega)$.*

A similar but stronger result for graphs was given in Theorem 2.3 of [13].

Theorem 5.26 *There is a highly recursive graph G which is not isomorphic to any locally primitive recursive graph.*

5.6 Equivalence Structures

Another type of relational structure is an *equivalence structure,* (A, R^A), where R^A is an equivalence relation on A. A recursive equivalence structure (A, R^A) is said to be *highly recursive* if the set of elements that belong to infinite equivalence classes is recursive, and there is a recursive function f such that $f(a)$ is the cardinality of $[a]^R$ when $[a]^R$ is finite (so that the equivalence class $[a]^R$ can be computed from a). We say that (A, R^A) is *highly p-time* if A is a p-time subset of $Bin(\omega)$, the set of elements that belong to infinite equivalence classes is p-time, and there is a p-time function f such that $f(a)$ codes $[a]^R$ when it is finite. The *full spectrum* of of (A, R^A) is the set of pairs $(0, n)$ such that there are at least $n + 1$ infinite equivalence classes in (A, R^A) and pairs (q, n) such that $q > 0$ and there are at least $n + 1$ equivalence classes of size q in (A, R^A).

The following results are Theorems 13, 14 and 16 of [16].

Theorem 5.27 *Any recursive equivalence structure (A, R^A) is recursively isomorphic to a p-time model with universe $Bin(\omega)$.*

Sketch of Proof: There are two cases. If all equivalence classes of (A, R^A) are finite, then Theorem 5.21 can be applied to (A, R^A) viewed as a graph. If (A, R^A) has an infinite equivalence class, then this class is a highly recursive relatively indiscernible substructure so that Theorem 5.13 may be applied. □

Theorem 5.28 *For any equivalence structure (A, R^A) with full spectrum S such that $S^* = \{\langle tal(q), tal(r)\rangle : (q, r) \in S\}$ is p-time, there is a highly recursive, p-time equivalence structure $(Bin(\omega), R)$ isomorphic to (A, R^A).*

Theorem 5.29 *There is an infinite recursive full spectrum S of ω such that no highly primitive recursive equivalence structure with universe $Bin(\omega)$ has full spectrum a subset of S.*

We now turn to the study of some structures with functions. There are three basic models which we have been considered: first, models of some fragment of arithmetic; second, Abelian groups; and, third, permutation structures.

5.7 Models of Arithmetic

Our first result, taken from [10], demonstrates that the unary exponential function 2^x may be adjoined to the standard model of arithmetic while being represented by a p-time function.

Theorem 5.30 $\mathbb{N} = (\omega, S, +, -, \cdot, 2^x, <, 0)$ *is isomorphic to a p-time structure* $\mathcal{M}$.

Sketch of Proof: The elements of $\mathcal{M}$ are terms in the language $\{0, A, I, E\}$. The natural number n is represented by the expression $\sigma(n)$ defined as follows.

$\sigma(0) = 0$

$\sigma(2^k) = E\sigma(k)$

$\sigma(2^k + m) = AE\sigma(k)\sigma(m)$ for $0 < 2m \leq 2^k$.

$\sigma(2^k - m) = IE\sigma(k)\sigma(m)$ for $0 < 2m < 2^k$.

It is easy to see that the universe of $\mathcal{M}$ is a linear time set and that the term $\sigma(n)$ which represents n can be computed from $bin(n)$ in polynomial time. It can be shown by induction that $|\sigma(n)| \leq 2|bin(n)|^2$ and that $|\sigma \pm^M \tau| \leq |\sigma| + |\tau| + 1$. □

We note that it is an open question whether $(\omega, S, +, -, \cdot, exp, <, 0)$ is is isomorphic to a polynomial time model where $exp(m, n) = m^n$ is the general exponential or even whether $(\omega, S, +, -, \cdot, 2^x, 3^x, <, 0)$ is isomorphic to a polynomial time. Bäuerle [5] proved that in the model of Theorem 5.30 the function 3^x is not polynomial time.

We also note that the model of Theorem 5.30 can be used to build a $EXPTIME$ group isomorphic to the integers under $+$, $\mathbb{Z} = (Z, +)$, which is not q-time isomorphic to the standard model of $\mathbb{Z}$ where the positive integer n is coded as $bin(2n)$ and a negative integer $-n$ is coded as $bin(2n - 1)$; see section 6.3.

5.8 Injection Structures

The simplest type of structure with a function is an *injection structure* (A, f^A) where f^A is a one-to-one mapping from A into itself. If f^A maps A onto A, then we say that (A, f^A) is a *permutation structure*. The *orbit* $\mathcal{O}(a)$ of an element a of A is $\{b \in A : (\exists n \in \omega)(f^n(a) = b \vee f^n(b) = a)\}$. There are two types of infinite orbits, one of type ω which is isomorphic to (ω, S) and the other of type $\mathbb{Z}$ which is isomorphic to $(\mathbb{Z}, S)$. The *order* $|a|$ of a is $card(\mathcal{O}(a))$ if $\mathcal{O}(a)$ is finite and is either ω or $\mathbb{Z}$ otherwise. The *full spectrum* of (A, f^A) is the set of pairs $(0, n)$ such that there are at least $n + 1$ orbits of type ω, pairs $(1, n)$ such that there are at least $n + 1$ orbits of type $\mathbb{Z}$, and pairs (q, n) such that $q > 1$ and there are at least $n + 1$ orbits of size $q - 1$ in (A, f^A).

It is easy to see that the full spectrum of a recursive injection structure is always a recursively enumerable set. It is shown in Theorem 3.4 of [14] that any r.e. spectrum can be realized by a p-time injection structure. Thus every recursive injection structure is isomorphic to a p-time structure. However, we know by Theorem 5.3 that this isomorphism need not be recursive. Now we consider the question of when we can obtain a p-time injection structure with a standard universe $Bin(\omega)$ or $Tal(\omega)$.

The basic result here is Theorem 3.2 of [11].

Theorem 5.31 *Any recursive permutation structure (A, f^A) with all finite orbits is recursively isomorphic to an honest p-time structure with universe a subset of $Tal(\omega)$.*

Sketch of Proof: The element $a \in A$ is represented by $tal(n)$, where $bin(n) = 1^a 0^t$ and t is the total time required to compute the orbit of a. The details follow as in the proof of Theorem 5.1 above. $\square$

Theorems 3.4 and 3.6 of [11] give two cases in which we can improve this result to obtain a standard universe.

Theorem 5.32 *Let $B = Bin(\omega)$ or $Tal(\omega)$. Any recursive injection structure (A, f) with at least one but only finitely many infinite orbits is recursively isomorphic to a p-time structure with universe B.*

Sketch of Proof: Let $F = \{a \in A : |a| \text{ is finite}\}$ and let $I = A \setminus F$. Since there are only finitely many orbits in I, both F and I are recursive. It is easy to see that (I, f) is recursively isomorphic to a p-time structure with universe B. By Theorem 5.31, (F, f) is recursively isomorphic to a p-time structure with universe a subset of $Tal(\omega)$. The result now follows from Lemma 4.8. $\square$

Theorem 5.33 *Let $B = Bin(\omega)$ or $Tal(\omega)$. Any recursive injection structure (A, f) with infinitely many orbits of size q, for some finite q, is recursively isomorphic to a p-time structure with universe B.*

Sketch of Proof: Let $C = \{a \in A : |a| = q\}$ and let $D = A \setminus C$. It follows from Theorem 5.31 that (D, f) is recursively isomorphic to a p-time structure with universe a subset of $Tal(\omega)$. It therefore suffices by Lemma 4.8 to show that C is recursively isomorphic to a p-time structure (B, g). For $B = Bin(\omega)$, the permutation g is simply defined by $g(bin(nq + i)) = bin(nq + i + 1)$, if $i + 1 < q$, and $g(bin(nq + q - 1) = bin(nq)$. The tally definition is similar. $\square$

A general result on the existence of p-time injection structures is given by Theorems 3.4 and 3.8 of [14].

Theorem 5.34 *(a) For any r.e. full spectrum S, there is a p-time injection structure (A, f) with full spectrum S.*

(b) If $\{\langle tal(q), tal(r)\rangle : (q, r) \in S\}$ is p-time, then A may be taken to be either $Bin(\omega)$ or $Tal(\omega)$.

Sketch of Proof: We sketch the proof of part (b) for $B = Tal(\omega)$, assuming that all orbits will be finite. Let $q_0, q_1, \ldots$ enumerate in non-decreasing order the set of orbit sizes (with repetitions). Then the permutation f may be defined by

$f(tal(q_0 + q_1 + \ldots + q_{k-1} + r)) = (tal(q_0 + q_1 + \ldots + q_{k-1} + r + 1))$ if $r < q_k$, and $= tal(q_0 + q_1 + \ldots + q_{k-1} + q_k))$, if $r = q_k$.

An infinite orbit of type ω is given by the standard successor function on $Tal(\omega)$ and an orbit of type $\mathbb{Z}$ is given by $f(tal(2n)) = tal(2n+2)$, $f(1) = 0$ and $f(tal(2n+3)) = tal(2n+1)$. Multiple infinite orbits and a combination of finite and infinite orbits may then be obtained by Lemma 4.8. $\square$

Corollary 5.35 *Any recursive injection structure is isomorphic to a p-time injection structure.*

Finally, we consider some negative results. The first is Theorem 3.13 of [11] and deals with structures with a fixed universe.

Theorem 5.36 *There is a recursive set M such that no injection structure with full spectrum M is isomorphic to any primitive recursive structure with universe $Bin(\omega)$ or $Tal(\omega)$.*

Sketch of Proof: Let f_e enumerate all primitive recursive unary functions and let $\mathcal{B}_e = (Bin(\omega), f_e)$. Construct a set $R = \{r_0 < r_1 < \ldots\}$ by a diagonal argument so that, for all e, either

(1) f_e is not one-to-one, or

(2) $\mathcal{B}_e$ has an infinite orbit, or

(3) $\mathcal{B}_e$ has two disjoint orbits of the same finite size, or

(4) $\mathcal{B}_e$ has an orbit of size $q \notin R$.

We establish this requirement, given $r_0 < \ldots < r_{e-1}$ by computing enough of $\mathcal{B}_e$ to either find two orbits of the same size, or an orbit (perhaps incomplete) of size $r > r_{e-1}$. If the orbit is complete, we let $r_e = r + 1$, thus ensuring that $\mathcal{B}_e$ has an orbit of size $r \notin R$. If the orbit is incomplete, we continue to build the orbit at later stages and take a similar action when the orbit becomes complete. If this never happens, then the orbit is infinite, so that (2) is satisfied.

It follows that no primitive recursive permutation structure with all finite orbits can have $M = \{(r, 1) : r \in R\}$ as a subset of its full spectrum. $\square$

The final result here is Corollary 3.18 of [11] and deals with the question of recursive isomorphism. The proof is omitted.

Theorem 5.37 *For any recursive injection structure (C, f) with infinitely many infinite orbits, there is a recursive structure (A, f^A) which is isomorphic to (C, f) but is not recursively isomorphic to any primitive recursive structure.*

5.9 Abelian Groups

We now turn to the study of feasible versus recursive Abelian groups. The results here are parallel to those for permutation structures. We begin by recalling some basic notation. For any natural number $n > 1$, $\mathbb{Z}(n)$ is the cyclic group of order n. For a prime number p, the group $\mathbb{Z}(p^\infty)$ is the inverse limit of the sequence $\mathbb{Z}(p^n)$, or more concretely, the set of rational numbers with denominator equal to a power of p and where the group operation is addition modulo 1. The group $\mathbb{Z}(p^\infty)$ is said to be *quasi-cyclic.* The additive group of rational numbers is denoted by $\mathbb{Q}$.

For any element a of an Abelian group $\mathcal{A} = (A, +^A, -^A, 0^A)$ and any integer n, $n \cdot a$ is defined recursively by $0 \cdot a = 0$ and $(n+1) \cdot a = a +^A n \cdot a$. Then $(-n) \cdot a = 0^A -^A n \cdot a$. The *order* $|a|_\mathcal{A}$ of a is the least n such that $n \cdot a = 0$. $\mathcal{A}$ is said to be *torsion* if all elements have finite order and *torsion-free* if all elements (except the identity) have infinite order.

We will frequently be concerned with products of Abelian groups.

Definition 5.38 *For any sequence $\mathcal{A}_0, \mathcal{A}_1, \ldots$ of Abelian groups, where $\mathcal{A}_i = (A_i, +_i, -_i, e_i)$ and $A_i \subseteq \{0,1\}^*$, the direct product $\mathcal{A} = \oplus_n \mathcal{A}_n$ is defined to have domain*

$$A = \{\langle \sigma_1, \sigma_2, \ldots, \sigma_k \rangle : k \in \omega, \sigma_i \in A_i \text{ for } 1 \leq i \leq k \text{ and } \sigma_k \neq e_k\}.$$

The identity of $\mathcal{A}$ is $e^A = \emptyset$, and addition $+^A$ and subtraction $-^A$ are defined as follows: for $\sigma = \langle \sigma_1, \sigma_2, \ldots, \sigma_m \rangle$ and $\tau = \langle \tau_1, \tau_2, \ldots, \tau_n \rangle$, $\sigma +^A / -^A \tau = \rho = \langle \rho_1, \rho_2, \ldots, \rho_k \rangle$, where $k = max\{i : (i \leq m \wedge i \leq n \wedge \sigma_i +_i / -_i \tau_i \neq e_i) \vee m < i \leq n \vee n < i \leq m\}$ and, for $i \leq k$,

$$\rho_i = \begin{cases} \sigma_i +_i / -_i \tau_i, & \text{for } i \leq min(m,n) \\ \sigma_i, & \text{for } n < i \leq k \\ \tau_i, & \text{for } m < i \leq k. \end{cases}$$

In particular, we write $\oplus_{i<\omega}\mathcal{G}$ to be the direct product of a countably infinite number of copies of $\mathcal{G}$.

Definition 5.39 *Let B be either $Bin(\omega)$ or $Tal(\omega)$. For any complexity class Γ, the sequence $\mathcal{A}_0, \mathcal{A}_1, \ldots$ of groups, where $\mathcal{A}_n = (A_n, +_n, -_n, e_n)$ is said to be* **uniformly Γ-computable over** B if

(i) $\{\langle b(n), a \rangle : a \in A_n\}$ is a Γ-computable subset of $B \otimes B$, where $b(n) = bin(n)$ if $B = Bin(\omega)$ and $b(n) = tal(n)$ if $B = Tal(\omega)$.

(ii) The functions $F(b(n), a, b) = a +_n b$ and $G(b(n), a, b) = a -_n b$ are both the restrictions of Γ-computable functions from B^3 into B where we set $F(b(n), a, b) = G(b(n), a, b) = \emptyset$ if either a or b is not in A_n.

(iii) The function from $Tal(\omega)$ into B given by $e(tal(i)) = e_i$ is Γ-computable.

The following is Lemma 4.2 of [11].

Lemma 5.40 *Let B be either $Bin(\omega)$ or $Tal(\omega)$ and let Γ be one of the following complexity classes: recursive, primitive recursive, exponential time, polynomial time. Suppose that the sequence $\mathcal{A}_i = (A_i, +_i, -_i, e_i)$ of Abelian groups is Γ-computable over B. Then*

(a) *The direct product $\mathcal{A}$ of the sequence is recursively isomorphic to a Γ-computable group with universe contained in B.*

(b) *If A_i is a subgroup of A_{i+1} for all i and if there is a Γ-computable function f such that, for all $a \in \cup_i A_i$, $a \in A_{f(a)}$, then the union $\cup_i \mathcal{A}_i$ is recursively isomorphic to a Γ-computable group with universe contained in B.*

(c) *If the sequence is finite and one of the components has universe B, then the product is recursively isomorphic to a Γ-computable group with universe B.*

(d) *If the sequence is infinite and if each component has universe B, then the product is recursively isomorphic to a Γ-computable group with universe $Bin(\omega)$.*

(e) *If each component has universe $Tal(\omega)$ and there is a uniform constant c such that for any i and any $a, b \in A_i$, $|a +_i b|$ and $|a -_i b|$ are both $\leq c(|a| + |b|)$, then the product is recursively isomorphic to a Γ-computable group with universe $Tal(\omega)$.*

If a torsion Abelian group $\mathcal{G}$ is isomorphic to a direct sum $\oplus_i \mathbb{Z}(q_i^{n_i})$ of prime power cyclic groups, then we define the *characteristic* $\eta(\mathcal{G})$ to be

$$\{(p^m, k) : q_i^{n_i} = p^m \text{ for at least } k+1 \text{ distinct values of } i\}.$$

Khisamiev shows in Corollary 3.4 of [40] that for any $k \leq \omega$ and any torsion Abelian p-group $\mathcal{G}$, $\mathcal{G} \oplus (\oplus_{i<k} \mathbb{Z}(p^\infty))$ is isomorphic to a recursive group if and only if $\eta(G)$ is a Σ_2^0 set. We will say that a subset Q of $\omega \times \omega$ is *hereditary* if $(m, k+1) \in Q$ implies $(m, k) \in Q$ for all m, k. It is clear that a subset Q of $\omega \times \omega$ is the characteristic $\eta(\mathcal{G})$ for some Abelian torsion group $\mathcal{G}$ if and only if Q is hereditary and $(m, k) \in Q$ implies m is a prime power. Therefore we will say that any such set Q is a *characteristic*.

The following are results 4.3 and 4.4 of [11].

Theorem 5.41 *Each of the groups $\mathbb{Z}$, $\oplus_{i<\omega} \mathbb{Z}(k)$, $\mathbb{Z}(p^\infty)$ and $\mathbb{Q}$ are isomorphic to polynomial time groups (a) with universe $Bin(\omega)$ and (b) with universe $Tal(\omega)$.*

Theorem 5.42 *Any finitely generated recursive Abelian group is recursively isomorphic to a p-time Abelian group (a) with universe $Bin(\omega)$ and (b) with universe $Tal(\omega)$.*

The simplest torsion groups are *primary* groups, or p-groups, in which every element has order a power of p where p is a prime. In [74], Smith characterized the recursively categorical p-groups as follows.

Theorem 5.43 *(Smith) A recursive p-group $\mathcal{G}$ is recursively categorical iff either*

(1) $\mathcal{G} \approx \oplus_{i<n}\mathbb{Z}(p^\infty) \oplus \mathcal{F}$ *or*

(2) $\mathcal{G} \approx \oplus_{i<n}\mathbb{Z}(p^\infty) \oplus \oplus_{i<\omega}\mathbb{Z}(p^m) \oplus \mathcal{F}$ *where $\mathcal{F}$ is a finite p-group and m and n are nonnegative integers.*

Corollary 5.44 *Any recursively categorical p-group is recursively isomorphic to a polynomial time group (a) with universe $Bin(\omega)$ and (b) with universe $Tal(\omega)$.*

Note that not every product of cyclic groups is recursively categorical. For example, consider $\oplus_{i<\omega}\mathbb{Z}(2) \oplus \oplus_{i<\omega}\mathbb{Z}(4)$. The following fundamental result from [11] shows that this group is recursively isomorphic to a p-time group.

Theorem 5.45 *Any recursive Abelian torsion group $\mathcal{G} = (G, +^G, -^G, e^G)$ is recursively isomorphic to a polynomial time group $\mathcal{H}$ with universe a subset of $Tal(\omega)$.*

Sketch of Proof: It suffices, by the remarks following Lemma 4.4, to define a p-time group $\mathcal{H}$ with universe a subset of $Bin(\omega)$ such that both $a +^{\mathcal{H}} b$ and $a -^{\mathcal{H}} b$ have length bounded by some constant multiple of $|a| + |b|$.

Let $\mathcal{A}_k$ be the subgroup generated by $\{1, 2, \ldots, k\}$. Renumber the elements of $\mathcal{A}$ as $a_0, a_1, \ldots$ so that the elements of $\mathcal{A}_k$ precede the elements of $\mathcal{A}_{k+1} \setminus \mathcal{A}_k$. This can be done so that the map taking i to a_i is a recursive isomorphism. Now map $a_k \in A$ to $\phi(a_k) = 1^k 0^{t(k)}$, where $t(k)$ is the total time required to compute the operation table for $\mathcal{A}_k$. The key to the proof is that whenever $a_i \pm^{\mathcal{G}} a_j = a_k$, where $i \leq j$, then we have $t(k) \leq t(j)$, since a_k will be in the group generated by $\{a_0, \ldots, a_j\}$. Furthermore $|\phi(a_k)| = k + t(k) \leq 2t(j) \leq 2(|\phi(a_i) + \phi(j)|)$ since $k \leq t(k)$. □

We need an effective version of the Fundamental Theorem of Abelian groups, which states that every torsion Abelian group is a direct product of primary groups. This is Lemma 4.8 of [11].

Theorem 5.46 *Any recursive Abelian torsion group $\mathcal{G}$ is recursively isomorphic to a p-time direct product of primary groups over B where B may be taken to be either $Tal(\omega)$ or $Bin(\omega)$.*

The main result on the existence of feasible groups with standard universe is the following theorem from [11].

Theorem 5.47 *Let $\mathcal{G}$ be an infinite recursive Abelian group with bounded order. Then $\mathcal{G}$ is recursively isomorphic to a polynomial time group with universe $Tal(\omega)$ and to a polynomial time group with universe $Bin(\omega)$.*

Sketch of Proof: Let B be either $Bin(\omega)$ or $Tal(\omega)$. We may assume that $\mathcal{G}$ is p-time by Theorem 5.45. Since the orders are bounded, there is no loss of generality in assuming that $\mathcal{G}$ is a p group for some prime p. Let p^m be the largest order of an element of $\mathcal{G}$. The proof is by induction on m. We can express $\mathcal{G}$ as a product $\mathcal{H} \oplus \mathcal{K}$, where $\mathcal{H}$ is generated by some independent set of elements of order p^m and $\mathcal{K}$ is maximal independent of $\mathcal{H}$ with no elements of order p^m. There are two cases.

Case 1. If $\mathcal{H}$ is finite, then $\mathcal{K}$ is infinite and may be assumed to have universe B by induction. The result now follows from Lemma 4.8.

Case 2. If $\mathcal{H}$ is infinite, then $\mathcal{H}$ is isomorphic to $\oplus_{i<\omega}\mathbb{Z}(p^m)$ and is therefore recursively isomorphic to a p-time group with universe B by Theorem 5.44. Since $\mathcal{K}$ is recursively isomorphic to a p-time group with universe a subset of $Tal(\omega)$ by Theorem 5.45, the result again follows from Lemma 4.8. $\Box$

Next we state some results on characteristics. The first result here follows from Theorem 5.45 together with the theorem of Khisamiev cited above.

Theorem 5.48 *For any Σ_2^0 characteristic Q, there is a p-time Abelian group with characteristic Q.*

We will show in Theorem 5.50 that not all recursive characteristics can be realized by p-time groups. The next result shows that any p-time characteristic can be so realized.

Theorem 5.49 *[14] Let Q be a nonempty, infinite characteristic such that $tal(Q) = \{tal([p^m, k]) : (p^m, k) \in Q\}$ is a p-time set. Then there exists a p-time Abelian group with characteristic Q and universe B where (a) $B = Bin(\omega)$ or (b) $B = Tal(\omega)$.*

Theorem 5.50 *[11] There is a recursive characteristic M such that no recursive group $\mathcal{G}$ with characteristic Q is can be isomorphic to any primitive recursive group with universe $Bin(\omega)$ or $Tal(\omega)$.*

Proof: Let $\mathcal{A}_0, \mathcal{A}_1, \ldots$ be an effective enumeration of the primitive recursive structures (ω, f_e) with one binary operation f_e. Let $Q_e = \{p : (p, 0) \in \eta(\mathcal{A}_e)\}$ and for any $a \in \mathcal{A}_e$, let $|a|_e$ be the order of a in $\mathcal{A}_e$. Construct a set $R = \{r_0 < r_1 < \ldots\}$ of prime numbers such that, for any e, $\mathcal{A}_e$ either

1. is not an Abelian group with identity 0, or

2. has an element of infinite order, or

3. has an element of order p^2 for some prime p, or

4. has two subgroups of the same prime order, or

5. has an element of prime order $p \notin Q$.

Now the order $|a|_e$ of an element a may not be a prime. Therefore we need some way to control the prime factors of $|a|_e$. We will now define a recursive function ν such that for any q and any $r > \nu(q)$, r either is divisible by p^2 for some prime p or r has a prime factor bigger than q. Given q, $\nu(q)$ is simply the product of all of the prime numbers $p \leq q$.

We will define the set Q in stages. At stage s, we will have $s+1$ elements $q_0 < q_1 < \cdots q_s$ in Q^s along with a certain finite subset I^s of $\omega \times \omega$ of restraints which will prevent numbers from coming into Q at stage s or at any later stage. Let $q_0 = 1$, $Q^0 = \{1\}$ and $I^0 = \emptyset$.

The initial stage of the construction proceeds as follows. Compute $f_0(1,1)$. There are then two cases.

(Case 1) If $f_0(1,1) = 0$, then we have $2 \in Q_0$ or else $\mathcal{A}_0$ is not an Abelian group with identity 0. Thus we can ensure that Q is not the characteristic of $\mathcal{A}_0$ by setting $q_1 = 3$ and restraining 2 from ever coming into Q. We let $I^0 = \emptyset$.

(Case 2) If $f_0(1,1) \neq 0$, then we know that either $\mathcal{A}_0$ is not an Abelian group with identity 0 or $|1|_0 > 2$ and therefore either has a prime factor $q > 2$ or is divisible by 4. Now let $q_1 = 2$ and let $I_0 = \{(1,0)\}$. This means that either $\mathcal{A}_0$ will have an element of order 4, thus satisfying part (4) of the requirement, or we will eventually restrain at least one of the prime factors of $|1|_0$ from ever coming into Q.

At stage $s+1$, we are given $q_0 < \cdots < q_s$ and the set I^s of previous restraints. Moreover, assume by induction that for every $(a,e) \in I^s$, either

(i) $|a|_e \leq \nu(q_s)$ and there is a prime factor $q < q_s$ of $|a|_e$ such that $q \neq q_i$ for any $i \leq s$, or

(ii) $|a|_e \leq \nu(q_s)$ and $|a|_e$ is divisible by the square of a prime, or

(iii) $|a|_e > \nu(q_s)$.

Now let $k = 1+\nu(q_s)^2$ and compute $i \cdot a$ in $\mathcal{A}_s$ for each $a \leq k$ and each $i \leq k$. This will produce a set of equivalence classes $[a]$, where a and b are equivalent if either $b = i \cdot a$ or $a = i \cdot b$ for some $i \leq k$. Note that every number $a \leq k$ belongs to some equivalence class, but numbers greater than k can also belong. Now all we need is that the computation of $i \cdot a$ in $\mathcal{A}_s$ produces a sequence of distinct elements up until it produces 0. If this is ever violated, then $\mathcal{A}_s$ is not an Abelian group, so that we will have satisfied the e-th requirement. In this case, we let $I^{s+1} = I^s$ and we choose q_{s+1} to be the least prime $p > q_s$ which does not violate any of the restraints $(t,b) \in I^{s+1}$. That is, q_{s+1} is the least q such that (i), (ii) or (iii) above are satisfied for all restraints in I^{s+1}. Otherwise, there are two further cases.

(Case 1) There is some equivalence class which has more than $\nu(q_s)$ elements. In this case, let a be the least such that $[a]$ has more than $\nu(q_s)$ elements. It follows that $|a|_s > \nu(q_s)$. Now put (a, s) into the set of restraints, so that $I^{s+1} = I^s \cup \{(a, s)\}$. Since we will keep this restraint active throughout the construction, it will be the case that either $|a|_s| = \omega$ or else it is finite and has a prime factor p such that either $p \notin Q$ or p^2 divides $|a|_s$.

Then let q_{s+1} be the least prime $p > q_s$ which does not violate any of the restraints $(t, b) \in I^{s+1}$. That is, q_{s+1} is the least q such that (i), (ii) or (iii) above are satisfied for all restraints in I^{s+1}.

(Case 2) Each class has $\nu(q_s)$ or fewer elements. In this case, $|a|_s \leq \nu(q_s)$ for all $a \leq k$ and each equivalence class is a cyclic subgroup of $\mathcal{A}_s$. Now, since $k = 1+\nu(q_s^2)$, there must be at least $\nu(q_s)$ different subgroups among the classes. Since there are no more than $\nu(q_s)$ possible orders (that is, numbers between 2 and $\nu(q_s)$) for these subgroups, there must be two distinct subgroups of the same order in $\mathcal{A}_s$. It follows from this that $\mathcal{A}_s$ has two distinct subgroups of some prime order and hence part (3) of the s-th requirement will be satisfied.

Then we again let q_{s+1} be the least $q > q_s$ which does not violate any of the restraints $(t, b) \in I^{s+1}$.

This completes the construction. The set $Q = \{q_0, q_1, \ldots\}$ is recursive since $q \in Q \iff (\exists s < q)(q = q_s)$. Let $M = Q \times \{0\}$.

Now suppose that $\eta(\mathcal{A}_s) = M$ for some s. Then $\mathcal{A}_s$ does not have two distinct subgroups of the same order so that Case 2 does not apply at stage $s+1$. Thus Case 1 must apply and hence there is an element a with finite order $|a|_s = q > \nu(q_s)$ such that (a, s) is in I^t for all $t > s$. Thus there must be a stage $t > s$ such that either condition (i) or (ii) is satisfied. That is, $|a|_s \leq \nu(q_t)$ and there is a prime factor $q < q_t$ of $|a|_s$ such that either

(i) $q \neq q_i$ for any $i \leq t$ or

(ii) $|a|_e$ is divisible by q^2.

In case (i), we have $(q, 0) \in \eta(\mathcal{A}_s)$ but $q \notin Q$. In case (ii), we have $(q^2, 0) \in \eta(\mathcal{A}_s)$ but $(q^2, 0) \notin M$. In either case, we see that $\eta(\mathcal{A}_s)$ is not a subset of M. □

Corollary 5.51 *There is a recursive torsion Abelian group $\mathcal{A}$ which is not isomorphic to any primitive recursive Abelian group with universe $Bin(\omega)$ or $Tal(\omega)$.*

For groups of unbounded order, Theorem 3.6 of [10] and Theorem 4.21 of [11] give different results.

Theorem 5.52 *[10]*

(a) There is a recursive Abelian group $\mathcal{A}$ which is not recursively isomorphic to any primitive recursive group.

(b) *There is an exponential-time Abelian group $\mathcal{B}$ which is not recursively isomorphic to any polynomial-time group.*

Theorem 5.53 *[11] There is a recursive torsion-free Abelian group which cannot be embedded into any p-time Abelian group.*

6 Uniqueness of Feasible Structures

In this section, we shall survey results on feasible categoricity. Again we shall concentrate mainly on polynomial time structures. As we shall see, unlike recursive model theory where there are many beautiful classification results on recursively categorical structures, there are very few examples of polynomial time categorical structures even if we restrict the universe. Thus most of the results on polynomial time categoricity are negative. Recall that a structure $\mathcal{A}$ with universe B is said to be *p-time categorical over* B if any structure $\mathcal{D}$ with universe B which is isomorphic to $\mathcal{A}$ is in fact p-time isomorphic to $\mathcal{A}$. A similar definition can be given for other notions of feasibility. We note that restricting the universe is crucial if we are to have any positive results due to the following general theorem of Cenzer and Remmel [14].

Theorem 6.1 *For any p-time relational structure $\mathcal{A} = (A, \{R_i^A\}_{i\in S})$, there are infinitely many p-time structures $\mathcal{B}_0 = \mathcal{A}$, $\mathcal{B}_1 = (B_1, \{R_i^1\}_{i\in S})$, $\mathcal{B}_2 = (B_2, \{R_i^2\}_{i\in S}), \ldots$ which are each recursively isomorphic to $\mathcal{A}$ and such that, for each $m < n$, there is a p-time map taking B_n one-to-one and onto B_m but there is no primitive recursive map from B_m into B_n which is at most c to 1, for some finite number c. Furthermore, the universes B_n may be taken to be subsets of $Tal(\omega)$ for each $n \geq 1$.*

Sketch of Proof: Let $B_0 = A$ and $\mathcal{B}_0 = \mathcal{A}$. Given $B_n = \{bin(b_0) < bin(b_1) < \ldots\}$, let $B_{n+1} = M(B_n) = \{bin(m_0) < bin(m_1) < \ldots\}$ as defined above in the proof of Lemma 4.9 and define the relations R_i on B_{n+1} to make the map taking $bin(b_e)$ to $bin(m_e)$ an isomorphism. □

6.1 Linear Orderings

In this subsection, we survey results of Remmel [69], which was the first paper on polynomial time categoricity. Remmel essentially showed that there are no polynomial categorical linear orderings over either $Tal(\omega)$ or $Bin(\omega)$.

The classic back-and-forth method of Cantor which shows that any two dense linear orderings without end points are isomorphic is crucial to the study of categoricity in linear orderings. The key step in defining an isomorphism between two structures requires a way to select, given two elements $a < b$ of one structures, an element $c < a$, an element $d > b$ and an element e with $a < e < b$. Thus we are led to the following effective notion of density functions in the effort of finding conditions which will provide some form of feasible categoricity.

Definition 6.2 *A Γ-computable dense linear ordering $L = (D, <)$ without end points is said to have Γ-computable density functions if there are Γ-computable functions f_a, f_b and f_i such that for any x and y in D, $f_b(x) < x < f_a(x)$ and $x < f_i(x, y) < y$.*

By carefully following the back-and-forth argument and keeping track of the number of steps required, we obtain the following, Theorems 3.1, 3.2 and 3.3 of [69].

Theorem 6.3 *Suppose $L_1 = (B, <_1)$ and $L_2 = (B, <_2)$ are polynomial-time dense linear orderings without endpoints with polynomial-time density functions. Then*

(a) *if $B = Tal(\omega)$, L_1 and L_2 are double-exponential-time isomorphic and*

(b) *if $B = Bin(\omega)$, L_1 and L_2 are triple-exponential-time isomorphic.*

Theorem 6.4 *Suppose $L_1 = (B, <_1)$ and $L_2 = (B, <_2)$ are polynomial-time dense linear orderings without endpoints with linear-time density functions. Then*

(a) *if $B = Tal(\omega)$, L_1 and L_2 are exponential-time isomorphic and*

(b) *if $B = Bin(\omega)$, L_1 and L_2 are double-exponential-time isomorphic.*

Theorem 6.5 *Suppose $L_1 = (Bin(\omega), <_1)$ and $L_2 = (Bin(\omega), <_2)$ are polynomial-time dense linear orderings without endpoints with quasi-real-time density functions. Then L_1 and L_2 are exponential-time isomorphic.*

Note that the standard ordering on the dyadic rationals in the interval $(0, 1)$ is in fact a p-time linear ordering with quasi-real density functions and has universe p-time isomorphic to $Bin(\omega)$. Details are given in Theorem 3.4 of [69]. On the other hand, there are p-time structures without nice density functions, as shown by Corollary 3.6 of [69].

Theorem 6.6 *There exist p-time dense linear ordering without end points with universe B for $B = Bin(\omega)$ and $B = Tal(\omega)$ which have no primitive recursive density functions.*

We note that there is a possible positive result, namely one can show that any two p-time linear orderings with universe $Tal(\omega)$ which have quasi-real-time density functions are polynomial time isomorphic. However Ash showed that there are no p-time linear orderings with universe $Tal(\omega)$ which have quasi-real-time density functions, see [69].

Examination of the previous theorems shows that the complexity of the back-and-forth isomorphism falls within the scope of exponential iteration. Thus we have the following.

Theorem 6.7 *Suppose $L_1 = (B, <_1)$ and $L_2 = (B, <_2)$ are polynomial-time dense linear orderings without endpoints with q-time density functions. Then for $B = Bin(\omega)$ or $B = Tal(\omega)$, L_1 and L_2 are q-time isomorphic.*

The main result of [69] improves Theorem 6.1 above by obtaining models with a fixed universe. This result shows that there really are no categorical linear orderings.

Theorem 6.8 *Let L be a p-time linear ordering with universe B, either $Tal(\omega)$ or $Bin(\omega)$. Then*

(a) *There exists a p-time linear ordering L' with universe B which is not primitive recursively isomorphic to L.*

(b) *If L is not recursively categorical, then there exists a p-time linear ordering L'' with universe B which is not recursively isomorphic to L.*

Sketch of Proof: We just sketch the proof of part (a). If L is recursively categorical, then L contains a copy of a dense linear ordering without end points. Then by Theorems 6.6 and 6.7, there exist p-time orderings L_1 and L_2 with universe B, one having p-time density functions and one without primitive recursive density functions. Thus L may not be primitive recursively isomorphic to both structures. $\Box$

6.2 Injection Structures

For injection structures, Cenzer and Remmel classified in Theorem 3.2 of [14] the recursively categorical injection structures.

Theorem 6.9 *A recursive injection structure (A, f) is recursively categorical if and only if it has only finitely many infinite orbits.*

The feasible categoricity results for injection structures depend on the spectrum of orbits. For example, there is one very nice positive result, Theorem 3.7 of [15].

Theorem 6.10 *Let $\mathcal{A} = (A, f)$ and $\mathcal{B} = (B, g)$ be two finitary permutation structures such that all but finitely many orbits have the same size q for some finite q.*

(a) *If $\mathcal{A}$ and $\mathcal{B}$ are both p-time over $Tal(\omega)$, then $\mathcal{A}$ is p-time isomorphic to $\mathcal{B}$.*

(b) *If $\mathcal{A}$ and $\mathcal{B}$ are both p-time over $Bin(\omega)$, then $\mathcal{A}$ is exponential time isomorphic to $\mathcal{B}$.*

(c) *If $\mathcal{A}$ and $\mathcal{B}$ are both q-time over either $Bin(\omega)$ or $Tal(\omega)$, then $\mathcal{A}$ is q-time isomorphic to $\mathcal{B}$.*

Sketch of Proof: We sketch the argument for $Tal(\omega)$. We may assume without loss of generality that all orbits have the same size q. The desired isomorphism ϕ is defined in stages ϕ^s, in which we enumerate s orbits $A_1, A_2, \ldots, A_s$ and $B_1, B_2, \ldots, B_s$ of each structure, by defining a sequence of elements $a_1, \ldots, a_s$ and $b_1, \ldots, b_s$ so that $A_i = \{a_i, f(a_i), \ldots, f^{q-1}(a_i)\}$ and similarly for B_i. Then we let $\phi_k(f^n(a_i)) = f^n(b_i)$. The key to measuring the complexity of this mapping is that since each orbit has q members, $a_k = tal(m)$ for some $m \leq kq$. $\square$

The general negative result is analogous to Theorem 6.8 above for linear orderings, except that we cannot specify the universe for the non-recursively categorical structures, since as seen by Theorem 6.10 there actually are some p-time categorical structures. Our next result combines Corollaries 3.3 and 3.5 of [14].

Theorem 6.11 *Let $\mathcal{A}$ be a p-time injection structure with universe B where B is either $Tal(\omega)$ or $Bin(\omega)$. Then*

(a) *There exists an infinite family $\mathcal{A}_i$ of p-time structures each recursively isomorphic to $\mathcal{A}$ which are pairwise not primitive recursively isomorphic.*

(b) *If $\mathcal{A}$ is not recursively categorical, then there exists a p-time structure $\mathcal{A}''$ with universe B which is not recursively isomorphic to $\mathcal{A}$.*

The most general result for recursively categorical structures is the following. This combines Theorems 3.6 and 3.10 of [14].

Theorem 6.12 *Let B be either $Bin(\omega)$ or $Tal(\omega)$ and let $\mathcal{A}$ be an injection structure such that either*

(a) *$\mathcal{A}$ has an infinite orbit or*

(b) *$\mathcal{A}$ has infinitely many orbits of size q for some finite q and has infinitely many other orbits.*

Then there is an infinite family $\mathcal{A}_i$ of p-time structures each with universe B and isomorphic to $\mathcal{A}$ which are pairwise not primitive recursively isomorphic.

Sketch of Proof: There are two distinct arguments. We first sketch the proof in the case that $\mathcal{A}$ has either an infinite orbit or infinitely many orbits of finite size q, together with an infinite set of other elements. We partition the structure into two parts. The first part $\mathcal{B}$ is either the infinite orbit or the infinitely many orbits of size q and may be assumed to have universe B by Theorem 5.31 and 5.32. The second part $\mathcal{C}$ has an infinite family of copies $\mathcal{C}_i$ with universe C_i such that B cannot be primitive recursively embedded in any C_i and such that, by Theorem 6.11, and for any $i \neq j$, $\mathcal{C}_i$ is not primitive recursively isomorphic to $\mathcal{C}_j$. Now just let $\mathcal{A}_i = \mathcal{B} \oplus \mathcal{C}_i$.

For the case of a single infinite orbit, we appeal directly to Lemma 4.9. Here is the construction of a copy of (ω, S) with universe $Bin(\omega)$ but not primitive recursively isomorphic to the standard structure. Let $m_0 < m_1 < \ldots$ be the set from Lemma 4.9, where $A = Bin(\omega)$, and assume $m_0 = 0$.

We define $0, f^B(0), f^B(f^B(0)), \ldots$ in blocks so that the k-th block is in three parts:

$$3m_k, 3m_k + 3, \ldots, 3m_{k+1} - 3,$$

followed by

$$3m_{k+1} - 2, 3m_{k+1} - 5, \ldots, 3m_k + 1,$$

and then

$$3m_k + 2, 3m_k + 5, \ldots, 3m_{k+1} - 1.$$

The unique isomorphism φ from (ω, S) to (B, f^B) maps $3m_k + 1$ to $2m_{k+1} - m_k - 1$ and is not primitive recursive by Lemma 4.9. $\square$

Finally, we note that if $\mathcal{A}$ has no infinite orbits and the spectrum of $\mathcal{A}$ is p-time in tally as in Theorem 5.34, then the conclusion of Theorem 6.12 also applies by Theorem 3.8 of [14].

6.3 Models of Arithmetic

Before focusing on the categoricity of torsion Abelian groups, we briefly present two results for the group $\mathbb{Z}$ of integers. These are Theorems 4.28, 4.29 and 4.30 of [14].

Theorem 6.13 *Let B be $Tal(\omega)$ or $Bin(\omega)$. There is a p-time structure $(B, S^B, +^B)$ isomorphic to $(\mathbb{Z}, S, +)$ but not exponential time isomorphic.*

Sketch of Proof: (Binary case) Let $< 0, bin(n) >$ represent $n \geq 0$ and let $< 1, bin(2^{n^2}) >$ represent $-n < 0$. $\square$

Theorem 6.14 *There is a fully p-time group $\mathcal{A}$ isomorphic to $\mathbb{Z}$ but not q-time isomorphic.*

Sketch of Proof: This is a corollary of 5.30, since the model defined there is not q-time isomorphic to $\mathcal{N}$. To see this, observe that the term $E^n 0$, which has length $n+1$ is mapped to the iterated exponential $2^{2^{\cdots}}$. $\square$

Let $Bin(\mathbb{Z})$ be the standard structure of $\mathbb{Z}$ with universe p-time isomorphic to $Bin(\omega)$ and similarly for $Tal(\mathbb{Z})$.

Theorem 6.15 *There is an EXPTIME (respectively, exponential-time) group $\mathcal{A}$ with universe $Bin(\omega)$ $(Tal(\omega))$ which is isomorphic to $Bin(\mathbb{Z})$ $(Tal(\mathbb{Z}))$ but not q-time isomorphic.*

6.4 Torsion Abelian Groups

The results for Abelian groups are parallel to those given above for injection structures. We begin with the positive results, Theorems 4.24 and 4.25 of [14].

Theorem 6.16 *Let p be a prime, and let $\mathcal{A}$ and $\mathcal{B}$ be two groups with universe B, where $B = Tal(\omega)$ or $B = Bin(\omega)$, both isomorphic to $\oplus_{n<\omega}\mathbb{Z}(p)$.*

(a) If $\mathcal{A}$ and $\mathcal{B}$ are p-time, then $\mathcal{A}$ and $\mathcal{B}$ are EXPTIME isomorphic if $B = Tal(\omega)$ and double-exponential-time isomorphic if $B = Bin(\omega)$.

(b) If $\mathcal{A}$ and $\mathcal{B}$ are q-time, then $\mathcal{A}$ and $\mathcal{B}$ are q-time isomorphic.

Sketch of Proof: We sketch the proof of (a) for universe $Tal(\omega)$. The standard structure $\mathcal{B}$ may be viewed as an infinite dimensional vector space over $\mathbb{Z}(p)$, where the general element $(c_1, \ldots, c_n)$ is represented by $tal(c_1 + c_2 \cdot p + \ldots + c_n \cdot p^{n-1})$. The arbitrary structure $\mathcal{A}$ will have a basis defined recursively by letting a_n be the least element independent of $\{a_1, \ldots, a_{n-1}\}$. It can then be seen that the map taking $tal(c_1 + c_2 \cdot p + \ldots + c_n \cdot p^{n-1})$ to $c_1 \cdot a_1 + \ldots + c_n \cdot a_n$ is exponential time and its inverse is EXPTIME. □

The next result, Theorem 4.26 of [14] shows that a p-time isomorphism is not always possible in Theorem 6.16.

Theorem 6.17 *For any prime p and for $B = Tal(\omega)$ or $B = Bin(\omega)$, there exist two p-time groups with universe B and which are isomorphic to $\oplus_{i<\omega}\mathbb{Z}(p)$ but which are not p-time isomorphic to each other.*

The case of $\oplus_\omega \mathbb{Z}(p^m)$ where $m > 1$ requires a stronger hypothesis. The difficulty is that only elements not divisible by p, can be used for the generators $a_1, a_2, \ldots$ and these may all be very large. (This is the basis for the proof of Theorem 6.17 above.) What is needed is the ability to compute a divisor of an element x which is divisible by p. Let us say that the group $\mathcal{A}$ has *recursive divisors* if there is an algorithm which, for any $a \in \mathcal{A}$, determines whether a is divisible and which computes a divisor of a if there is one; if the algorithm runs in polynomial time, then we say that $\mathcal{A}$ has *p-time divisors.* Note that the standard models of the recursively categorical groups all have p-time divisors.

Theorem 6.18 *Let p be a prime, let $m > 1$ be finite and let $\mathcal{A}$ and $\mathcal{B}$ be two groups with universe B, $B = Tal(\omega)$ or $B = Bin(\omega)$, both isomorphic to $\oplus_\omega \mathbb{Z}(p^m)$.*

(a) If $\mathcal{A}$ and $\mathcal{B}$ are p-time and have p-time divisors, then $\mathcal{A}$ and $\mathcal{B}$ are EXPTIME isomorphic if $B = Tal(\omega)$ and double-exponential-time isomorphic if $B = Bin(\omega)$.

(b) If $\mathcal{A}$ and $\mathcal{B}$ are q-time and have q-time divisors, then $\mathcal{A}$ and $\mathcal{B}$ are q-time isomorphic.

The next result, Theorem 4.19 of [14] shows that the hypothesis of p-time divisibility is needed in Theorem 6.18.

Theorem 6.19 *Let B be either $Bin(\omega)$ or $Tal(\omega)$, let p be a prime and let $m > 1$ be finite. Then there exists an infinite family of p-time groups $\mathcal{G}_i$ each recursively isomorphic to $\oplus_{i<\omega}\mathbb{Z}(p^m)$ and having universe B such that there is no primitive recursive structure preserving embedding from $\mathcal{G}_i$ into $\mathcal{G}_j$ for any $i < j$.*

The basic non-categoricity result for torsion groups is Theorem 4.11 of [14].

Theorem 6.20 *For any infinite recursive Abelian torsion group $\mathcal{A}$, there is an infinite family $\mathcal{A}_i$ of p-time groups each recursively isomorphic to $\mathcal{A}$ and having universe a subset of $Tal(\omega)$ which are pairwise not primitive recursively isomorphic.*

It is also the case that if some p-primary component of $\mathcal{A}$ is infinite and has bounded order, or is isomorphic to $Z(p^\infty)$, then each $\mathcal{A}_i$ in Theorem 6.20 may be taken to have standard universe.

Next we give two results for p-groups, Theorems 4.9 and 4.23 of [14]. The first is the fundamental result for non-recursively categorical p-groups and the second is a summary of results for products of basic p-groups.

Theorem 6.21 *Let $\mathcal{G}$ be a recursive p-group which is not recursively categorical. Then there exist p-time groups $\mathcal{H}_1$ and $\mathcal{H}_2$ both isomorphic to $\mathcal{G}$ but not recursively isomorphic to each other. If $\mathcal{G}$ has bounded order, then we may take $\mathcal{H}_1$ and $\mathcal{H}_2$ to have universe B where either $B = Tal(\omega)$ or $B = Bin(\omega)$.*

Theorem 6.22 *Let p be a prime number, let $B = Tal(\omega)$ or $Bin(\omega)$, and let $\mathcal{C}$ be an infinite recursive group which is a product of cyclic and quasi-cyclic p-groups and which is not isomorphic to $(\oplus_{i<\omega}\mathbb{Z}(p)) \oplus \mathcal{F}$ for any finite group $\mathcal{F}$ and either $\mathcal{C}$ has a quasicyclic factor or is a product of cyclic groups such that $\eta(\mathcal{C})$ is p-time in tally.*

- *(a) Then there exists an infinite family $\mathcal{A}_i$ of p-time groups with universe B and isomorphic to $\mathcal{C}$ which are pairwise not primitive recursively isomorphic.*
- *(b) If $\mathcal{C}$ is not recursively categorical, then there exist p-time groups $\mathcal{A}_1$ and $\mathcal{A}_2$, each with universe B and isomorphic to $\mathcal{C}$, which are not recursively isomorphic to each other.*

The group $\mathbb{Q}$ of rationals is closely related to the quasicyclic groups $\mathbb{Z}(p^\infty)$, since $\mathbb{Q} \cap [0, 1]$ is isomorphic to the product of the quasicyclic groups. We use this to obtain the following result, Theorem 4.31 of [14].

Theorem 6.23 *Let B be either $Bin(\omega)$ or $Tal(\omega)$. Then there is an infinite family of p-time groups $\mathcal{H}_i$ each with universe B and isomorphic to $\mathbb{Q}$ but not pairwise primitive recursively isomorphic.*

6.5 Scott Families

We now consider some general, syntactic conditions which lead to some feasible categoricity results. Nurtazin [60] and Goncharov [29] provided sufficient conditions to ensure that a model $\mathcal{A}$ with universe A is recursively categorical, namely if there is a finite sequence $(c_0, \ldots, c_{k-1})$ of elements of A and a recursive sequence (called a *Scott family*) of recursive existential formulas $\{\phi_n(x_1, \ldots, x_m, c_0, \ldots, c_{k-1}) : n < \omega\}$ in the extended language with names for $c_0, \ldots, c_{k-1}$ satisfying the following two conditions:

(1) Every $a_1, \ldots, a_m \in A$ satisfies one of the formulas ϕ_n;

(2) For each n and for any $(a_1, \ldots, a_m)$ and $(d_1, \ldots, d_m)$, if $\mathcal{A}$ satisfies $\phi_n(a_1, \ldots, a_m, c_0, \ldots, c_{k-1})$ and $\phi_n(d_1, \ldots, d_m, c_0, \ldots, c_{k-1})$, then $(A, a_1, \ldots, a_m, c_0, \ldots, c_{k-1})$ is isomorphic to $(A, d_1, \ldots, d_m, c_0, \ldots, c_{k-1})$ via the map which sends a_i to d_i for $i = 1$ to m and c_i to c_i for $i < k$.

Several notions of feasible Scott families were developed in [15] and applied to the feasible structures we have studied. We will present one such formulation here.

A Scott family $\{\phi_n(x_1, \ldots, x_m, c_0, c_1, \ldots, c_{k-1}) : n < \omega\}$ of p-time existential formulas, for a p-time model $\mathcal{A}$ with universe A, satisfying (1) and (2) as described above is said to be *strongly p-time* if there is some fixed integer $r > 1$ such that the following conditions are satisfied, for each $m \geq 0$.

(3) For any finite sequence $a_1, \ldots, a_m$ of elements of $\mathcal{A}$, we can compute in time $\leq (max\{2, m, |a_1|, \ldots, |a_m|\})^r$ a formula ϕ_t from the list such that $\phi_t(a_1, \ldots, a_m, c_0, c_1, \ldots, c_{k-1})$ holds in $\mathcal{A}$.

(4) For each formula $\phi_t(x_1, \ldots, x_m, c_0, \ldots, c_{k-1})$ and each $a_1, \ldots, a_m \in A$, if there exists a such that $\mathcal{A}$ satisfies $\phi_t(a_1, \ldots, a_m, a, c_0, c_1, \ldots, c_{k-1})$, then there exists such an a with $|a| \leq (m+2)^r + max\{|a_1|, \ldots, |a_m|\}$.

(5) For each $\phi_t(x_1, \ldots, x_m, c_0, c_1, \ldots, c_{k-1})$ and each $a_1, \ldots, a_m \in A$, if there exists a such that $\mathcal{A}$ satisfies $\phi_t(a_1, \ldots, a_m, a, c_0, c_1, \ldots, c_{k-1})$, then we can compute an a as described in (4) in time $\leq (max\{2, m, |a_1|, \ldots, |a_m|\})^r$.

Note that clause (4) above implies that the structure $\mathcal{A}$ has only finitely many types of each arity. The following theorem is proved by a careful analysis of the back-and-forth method.

Theorem 6.24 *If $\mathcal{A}$ and $\mathcal{B}$ possess a common strongly p-time Scott family, then $\mathcal{A}$ and $\mathcal{B}$ are p-time isomorphic if both have universe $Tal(\omega)$ and are exponential time isomorphic if both have universe $Bin(\omega)$.*

Theorem 6.5 can be proved directly from this general result. We give one other corollary here which provides some additional feasible categoricity for permutation structures.

Corollary 6.25 *Let $\mathcal{A} = (Tal(\omega), f)$ and $\mathcal{B} = (Tal(\omega), g)$ be two isomorphic p-time permutation structures such that for some fixed integer k,*

(i) *for any a and a' in the same orbit,*

$$|a'| \leq |a| + k$$

and

(ii) *for any $a_0, a_1, \ldots, a_{m-1} \in B$ and any finite q, if there is an orbit of size q not containing any of $a_0, \ldots, a_{m-1}$, then there is such an orbit containing an element a of size*

$$|a| \leq max\{|a_0|, \ldots, |a_{m-1}|\} + (m+2)^k.$$

Then $\mathcal{A}$ and $\mathcal{B}$ are p-time isomorphic.

Weaker notions of Scott families defined in [15] include the *strongly exponential time Scott family*, which leads to exponential time isomorphism for universe $Tal(\omega)$ and double exponential time isomorphism for universe $Bin(\omega)$ and the *strongly EXPTIME Scott family,*which leads to EXPTIME isomorphism for universe $Tal(\omega)$ and double exponential time isomorphism for universe $Bin(\omega)$. The following applications are given in [15].

Corollary 6.26 *Let $\mathcal{A} = (B, \equiv^A)$ and $\mathcal{B} = (B, \equiv^B)$ be two polynomial time models of an equivalence relation $\equiv$ such that, for some fixed integer k, both models satisfy the following:*

(i) *for any a and a' in the same equivalence class,*

$$|a'| \leq k \cdot |a| \text{ if } B = Tal(\omega)$$

or (where $a = bin(n)$ and $a' = bin(n')$)

$$n - k|a| \leq n' \leq n + k|a| \text{ if } B = Bin(\omega)$$

and

(ii) *for any $a_0, \ldots, a_{m-1} \in B$ and any finite q, if there is an equivalence class of size q not containing any of $a_0, \ldots, a_{m-1}$, then there is such a class containing an element b of size*

$$|b| \leq k \cdot max\{k^m, |a_0|, \ldots, |a_{m-1}|\} \text{ if } B = Tal(\omega)$$

or

$$|b| \leq k \cdot max\{2, m\} \text{ if } B = Bin(\omega).$$

Then $\mathcal{A}$ and $\mathcal{B}$ are exponential time isomorphic if $B = Tal(\omega)$, and double exponential time isomorphic if $B = Bin(\omega)$.

Corollary 6.27 *Let $\mathcal{A}$ and $\mathcal{B}$ be two isomorphic p-time torsion Abelian groups with the same universe $Tal(\omega)$ such that for some fixed integer k:*

(i) *for any a, b,*

$$|a + b| \leq k \cdot max\{|a|, |b|\}$$

and

(ii) *for any $a_0, \ldots, a_{m-1}$ in either $\mathcal{A}$ or $\mathcal{B}$ and any finite q, if there is an element of order q not in $G(a_0, \ldots, a_{m-1})$ (that is, the subgroup generated by $\{a_0, a_1, \ldots, a_{m-1}\}$), then there is such an element b of size*

$$|b| \leq k^m \cdot max\{|a_0|, \ldots, |a_{m-1}|\}.$$

Then $\mathcal{A}$ and $\mathcal{B}$ are EXPTIME isomorphic if $B = Tal(\omega)$ and are double exponential time isomorphic if $B = Bin(\omega)$.

7 Complexity Theoretic Algebra

In this section, we introduce the second theme of our survey. That is, instead of focusing on problems of comparing polynomial time versus recursive models, we will fix a given polynomial time model such as an infinite dimensional vector space over a polynomial time field or a polynomial time atomless Boolean algebra and consider the internal structure of that model. Once again we shall use established results from recursive algebra as a guide.

In recursive algebra, one studies the effective content of results like the fact that every independent subset of a vector space V can be extended to a basis. If the vector space V is infinite dimensional, then all known proofs of this fact use some version of the axiom of choice, e.g. Zorn's Lemma, which is known to be non-constructive. Thus one would expect that it is not the case that every recursive independent set can be extended to a recursive basis in infinite dimensional recursive vector space. Indeed, Metakides and Nerode [48] proved that not every recursive independent set of a recursively presented infinite dimensional vector space over a recursive field could be extended to a recursive basis. Another theme in the study of recursive algebra has been to study the lattice of r.e. substructures of various recursive structures structures; see the survey article by Nerode and Remmel [51]. Nerode and Remmel began the study of complexity theoretic algebra in a series of papers, [53], [55], [54], and [58]. We survey their results as well as results by Bäuerle [5] in the next two sections.

The overriding paradigm of Nerode and Remmel's study of complexity theoretic algebra was to use the admittedly flawed analogy that "recursive is to r.e." as "P is to NP" to formulate natural complexity theoretic analogues of theorems in recursive algebra. For example, Dekker [24] proved that every r.e. subspace of a recursively presented infinite dimensional vector space over a recursive field with a dependence algorithm has a recursive basis. The natural complexity theoretic analogue of Dekker's Theorem is that in a suitable polynomial time infinite dimensional vector space V over a polynomial time field with a polynomial time dependence algorithm, every NP subspace of V has a basis in P. It turns out that the proof of Dekker's Theorem is not uniform in that the proof breaks up into two cases depending on whether the underlying field of V is finite or infinite. The complexity theoretic analogue of Dekker's Theorem behaves very differently in these two cases. That is, Nerode and Remmel [55] proved that if the underlying field is infinite and has a polynomial time representation with certain nice properties, then every NP subspace of V has a basis in P. However if the underlying field is finite, then Dekker's Theorem is oracle dependent. That is, there is an oracle X such that $P^X \neq NP^X$ and every NP^X subspace of V has a basis in P^X and there is an oracle Y such that $P^Y \neq NP^Y$ and there is a subspace W of V which is NP^Y but has no basis in P^Y. This presents us with two general themes. Sometimes the complexity theoretic analogue of a theorem of recursive algebra is true but must be proved by more delicate methods which take into account the bounds of the resources used in a computation. Sometimes the complexity theoretic analogue is false or oracle dependent because the proof of the recursive algebra result uses unbounded resources available in a recursive construction in a crucial way. Thus complexity theoretic algebra is not just a mere translation of the results of recursion theoretic algebra.

Another problem that complicates the study of complexity theoretic algebra is that fact that not all polynomial time models are equivalent, as we have seen in the previous sections. That is, Metakides and Nerode showed that all infinite dimensional recursive vector spaces with an effective dependence algorithm are recursively isomorphic. Similarly, Cantor's basic back and forth argument which shows that all countable free Boolean algebras are isomorphic is effective so that all recursive free Boolean algebras are recursively isomorphic. As we have seen in the previous section it is certainly not the case that all polynomial time free Boolean algebras are polynomial time isomorphic. Thus in complexity theoretic algebra, one fixes a polynomial time presented structure over a natural universe such as the tally representation of the natural numbers or the binary representation of the natural numbers and studies that particular structure. Indeed, Nerode and Remmel studied two basic models of vector spaces, the *tally representation* of an infinite dimensional vector space of a polynomial time field with a polynomial time dependence algorithm, $Tal(V_\infty)$, where the underlying universe is the tally representation of the natural numbers and the *binary representation* of an infinite dimensional vector space of a polynomial time field with a polynomial time dependence algorithm, $Bin(V_\infty)$, where the underlying universe is the binary representation of the natural numbers. Similarly they consider a tally representation and a binary representation of the free Boolean algebra. The results for the tally representation and standard representation of a structure are not always the same.

Another basic question in the study of complexity theoretic algebra is whether the priority method which was so useful in the study of recursive algebra would again play a central role. In 1975, Metakides and Nerode [47] initiated the systematic study of recursion theoretic algebra and introduced the use of the finite injury priority method from recursion theory as a uniform tool to meet algebraic requirements. Prior to that time the priority method has been limited primarily to internal applications within recursion theory in the theory of recursively enumerable sets and in the theory of degrees of unsolvability and their generalizations. Recursion theoretic algebra has been developed since, in depth, by many authors in such subjects as commutative fields, vector spaces, orderings, and Boolean algebras (see Crossley [21] for references and a cross-section of results before 1980). Recursion theoretic algebra yielded as a byproduct a theory of recursively enumerable substructures (see the survey article Nerode-Remmel [51] for references).

Simultaneously in computer science there was a vast development of P and NP problems in complexity theory. This subject started out as a tool for measuring the relative difficulties of classes of computational problems (see Cobham [19], Cook [20], Hartmanis and Stearns [35]). Many papers in this area have dealt with coding a given problem M into a calibrated problem to find an upper bound on the complexity of M, and coding a calibrated problem into a given problem M to find a lower bound the complexity of M (see Hopcroft and Ullman [37] and Garey and Johnson [28]). Due to the intractability of the fundamental problem $P = NP$, Baker-Gill-Solovay [4] began a line of inquiry using diagonal arguments to produce sets ("oracles") R_1, R_2 such that $P^{R_1} = NP^{R_1}, P^{R_2} \neq NP^{R_2}$. Typical of recent work in this direction is the construction by Yao [76] of oracles relative to which none of the polynomial time hierarchy collapses, and the result of Cai [8] that this holds for oracles with probability 1. The Baker-Gill-Solovay, Yao, and Cai results are fundamental, but they do not use the priority method which was used systematically with success in recursion-theoretic algebra.

Priority arguments have been used by many authors in the study of P^A and NP^A sets for recursive or recursively enumerable oracles A. For example, Homer and Maass [36], used priority arguments to investigate the lattice of NP^A sets. Shinota and Slaman [72] and Shore and Slaman [73] have used priority argument to study the structure of the polynomial time Turing degrees relative to a recursive oracle. Downey and Fellows [25] used priority arguments to study the density of their fixed parameter complexity classes. Nerode and Remmel ([53], [55], [54], and [58]) showed that indeed priority methods play a central role in the study of complexity theoretic algebras as we will bring out in the following sections.

We will start by surveying results of Nerode, Remmel and Bäuerle on polynomial time vector spaces.

8 Polynomial Time Vector Spaces

In this section, we shall study the structure of an infinite dimensional vector space V_∞ over a polynomial time field. We will start by giving some basic definitions and defining the binary or standard representation of V_∞ and the tally representation of V_∞. Our definitions of the standard and tally representation of V_∞ will be broken down into two cases depending on whether the underlying field F is finite or infinite.

A recursive field $F = \langle U_F, +_F, \cdot_F, AI_f, MI_F \rangle$ consists of a recursive subset U_F of the natural numbers ω and partial recursive functions $+_F$ (field addition), $\cdot_F$ (field multiplication), AI_F (field additive inverse), and MI_F (field multiplicative inverse) such that these operations restricted to F turn U_F into a field. A *recursively presented vector space* $V = \langle U_V, +_V, \cdot_V \rangle$ consists of a recursive subset U_V of the natural numbers and partial recursive functions $+_V$ (vector space addition) and $\cdot_V : U_F \times U_V \to U_V$ (scalar multiplication) which turn U_V into a vector space. V is said to have a *dependence algorithm* if there is a uniform effective procedure which given any n-tuple $v_0, \ldots, v_{n-1}$ will determine if $v_0, \ldots, v_{n-1}$ are dependent.

We say that a recursive field $F = \langle U_F, +_F, \cdot_F, AI_f, MI_F \rangle$ is a *polynomial time field* if U_F is a polynomial time subset of $\{0,1\}^*$ and the operations $+_F, \cdot_F, AI_F, MI_F$ are the restrictions of total polynomial time functions. We will always assume that $0, 1 \in U_F$ and that 0 is the zero of F and that 1 is the multiplicative identity of F.

Let V_∞ be the infinite dimensional vector space over a polynomial time field F which consists of all finite sequences $< a_1, \ldots a_n >$ of elements of F where $a_n \neq 0$ together with the empty sequence $\emptyset$ which is the zero of the vector space. The operations on V_∞ are induced by coordinate-wise addition and scalar multiplication. Finally we say that a vector $v =< a_1, \ldots a_n >$ of V_∞ where $a_i \in F$ for $1 \leq i \leq n$ and $a_n \neq 0$ has *height* n. We say that the zero vector of V_∞ has height 0.

Case 1 F is finite.

Suppose that $F = \{0, 1, \ldots, k-1\}$ is a finite field where 0 is zero of F and 1 is the multiplicative identity of F. The space V_∞ can be coded into the natural numbers $\omega = \{0, 1, 2, \ldots\}$ as a polynomial time vector space in many ways. We refer to $e_1, e_2, \ldots$ as the standard basis of V_∞ where e_n is the sequence of the length n, $\langle 0, \ldots, 0, 1 \rangle$ with $n-1$ zeros and 1 denotes the unit of F. Now the question of whether V_∞ is polynomial time, recursive, etc., depends on how we code the sequences $\langle a_1, \ldots, a_n \rangle$. Following [53, 55], we will distinguish two specific polynomial time representations of V_∞ which we call the *tally* and *binary* (or *standard*) representations of V_∞. We identify each vector $v \in V_\infty$ with a natural number $R(v)$ by $R(\vec{0}) = 0$ and

$$R(\langle a_1, \dots, a_n \rangle) = a_1 + a_2 k + \dots a_n k^{n-1} \text{ if } a_n \neq 0.$$

Next, with a slight abuse of notation, we define maps $b_k : V_\infty \to B_k(\omega)$, $bin : V_\infty \to Bin(\omega)$ and $tal : V_\infty \to Tal(\omega)$ by $b_k(v) = b_k(R(v))$, $bin(v) = bin(R(v))$ and $tal(v) = tal(R(v))$.

Then $B_k(V_\infty)$ consists of the set $B_k(\omega)$ with the operations of vector addition $+_{B_k}$ and scalar multiplication $\cdot_{B_k}$ induced by the corresponding operations from V_∞. Similarly, $bin(V_\infty)$ consists of the set $Bin(\omega) = \{bin(v) : v \in V_\infty\}$ with corresponding induced operations $+_{bin}$ and $\cdot_{bin}$ and $tal(V_\infty)$ consists of the set $Tal(\omega)$ with the induced operations $+_{tal}$ and scalar multiplication $\cdot_{tal}$. It is easy to see that $B_k(V_\infty)$, $bin(V_\infty)$ and $tal(V_\infty)$ are polynomial time structures and it follows from Lemma 4.4 that $B_k(V_\infty)$ and $bin(V_\infty)$ are p-time isomorphic. We shall normally refer to either of these two structures as the *standard* representation $st(V_\infty)$ of V_∞ and write the operations as $+_{st}$ and $\cdot_{st}$.

Case 2 F is infinite.

Recall the p-time coding functions $\langle \sigma_1, \dots, \sigma_k \rangle_k$ defined in section 4. Now suppose that $F = \langle U_F, +_F, \cdot_F, AI_f, MI_F \rangle$ is an infinite polynomial time field of characteristic 0. Let $\mathbf{0}$ and $\mathbf{1}$ denote the zero and 1 of F respectively. For any positive integer n, let $\mathbf{n} = \mathbf{1} + \dots + \mathbf{1}$ where there are n summands and let $-\mathbf{n} = AI_F(\mathbf{n})$. For any integers n and $m \neq 0$, let $\mathbf{n}/\mathbf{m} = \mathbf{n} \cdot_F MI(\mathbf{m})$. Then set

$$Q^+ = \{\mathbf{n}/\mathbf{m} : n \in N, m \in N \setminus \{0\}\}.$$

Thus Q^+ is a copy of the nonnegative rationals inside of F. We say that Q is *properly embedded* in F if

(i) Q^+ is a polynomial time subset of $\{0,1\}^*$ and

(ii) the map $f : Q^+ \to \{0,1\}^*$ given by $f(\mathbf{n}/\mathbf{m}) = [bin(n), bin(m)] = bin([n, m])$ is the restriction of polynomial time function from $\{0,1\}^*$ to $\{0,1\}^*$.

Now suppose that $F = \langle U_F, +_F, \cdot_F, AI_f, MI_F \rangle$ is a polynomial time field where Q^+ is properly embedded and $U_F = \{0,1\}^*$. Define $bin : V_\infty \to Bin(\omega)$ by $bin(\vec{0}) = 0$ and

$$bin(< a_1, \dots a_n >) = \langle a_1, \dots a_n \rangle_n \text{ for } a_1, \dots, a_n \text{ in } F \text{ with } a_n \neq 0.$$

In this case, we let $st(V_\infty) = (U_b, +_b, \cdot_b)$, where $U_b = \{bin(v) : v \in V_\infty\}$ and where the operations $+_b$ and $\cdot_b$ are defined so that bin is an isomorphism from V_∞ onto $st(V_\infty)$. It is easy to see that the operations $+_b$ and $\cdot_b$ are the restrictions of polynomial time functions and that U_b is polynomial time isomorphic to $\{0,1\}^*$. We call $st(V_\infty)$ the *binary representation* (or the *standard representation*) of V_∞ in this case.

The tally representation of V_∞ is defined by observing that that if $\sigma = \sigma_0 \cdots \sigma_n$ is any string of U_{st} other than the empty string, then σ ends in a 1. Hence there is an integer n_σ such that $bin(n_\sigma) = \sigma_n \cdots \sigma_1$.

Now define a map $tal : V_\infty \to Tal(\omega)$ by $tal(v) = tal(n_v)$, where n_v is the natural number such that $bin(v) = bin(n)$ and let $tal(V_\infty) = (U_t, +_t, \cdot_t)$, where $U_t = \{tal(v) : v \in V_\infty\}$ and the operations $+_t$ and $\cdot_t$ are defined so that tal is an isomorphism from V_∞ onto $tal(V_\infty)$. It is easy to see that the operations $+_t$ and $\cdot_t$ are the restrictions of polynomial time functions and that U_t is polynomial time isomorphic to $Tal(\omega)$. We call $tal(V_\infty)$ the *tally representation* of V_∞ in this case.

Finally we argue that both the standard and tally representation of V_∞ have polynomial time dependence algorithms. First the decoding functions π_i^k defined in section 4 allow us to recover the coefficients $a_1, \ldots, a_k$ from any vector $bin(v) = \langle a_1, \ldots, a_k \rangle_k \in st(V_\infty)$. We can then similarly recover the coefficients from $tal(v)$ by first computing $bin(v)$. It follows that, given any set $v_1, \ldots v_n$ of vectors in either of our representations of V_∞, we can recover the matrix of coefficients of $v_1, \ldots, v_n$ corresponding to the expansions of those vectors in terms of the standard basis $e_1, e_2, \ldots$ of V_∞ in polynomial time in the sums of the lengths $|v_1| + \cdots + |v_n|$. We can then use Gaussian elimination on the matrix of coefficients to determine whether or not $\{v_1, \ldots, v_n\}$ is an independent set. Since Gauss elimination is polynomial time over the coefficients (since the operations of F are polynomial time), it follows that in each of our representations, there is a polynomial p such that we can decide if $\{v_1, \ldots, v_n\}$ is dependent in $p(|v_1| + \cdots + |v_n|)$ steps.

We end this section with some basic definitions and notations for vector spaces. Let V be either V_∞, $st(V_\infty)$ or $tal(V_\infty)$. We shall abuse notation and let $\vec{0}$ denote the zero vector for V_∞, $st(V_\infty)$, and $tal(V_\infty)$ even though technically the zero vectors of the three vector spaces are distinct objects. Given a subset A of V, we let $space(A)$ denote the subspace of V generated by A. Given two subspaces U and W of V, we let $U + W$ denote the subspace generated by $U \cup W$. We shall write $W = U_1 \oplus U_2$ if W, U_1 and U_2 are subspaces of V such that $W = U_1 + U_2$ and $U_1 \cap U_2 = \{\vec{0}\}$. We say U is a complementary subspace of W if $U \oplus W = V$. Given $x \in V$, we let $ht(x)$ denote the height of x. We note that if $x \in st(V_\infty)$, then in polynomial time in $|x|$, we can produce the binary representations of the integers $a_1, \ldots, a_n$ such that $x = bin(\langle a_1, \ldots, a_n \rangle)$ with $a_n \neq 0$ so that we can find the height of x in polynomial time in $|x|$. Similarly if $x \in tal(V_\infty)$, then in polynomial time in $|x|$, we can produce the tally representations of the integers $a_1, \ldots, a_n$ such that $x = tal(\langle a_1, \ldots, a_n \rangle)$ with $a_n \neq 0$ so that we can find the height of x in polynomial time in $|x|$.

8.1 Subspaces and Bases over infinite polynomial time fields

We shall see that there is a vast difference between the theory of bases and subspaces of $st(V_\infty)$ or $tal(V_\infty)$ when the underlying field is infinite as opposed to when the underlying field is finite. For example, Nerode and Remmel proved the following strengthening of Dekker's Theorem that every r.e. subspace of a recursively presented vector space over a recursive field with a dependence algorithm has a recursive basis.

Theorem 8.1 *([55])*
Let F be a polynomial time field where Q is properly embedded. Then

(a) *every r.e. subspace V of $tal(V_\infty)$ has a basis in P and*

(b) *every r.e. subspace W of $st(V_\infty)$ has a basis in P.*

Bäuerle [5] proved the existence of simple and maximal subspaces of $tal(V_\infty)$ which are in P. To properly state Bäuerle's results, we first need some definitions.

In the lattice, $\mathcal{E}$, of recursively enumerable (r.e.) sets of natural numbers, a r.e. set S is *simple* if $\omega \setminus S$ is infinite and for any infinite r.e. set W, $W \cap S \neq \emptyset$. A r.e. set M is *maximal* if $\omega \setminus M$ is infinite and for any r.e. set $W \supseteq M$, either $\omega \setminus W$ or $W \setminus M$ is finite. The analogues of these notions in the lattice of NP^A sets, $\mathcal{E}_{NP^A}$, for any oracle A are the following. A NP^A set $S \subseteq \{0,1\}^*$ is NP^A-*simple* if $\{0,1\}^* \setminus S$ is infinite and for any infinite NP^A set $W \subseteq \{0,1\}^*$, $W \cap S \neq \emptyset$. A NP^A set $M \subseteq \{0,1\}^*$ is NP^A-*maximal* if $\{0,1\}^* \setminus M$ is infinite and for any NP^A set $W \supseteq M$, either $\{0,1\}^* \setminus W$ or $W \setminus M$ is finite.

It was shown by Homer and Maass [36], that there exists oracles A and B such that $NP^A \neq P^A$ and no NP^A-simple sets exist and $NP^B \neq P^B$ and there exist NP^B-simple sets. It follows from a result of Briedbart [7] that there are no NP^A-maximal sets for any A.

In the lattice, $\mathcal{L}(V_\infty)$, of r.e. subspaces of a recursively presented copy of V_∞, a r.e. subspace S of V_∞ is *simple* if the dimension of the quotient space V_∞/S is infinite and for any infinite dimensional r.e. subspace W of V_∞, $W \cap S \neq \{\vec{0}\}$. A r.e. subspace M is *maximal* if the dimension of V_∞/M is infinite and for any r.e. subspace $W \supseteq M$, either the dimension of V_∞/W or the dimension of W/M is finite. A r.e. subspace M is *supermaximal* if the dimension of V_∞/M is infinite and for any r.e. subspace $W \supseteq M$, either $V_\infty = W$ or the dimension of W/M is finite. The NP analogues of these notions in $st(V_\infty)$ and $tal(V_\infty)$ are the following. Let A be an oracle, then a NP^A subspace S of $st(V_\infty)$ ($tal(V_\infty)$) is NP^A-*simple* if the dimension of $st(V_\infty)/S$ ($tal(V_\infty)/S$) is infinite and for any infinite NP^A subspace W of $st(V_\infty)$ ($tal(V_\infty)$), $W \cap S \neq \{bin(\vec{0})\}$ ($W \cap S \neq \{tal(\vec{0})\}$). A NP^A subspace M is NP^A-*maximal* if the dimension of $st(V_\infty)/M$ ($tal(V_\infty)/M$) is infinite and for any NP^A subspace W of $st(V_\infty)$

($tal(V_\infty)$), either the dimension of $st(V_\infty)/W$ ($tal(V_\infty)/W$) or the dimension of W/M is finite. A NP^A subspace M is NP^A-*supermaximal* if the dimension of $st(V_\infty)/M$ ($tal(V_\infty)/M$) is infinite and for any NP^A subspace W of $st(V_\infty)$ ($tal(V_\infty)$), either $st(V_\infty) = W$ ($tal(V_\infty) = W$) or the dimension of W/M is finite.

Nerode and Remmel [58] introduced a slightly weaker notion than NP^X-simple subspace which they called P^X-simple subspace. Note that in the case of simple sets or simple subspaces, we can replace the infinite r.e. set W or the infinite dimensional r.e. subspace W by an infinite recursive set W or an infinite dimensional recursive subspace. That is, every infinite r.e. set W contains an infinite recursive set and every infinite dimensional r.e. subspace V of V_∞ contains an infinite dimensional recursive subspace. Thus a r.e. set S is simple iff $\omega \setminus S$ is infinite and for any infinite recursive set W, $W \cap S \neq \emptyset$. Similarly an r.e. subspace S of V_∞ is simple iff the dimension of V_∞/S is infinite and for any infinite dimensional recursive subspace W of V_∞, $W \cap S \neq \{\vec{0}\}$. Thus we make the following definition. Let A be an oracle, then a NP^A subspace S of $st(V_\infty)$ ($tal(V_\infty)$) is P^A-*simple* if the dimension of $st(V_\infty)/S$ ($tal(V_\infty)/S$) is infinite and for any infinite dimensional P^A subspace W of $st(V_\infty)$ ($tal(V_\infty)$), $W \cap S \neq \{bin(\vec{0})\}$ ($W \cap S \neq \{tal(\vec{0})\}$). It follows from results of Nerode and Remmel [55] that there exists oracles A such that there exists an infinite dimensional NP^A subspace V of $tal(V_\infty)$ such that V has no infinite dimensional subspace $W \in P^A$. Thus while a subspace W which is NP^A-simple is certainly P^A-simple, it is not clear that every P^A-simple subspace of $tal(V_\infty)$ is NP^A-simple.

Given a subspace V of $st(V_\infty)$ ($tal(V_\infty)$), we let

$$\begin{aligned} D_n(V) &= \{\langle v_1, \ldots, v_n\rangle_n : v_1, \ldots, v_n \text{ are dependent}\} \\ D(V) &= \bigcup_{n \geq 1} D_k(V). \end{aligned}$$

The Turing degree of $D_n(V)$ is called the n-th dependence degree and the Turing degree of $D(V)$ is called the dependence degree of V. (The sets $D_n(V)$ and $D(V)$ can be defined for any subspace of a recursively presented vector space over a recursive field using a suitable coding of the finite sequences of N.) Nerode and Remmel [50] proved the following.

Theorem 8.2 *Assume the underlying field F of $tal(V_\infty)$ is an infinite recursive field. Let $A_0, A_1, A_2, \ldots$ be any effective sequence of r.e. sets such that $A_1 \leq_T A_2 \leq_T \cdots \leq_T A_0$ and A_0 is not recursive. Then there is a supermaximal subspace V in $tal(V_\infty)$ such that $D(V) \equiv_T A_0$ and $D_k(V) \equiv_T A_k$.*

There is a nice application of Theorem 8.2 in the case where we pick $A_1, A_2, \ldots$ to be recursive and A_0 to be nonrecursive. In that case, the supermaximal space V of Theorem 8.2 is recursive so the quotient space $W = tal(V_\infty)/V$ is a recursively presented vector space such that

(i) every r.e. independent set I of W is finite,

(ii) for any fixed n, there is an effective procedure which given an n-tuple $w_1, \ldots, w_n$ will determine if $w_1, \ldots, w_n$ are dependent, but

(iii) W has no dependence algorithm.

Bäuerle [5] proved the following result for $tal(V_\infty)$.

Theorem 8.3 *([5]) Let F be a polynomial time field where Q is properly embedded and δ be any nonzero r.e. degree. Then for any finite $k \geq 1$, there is a supermaximal subspace V of $tal(V_\infty)$ such that*

(i) *$D_1(V), \ldots, D_k(V)$ are polynomial time,*

(ii) *for all j, $D_j(V) \in PSPACE \cap DEXT$, and*

(iii) *$D(V) \in \delta$.*

Thus in particular, there exist a polynomial time supermaximal subspace W of $tal(V_\infty)$ which is of course automatically NP-simple and NP-maximal. Moreover if we consider the quotient space $U = tal(V_\infty/W)$, then it is easy to see that U is a polynomial time vector space. That is, if we identify U with the set of minimal elements in each equivalence class of $tal(V_\infty/W)$, the Q will be a polynomial time set and the operations of $tal(V_\infty)$ will induce polynomial time operations on U which will make it isomorphic to $tal(V_\infty/W)$. Thus we have the following

Theorem 8.4 *There exists a polynomial time presented vector space U such that the only r.e independent sets of U are finite.*

As we shall see in the next section, the analogues of Theorems 8.1 and 8.3 are oracle dependent.

8.2 Subspaces and Bases over finite fields

In this section, we shall state several results on the relation between the complexity of a subspace V of either $st(V_\infty)$ or $tal(V_\infty)$ and the complexity of a basis of that subspace when the underlying field is finite. These results turn out to be essential for many of the more complicated results and constructions in polynomial time vector spaces.

Note that since the universe of $st(V_\infty)$ is $Bin(\omega)$, there is a natural order $<$ on the elements of $st(V_\infty)$ inherited from the standard ordering of the natural numbers. Similarly since the universe of $tal(V_\infty)$ is $Tal(\omega)$, there is a natural order $<$ on the elements of $st(V_\infty)$ inherited from the standard ordering of the natural numbers. This given, we can now state some very useful definitions for our purposes. Recall that $e_1, e_2, \ldots$ is the standard basis for V_∞. Thus $R(e_n) = k^{n-1}$.

We start with the definition of a height increasing basis.

Definition 8.5 *Let V be a subspace of $st(V_\infty)$ or $tal(V_\infty)$.*

(1) *Call B a* **height increasing** *basis of V if B is a basis for V and for all $n \geq 1$, B has at most one element of height n.*

(2) *The* **standard height increasing basis** *of V, B_V, is defined by declaring that $x \in B_V$ iff $x \in V$ and there is no $y \in V$ such that $y < x$ and $ht(y) = ht(x)$.*

(3) *The* **standard height increasing complementary basis** *of $V \subseteq tal(V_\infty)$, $B_{\overline{V}}$, is defined in $tal(V_\infty)$ by declaring that $tal(e_n) \in B_{\overline{V}}$ iff $tal(e_n) \notin V$ and there is no $y \in V$ such that $ht(y) = n$. Similarly the* **standard height increasing complementary basis** *of $V \subseteq st(V_\infty)$, $B_{\overline{V}}$, is defined in $st(V_\infty)$ by declaring that $bin(e_n) \in B_{\overline{V}}$ iff $bin(e_n) \notin V$ and there is no $y \in V$ such that $ht(y) = n$.*

(4) *We call the $space(B_{\overline{V}})$, the* **standard complement** *of V.*

There is a crucial difference between $st(V_\infty)$ and $tal(V_\infty)$ with respect to searches. That is, the vector of height n with the smallest R value is e_n and $R(e_n) = k^{n-1}$. The vector of height n with the largest R value is $(k-1)e_1 + \cdots + (k-1)e_n$ and

$$R((k-1)e_1 + \cdots + (k-1)e_n) = \sum_{i=1}^{N}(k-1)k^{i-1} = k^n - 1.$$

Thus in $tal(V_\infty)$, given a vector v of height n, we can produce in polynomial time in $|v|$, a list of all vectors of height n in $tal(V_\infty)$. However in $st(V_\infty)$, given a vector v of height n, it takes exponential time in $|v|$ to produce a list of all vectors of height n in $st(V_\infty)$. For this reason, the relation between the complexity of V, B_V, $B_{\overline{V}}$, and $space(B_{\overline{V}})$ is very different in $tal(V_\infty)$ than in $st(V_\infty)$. For this reason, we shall divide this subsection into two parts, one for $tal(V_\infty)$ and one for $st(V_\infty)$, and discuss the relation between the complexity of bases and subspaces for each case separately.

8.2.1 Bases and Subspaces for $tal(V_\infty)$.

Nerode and Remmel in [55] studied bases of NP-subspaces of $tal(V_\infty)$, so we start by listing a number of results from that paper.

Theorem 8.6 *([55])*
Let V be a subspace of $tal(V_\infty)$.

(a) *If B is a height increasing basis of V, then $V \leq_T^P B$.*

(b) *$B_V \leq_T^P V$ and $B_{\overline{V}} \leq_T^P V$.*

Proof: The key point here is that in our tally representation, $card(\{x \in V_\infty : ht(x) \leq n\}) = k - 1 + (k-1)k + \ldots + (k-1)k^{n-1} = k^n - 1$. Moreover, if $ht(y) \leq n$, then $|y| < k^n$. Given $x \in V_\infty$ such that $ht(x) = n$, we know that $|x| \geq k^{n-1}$. So there are at most $k|x|$ elements of $tal(V_\infty)$ with height less than or equal to $ht(x)$. For q fixed we can run any (uniform) computation which take at most n^q steps on strings of length n for all the elements of $tal(V_\infty)$ of height less than or equal to $ht(x)$ in polynomial time. This is because

$$\sum_{y \in tal(V_\infty), ht(y) \leq ht(x)} |y|^q \leq \sum_{i=0}^{(k|x|)^q} i = \left(\frac{(k|x|)^q[(k|x|)^q - 1]}{2}\right) \leq k^2|x|^{2q}.$$

Given these observations it is immediate from our definitions of B_V and $B_{\overline{V}}$ that $B_V \leq_T^P V$ and $B_{\overline{V}} \leq_T^P V$.

To prove Theorem 8.6 (a), note that if B is a height increasing basis for V, then $x \in V$ iff $x \in space(\{y \in B : ht(y) \leq ht(x)\})$. Thus to decide if $x \in V$, we simply search all the elements y in V_∞ with $ht(y) \leq ht(x)$ and produce all vectors $y_1, \ldots, y_k$ in $\{y \in B : ht(y) \leq ht(x)\}$. We can then use the polynomial time dependence algorithm to determine if $y \in space(\{y \in B : ht(y) \leq ht(x)\})$ in polynomial time in $|y|$. Thus $V \leq_T^P B$. □

An immediate corollary of Theorem 8.6 is the following.

Corollary 8.7 *([55])*

(i) *A subspace V of $tal(V_\infty)$ is in P iff V has a height increasing basis B in P.*

(ii) *If V is a subspace of $tal(V_\infty)$ and $V \in P$, then V has a complementary subspace W in P.*

We note that one cannot replace $\leq_T^P$ by $\leq_m^P$ in the statement of Theorem 8.6 due to the following result of Nerode and Remmel .

Theorem 8.8 *([55])*
There exists a subspace V of $tal(V_\infty)$ such that neither $B_V \leq_P^m V$ nor $V \leq_P^m B_V$.

Next we observe that height increasing bases in NP generate NP spaces.

Theorem 8.9 *([58] Suppose that A is a height increasing independent set of $tal(V_\infty)$ in NP. Then $space(A) \in NP$.*

Proof: Note that if A is a height increasing independent set, then $x \in space(A)$ iff $x \in space(\{y \in A : ht(y) \leq ht(x)\})$. Thus $x \in space(A)$ iff there are elements $b_1, \ldots, b_n$ of height $\leq ht(x)$ and $\lambda_1, \ldots, \lambda_n \in F$ such that $x = \sum_{i=1}^n \lambda_i b_i$. Moreover, if $ht(x) = m$, then $k^{m-1} \leq |x| \leq k^m - 1$ so that each b_i must have length $\leq k|x|$. Thus in nondeterministic polynomial time, we can guess $\lambda_1, \ldots, \lambda_n$, $b_1, \ldots, b_n$, and computations which show that $b_i \in A$ and then verify that $x = \sum_{i=1}^n \lambda_i b_i$. Thus $space(A)$ is in NP if $A \in NP$. □

Similarly one can show that if $NP^X =$ co-NP^X, then we have the following.

Theorem 8.10 *([55])*
Suppose $NP^X = co\text{-}NP^X$ and V is a subspace of $tal(V_\infty)$. Then

(i) *$V \in NP^X$ iff V has a height increasing basis in NP^X;*

(ii) *$V \in NP^X$ implies V has a complementary subspace W in NP^X.*

Our next result will allow us to show that the property of a subspace V of $tal(V_\infty)$ having a basis in P does not necessarily tell us anything about the complexity of V other than that V is recursively enumerable.

Theorem 8.11 *([55])*
Let V be a recursively enumerable infinite dimensional subspace of $tal(V_\infty)$. Then the following are equivalent:

(1) *V has a basis C in P;*

(2) *V contains an infinite dimensional subspace W in P;*

(3) *V contains an infinite height increasing independent subset S in P.*

Another consequence of a subspace containing an infinite independent subset in P is the following.

Theorem 8.12 *Let V be a recursive subspace of $tal(V_\infty)$ such that V contains an infinite height increasing independent set C in P. Then if the dimension of $tal(V_\infty)/V$ is infinite, there is an infinite height increasing independent set D in P such that $V \cap space(D) = \{\vec{0}\}$.*

Proof: Note that $B_{\overline{V}}$ is recursive. Let $b_0, b_1, \ldots$ be a list of the elements of $B_{\overline{V}}$ such that $h(b_0) < h(b_1) < \ldots$. Let f be a recursive function such that $f(0^n) = b_n$. Similarly let $c_0, c_1, \ldots$ be a list of the elements of C such that $h(c_0) < h(c_1) < \ldots$. Then let $d_s = b_s +_{tal} c_{r(s)}$ where

$$r(s) = 1 + \sum_{i=0}^{s} h(b_i) + \text{the number of steps to compute } f(0), \ldots, f(s).$$

Then we claim that $D = \{d_0, d_1, \ldots\}$ is our required height increasing independent set. First observe that by our definition of $r(s)$, $r(s) > h(b_s)$ so that $h(d_s) = h(c_{r(s)})$. Also it is clear that $r(0) < r(1) < \ldots$ so that $h(d_0) < h(d_1) < \ldots$. Thus D is a height increasing basis. Moreover it is easy to see that D is independent over V. Thus we need only show that D is p-time. To decide whether a given $x \in tal(V_\infty)$ is in D, we first compute which elements y with $h(y) \leq h(x)$ are in C. Now C is a p-time set so that for all z we can determine whether $z \in C$ in $max(2, |z|)^m$ steps for some fixed m. Moreover, if $h(x) = n$, then $x = 1^{|x|}$ where $k^{n-1} \leq |x| \leq k^n - 1$ so that it requires at most

$2^m + 2^m + \sum_{j=2}^{k^n-1} j^m \leq \sum_{j=0}^{k|x|+1} j^m < ((k|x|+1)^m)^2 = (k|x|+1)^{2m}$ steps to find the elements of C of height less than or equal to $h(x)$. If no element of height $h(x)$ is in C, then clearly $x \notin D$. If there is an element of height $h(x)$ in C, then in polynomial time in $|x|$, we can find r such that $h(c_r) = h(x)$. At this point, we start to compute the sequence of elements $f(0), f(1), \ldots$ in order for r steps. Suppose that end the end of r steps, we have successfully computed $f(0), \ldots, f(t)$. Note that if we are not successful in computing $f(0)$ by the end of r steps, then $x \notin D$. Otherwise, see if there is some $s \leq t$ such that

$$r = 1 + \sum_{i=0}^{s} h(b_i) + \text{the number of steps to compute } f(0), \ldots, f(s).$$

If there is no such s, then $x \notin D$ and if there is such an s, then $x \in D$ iff $x = f(s) +_{tal} c_r$. It follows that we can decide if $x \in D$ in polynomial time in $|x|$, so that D is a p-time height increasing independent set which is independent over V. $\square$

Next we show that having a basis in P does not restrict the degree of a subspace other than ensuring the subspace is recursively enumerable.

Theorem 8.13 *Let δ be any r.e. degree. Then there there exists a r.e. subspace V in $tal(V_\infty)$ such that V has a basis in P.*

Proof: Let B_1 be an infinite subset of $\{e_{2n} : n \geq 1\}$ in P and for any given r.e. degree δ, let B_δ be an infinite r.e. subset of $\{e_{2n+1} : n \geq 0\}$ of degree δ. Then it is easy to see that the Turing degree of $V_\delta = space(B_1 \cup B_2)$ is δ. By Theorem 4, V_δ has a basis in P since $space(B_1)$ is an infinite dimensional subspace of V_δ which is in P. $\square$

It is also easy to construct spaces with no basis in P. In fact, Nerode and Remmel [55] gave a general construction which, given any effective list of r.e. independent sets of $tal(V_\infty)$ $A_0, A_1, \ldots$, produced a subspace V of $tal(V_\infty)$ such that $V \cap A_i$ is finite for all. Their construction can be specialized to prove the following results.

Theorem 8.14 **(1)** *There is a subspace V of $tal(V_\infty)$ in $DEXT$ such that V has no basis in P.*

(2) *There is a recursive subspace V of $tal(V_\infty)$ such that V has no primitive recursive basis.*

(3) *There is a subspace V of $tal(V_\infty)$ which is recursive in $\mathbf{0}'$ such that for any r.e. independent set I, $I \cap V$ is finite.*

We should also note that every r.e. subspace has a basis which has high complexity.

Theorem 8.15 *([55])*
Let V be an r.e. subspace of either $tal(V_\infty)$ or $st(V_\infty)$. Then V has a recursive basis B which is not primitive recursive.

All of the results so far do not settle the question of whether every subspace V of $tal(V_\infty)$ which is in NP has a basis in P. In fact, this question is oracle dependent. To prove the existence of an oracle B such that every subspace V of $tal(V_\infty)$ which is in NP^B has a basis in P^B, Nerode and Remmel proved the following result which strengthens a similar result of Homer and Maass [36].

Theorem 8.16 *([55]) There is a recursive oracle B such that $P^B \neq NP^B$ and such that every infinite set X which is p-time Turing reducible to a set Y in NP^B contains an infinite subset in P^B.*

We note that in light of Theorem 8.11, it also follows that for the oracle B of Theorem 8.16, every NP^B subspace V of $tal(V_\infty)$ has a basis in P^B. Thus we have the following.

Theorem 8.17 *([55]) There is a recursive oracle B such that $P^B \neq NP^B$ and every every NP^B subspace V of $tal(V_\infty)$ has a basis in P^B.*

Via a delayed diagonal argument, Nerode and Remmel also proved the following.

Theorem 8.18 *([55]) There is a recursive oracle A such that*

(a) *there is an infinite dimensional subspace V in NP^A such that V has no basis in P^A (and hence $NP^A \neq P^A$) and*

(b) *$NP^A =$ co-NP^A.*

Combining Theorems 8.17 and 8.18, we have the following

Theorem 8.19 *([55]) Arguments valid under relativization are not sufficient to prove*

1. *$P \neq NP \Rightarrow$ every subspace of $tal(V_\infty)$ in NP has a basis in P and*
2. *$P \neq NP \Rightarrow$ there is a subspace V of $tal(V_\infty)$ in NP which has no basis in P.*

We end this section with some results of Bäuerle [5]. We say a set $A \subseteq \{0,1\}^*$ is P^X-immune if there are no infinite subset of A in P^X. The next results show that a subspace $V \subseteq tal(V_\infty)$ can have a basis in P without the standard basis being in P.

Theorem 8.20 *([5]) There exists an exponential time subspace V of $tal(V_\infty)$ which has a basis in P but for which the standard height increasing basis of V, B_V, is P-immune.*

Theorem 8.21 *([5]) There exists a recursive oracle A such that there exists a subspace V of $tal(V_\infty)$ which is in $NP^A \setminus P^A$, has a basis in P^A, and yet the standard hieght increasing basis of V, B_V, is P^A-immune.*

Theorem 8.22 *([5])*

There exists a recursive oracle B such that there exists a subspace V of $tal(V_\infty)$ in $NP^B \setminus P^B$ and such that, for all $NP^B \setminus P^B$ subspaces V of $tal(V_\infty)$, the standard height increasing basis B_V has an infinite subset in P^B.

Theorem 8.23 *Let $\mathcal{F}$ be finite and $V \subset tal(V_\infty)$. If V has an infinite dimensional subspace in P, then V has a height increasing basis D with a subset in P such that $B_V \equiv_T^P D$.*

Theorem 8.24 *([5]) Let A be an oracle such that $NP^A \setminus P^A$-subspaces of $tal(V_\infty)$ exist. Then if V has an infinite dimensional subspace in P^A, then V has a height increasing basis D with a subset in P^A such that $B_V \equiv_T^P D$.*

8.2.2 Bases and Subspaces of $st(V_\infty)$.

It will be convenient to think of $st(V_\infty)$ via the representation $B_k(V_\infty)$ defined above. The advantage is that for nonzero $x \in B_k(V_\infty)$, $ht(x) = |x|$. The standard basis for $B_k(V_\infty)$ is given by $e_n = b_k(k^{n+1}) = 0^n1$.

As pointed out in the introduction to this section, there is a significant difference between $st(V_\infty)$ and $tal(V_\infty)$ with regard to searches. Indeed many of the proofs of the propositions and theorems in the previous subsection relied on the fact that given an $x \in tal(V_\infty)$, we could produce a list of all elements $tal(V_\infty)$ of height $\leq ht(x)$ in polynomial time in $|x|$. This is no longer the case in $st(V_\infty)$. That is, if $x \in tal(V_\infty)$ and $ht(x) = n$, then $k^{n-1} \leq |x| \leq k^n - 1$ while if $x \in st(V_\infty)$, then $ht(x) = |x|$ so that there are $k^{|x|}-1$ elements of height less than or equal to $ht(x)$ in $st(V_\infty)$. Thus in $st(V_\infty)$, we can not find all the elements of height less than or equal to $ht(x)$ in a p-time height increasing set S in polynomial time in $|x|$. However there is a special class of p-time independent sets of $st(V_\infty)$, which we call strongly p-time independent sets, which do have most of the useful properties possessed by p-time height increasing bases of $tal(V_\infty)$.

Definition 8.25 *An independent set $B \subseteq st(V_\infty)$ is called* **strongly p-time** *if*

(i) *B is a p-time set,*

(ii) *B is height increasing, and*

(iii) *if $B = \{b_0, b_1, \ldots\}$ where $ht(b_0) < ht(b_1) < \ldots$, then there is a polynomial time function f such that for all $n > 0$*

> **(iiia)** *$f(1^n) = b_k$ if $ht(b_k) = n$ and B has an element of height n,*

(iiib) *$f(1^n) = 0$ if B has no element of height n.*

We note that condition (iii) allows us to find, for any $x \in st(V_\infty)$, all elements of b of height $\leq ht(x)$ in polynomial time in $|x|$. That is, given $x \in st(V_\infty)$, $ht(x) = |x|$ and we can compute $f(1), f(1^2), \ldots, f(1^{ht(x)})$ in polynomial time in $|x|$. Then $\{b : b \in B \wedge ht(b) \leq ht(x)\} = \{f(1^n) : n \leq |x| \wedge f(1^n) \neq 0\}$. As noted above, any p-time height increasing independent set B in $tal(V_\infty)$ also has the property that, for any x, we can find all elements of B of height $\leq ht(x)$ in polynomial time in $|x|$. Thus condition (iii) is specifically designed to give us this property which holds for all p-time height increasing bases in $tal(V_\infty)$ automatically. It is easy to see that our standard basis $\{e_1, e_2, e_3, \ldots\}$ of $st(V_\infty)$ is strongly p-time.

Our next proposition lists several basic properties of subspaces generated by subsets of a strongly p-time basis.

Theorem 8.26 *Let B be a strongly p-time basis of $st(V_\infty)$ and suppose that $S \subseteq B$. Then*

(i) *$S \in P$ iff $space(S) \in P$.*

(ii) *$S \in NP$ iff $space(S) \in NP$.*

(iii) *$S \in$ co-NP iff $space(S) \in$ co-NP.*

(iv) *$S \equiv_T^P space(S)$.*

Proof: Since $S = space(S) \cap B$, it follows that $S \leq_T^P space(S)$ and S is in $P(NP, \text{co-}NP)$ if $space(S)$ is in $P(NP, \text{co-}NP)$.

Let f be the p-time function such that $f(1^n) = b_n$, where b_n is the element of height n in B. Then, given an $x \in st(V_\infty)$ of height n, we can compute $f(1) = b_1, \ldots, f(1^n) = b_n$ and test $b_1, \ldots, b_n$ for membership in S, all in time polynomial in $|x|$. Thus in polynomial time in $|x|$, we can find $\{s_1, \ldots, s_k\}$, where $\{b_{s_1}, \ldots, b_{s_k}\} = \{y \in S : ht(y) \leq ht(x)\}$. Moreover the fact that B is a height increasing basis means that $x = \sum_{i=1}^{|x|} \lambda_i b_i$ for some $\lambda_1, \ldots, \lambda_{|x|}$ in F. Now suppose that $|x| = n$, then we can write $x = x_1 \ldots x_n$ where all $x_i \in F$ and and each $b_i = b_{i,1} \ldots b_{i,n}$ where $b_{i,j} \in F$. Then we can solve the matrix equation over F

$$BY = X$$

where $B = (b_{i,j})$, Y is a column vector of unknowns, and X is the column vector $(x_1, \ldots, x_n)$ in polynomial time in $n = |n|$. Thus in polynomial time in $|x|$, we can find $\lambda_1, \ldots, \lambda_k$ such that $x = \sum_{i=1}^{|x|} \lambda_i b_i$. This given,

$$x \in space(S) \text{ iff } \{i : \lambda_i \neq 0\} \subseteq \{s_1, \ldots, s_k\}.$$

It then easily follows that $space(S) \leq_T^P S$ and $space(S)$ is in P $(NP, \text{co-}NP)$ if S is in P $(NP, \text{co-}NP)$. □

Our next result is a weak analogue for $st(V_\infty)$ of Theorem 8.7.

Theorem 8.27 *Let V be a subspace of $st(V_\infty)$ with strongly p-time basis R. Then $R \cup B_{\bar{V}}$ is a strongly p-time basis for $st(V_\infty)$ and both V and $space(B_{\bar{V}})$ are in P.*

Our next theorem shows that no extra condition on a height increasing basis, such as condition (iii), is required to generate subspaces of $st(V_\infty)$ in NP.

Theorem 8.28 *Let B be a height increasing independent set of $st(V_\infty)$ which is in NP. Then $space(B)$ is in NP.*

Proof: The key property of a height increasing basis is that if $x \in space(B)$, then $x \in space(\{b \in B : ht(b) \leq ht(x)\})$. That is, x must be generated by the elements of height $\leq ht(x)$ in B if $x \in space(B)$. Thus to see that $space(B) \in NP$, we simply guess the elements of B of height $\leq ht(x)$, say $\{b_1, \ldots, b_k\} = \{b \in B : ht(b) \leq ht(x)\}$, where $ht(b_1) < \ldots < ht(b_k)$. Now, for all nonzero $y \in st(V_\infty)$, $ht(y) = |y|$ so $|b_i| \leq |x|$ for all i and $k \leq |x|$. Then we perform a nondeterministic polynomial time computation to check if $b_1, \ldots, b_k$ are all in B. Finally, we use our polynomial time dependence algorithm to check whether $x \in space(\{b_1, \ldots, b_k\})$. Thus $space(B)$ is in NP. $\square$

Theorem 8.29 *Suppose NP^X = co-NP^X and V is a subspace of $st(V_\infty)$. Then*

(i) $V \in NP^X$ iff V has a height increasing basis in NP^X.

(ii) $V \in NP^X$ implies V has a complementary subspace W in NP^X.

Our next result is a weak analogue of Theorem 8.11 of [55] for $st(V_\infty)$.

Theorem 8.30 *Let V be an r.e. infinite dimensional subspace of $st(V_\infty)$. Suppose that there exists an infinite strongly p-time independent subset $I \subseteq V$. Then V has a basis in P.*

Our next next result is the analogue of Theorem 8.12 for $st(V_\infty)$.

Theorem 8.31 *Let V be a recursive co-infinite dimensional subspace of $st(V_\infty)$ such that V contains an infinite strongly p-time height increasing independent set C. Then there is an infinite strongly p-time height increasing independent set D such that $V \cap space(D) = \{\vec{0}\}$.*

Theorem 8.32 *Given any r.e. Turing degree δ, there exists an r.e. subspace V of $st(V_\infty)$ such that V has degree δ and V has a basis in P.*

Again one can show that exists an exponential time subspace of $st(V_\infty)$ which has no basis in P.

Theorem 8.33 *There is a subspace V of $st(V_\infty)$ such that $V \in DEXT$ and V has no basis in P.*

8.2.3 The semilattice of NP^X subspaces

In this section we shall study various properties of the lower semilattice of NP^X-subspaces of $tal(V_\infty)$ and $st(V_\infty)$ for various oracles X. Our first result shows that in contrast to the collection of r.e. subspaces which is closed under both intersection ($\cap$) and sum ($+$) and hence forms a lattice, the collection of NP^X-subspaces of either $tal(V_\infty)$ and $st(V_\infty)$ is only closed under intersection and hence only forms a lower semilattice.

Theorem 8.34 *There exist two polynomial time subspaces W and V of $tal(V_\infty)$ ($st(V_\infty)$) such that $W \cap V = \{\vec{0}\}$ and $W + V$ is not recursive.*

Proof: The proof that we present below works equally well for both $tal(V_\infty)$ and $st(V_\infty)$. Thus we shall write a generic proof where V_∞ may be interpreted as either $tal(V_\infty)$ or $st(V_\infty)$ and the standard basis $e_1, e_2, \ldots$ may be interpreted as either the standard basis $tal(e_1), tal(e_2), \ldots$ of $tal(V_\infty)$ or the standard basis $st(e_1), st(e_2), \ldots$ of $st(V_\infty)$ as appropriate.

By a result of Metakides and Nerode [47], a subspace V of V_∞ is recursive iff V is r.e. and V has an r.e. complementary space. It is easy to see that we can form an effective list $(A_0, B_0), (A_1, B_1), \ldots$ of all pairs of r.e. subspaces W_i and W_j of V_∞ such that $W_i \cap W_j = \{\vec{0}\}$. That is, if $W_0, W_1, \ldots$ is an effective list of all r.e. subspaces of V_∞ and W_i^n denotes the set of elements enumerated into W_i after n steps, then (A_i, B_i) is the pair of r.e. subspaces given by letting (A_i, B_i) be (W_k, W_ℓ) iff $i = [k, \ell]$ and $W_k \cap W_\ell = \{\vec{0}\}$ or letting (A_i, B_i) be $(space(W_k^n), space(W_\ell^n))$ where n is the least m such that $space(W_k^{m+1}) \cap space(W_\ell^{m+1}) \neq \{\vec{0}\}$ if $W_k \cap W_\ell \neq \{\vec{0}\}$.

Given the list $(A_0, B_0), (A_1, B_1), \ldots$, we shall construct W and V so that $W + V \neq A_i$ for any i such that $A_i + B_i = V_\infty$. Thus $W + V$ will not be recursive. In the construction that follows we will in fact construct two p-time height increasing disjoint independent sets K and L so that $W = space(K)$ and $V = space(L)$ will be our desired polynomial time subspaces. Let $r_0, r_1, \ldots$ be a list of all prime numbers in increasing order. Our idea is to use the vectors $e_{r_i} + e_{r_i \cdot 2^n}, e_{r_i \cdot 2^n}$ where $n \geq 1$ to help us ensure that $A_i \neq W + V$ if $A_i + B_i = V_\infty$. The only vectors which will be placed into K will be of the form $e_{r_i} + e_{r_i \cdot 2^n}$ for some $i \geq 0$ and $n \geq 1$, and the only vectors which will be placed into L will be of the form $e_{r_i \cdot 2^n}$ for some $i \geq 0$ and $n \geq 1$. In fact, for any fixed i either

$$K \cap \{e_{r_i} + e_{r_i \cdot 2^n} : \ n \geq 1\} = \emptyset$$

and

$$L \cap \{e_{r_i \cdot 2^n} : \ n \geq 1\} = \emptyset$$

or there will be an m such that

$$K \cap \{e_{r_i} + e_{r_i \cdot 2^n} : \ n \geq 1\} = \{e_{r_i} + e_{r_i \cdot 2^m}\}$$

and

$$L \cap \{e_{r_i \cdot 2^n} : \; n \geq 1\} = \{e_{r_i \cdot 2^m}\}.$$

Note that in the standard representation of V_∞, L will be a polynomial time subset in the strongly p-time height increasing basis $\{st(e_n) : n > 0\}$ and K will be a polynomial time subset of the strongly p-time height increasing independent set $\{e_k + e_{k \cdot 2^n} : k \text{ is odd and } n \geq 1\}$ so that L and K themselves will be strongly p-time independent sets. Thus by Theorem 8.26, W and V will be polynomial time subspaces of $st(V_\infty)$. In the tally representation of V_∞, K and L will be polynomial time height increasing independent sets so that by Theorem 8.6, W and V will be polynomial time subspaces of $tal(V_\infty)$.

Now to decide if $e_{r_i} + e_{r_i \cdot 2^m} \in K$ and $e_{r_i \cdot 2^m} \in L$, we run the enumerations of A_i and B_i for m steps. Let A_i^m and B_i^m denote those elements enumerated into A_i and B_i respectively after m steps. If $m > |e_{r_i}|$ and $[space(A_i^m) + space(B_i^m)] \setminus [space(A_i^{m-1}) + space(B_i^{m-1})] \neq \emptyset$, then we place $e_{r_i} + e_{r_i \cdot 2^m}$ into K and $e_{r_i \cdot 2^m}$ into L iff $e_{r_i} \in [space(A_i^m) + space(B_i^m)] \setminus space(A_i^m)$. Otherwise we place neither $e_{r_i} + e_{r_i \cdot 2^m}$ into K nor $e_{r_i \cdot 2^m}$ into L. Using the fact that in m steps, we can at most enumerate m vectors which are of length at most m and the fact that Gaussian elimination is polynomial time in the dimensions of the matrix, it is easy to see that both K and L are p-time height increasing independent sets.

Now suppose $e_{r_i} + e_{r_i \cdot 2^m} \in K$ and $e_{r_i \cdot 2^m} \in L$. Since $A_i \cap B_i = \{\vec{0}\}$, we know that each element $v \in space(A_i) + space(B_i)$ has a unique expression in the form $v = a + b$ with $a \in space(A_i)$ and $b \in space(B_i)$. By our construction, it follows that $e_{r_i} \in [space(A_i^m) + space(B_i^m)] \setminus [space(A_i^m)]$ so that $e_{r_i} \notin space(A_i)$. But clearly $e_{r_i} \in space(K) + space(L)$, so that $A_i \neq space(K) + space(L)$.

Suppose there is no m such that $e_{r_i} + e_{r_i \cdot 2^m} \in K$ and $e_{r_i \cdot 2^m} \in L$. Then either there is no m such that $e_{r_i} \in space(A_i^m) + space(B_i^m)$ in which case $space(A_i) + space(B_i) \neq V_\infty$ so that we don't have to worry about A_i and B_i, or $e_{r_i} \in space(A_i^m)$ for some m (in which case $e_{r_i} \notin space(K) + space(L)$ so again $space(K) + space(L) \neq A_i$). □

Next we make some observations about the existence of subspaces V of $tal(V_\infty)$ which are in $NP \setminus P$. We note that even with the assumption $P \neq NP$, the existence of $NP \setminus P$-subspaces requires further complexity theoretic assumptions. That is, in [34] Hartmanis proved that the existence of sparse sets in $NP \setminus P$ is equivalent to the separation of deterministic and nondeterministic exponential time $DEXT \neq NEXT$. Thus if $DEXT = NEXT$, then no $NP \setminus P$subspaces of $tal(V_\infty)$ can exist even if $NP \neq P$. Since the existence of an oracle such that $NP^A \neq P^A$ and $DEXT^A = NEXT^A$ was proven by Wilson in [75], we have the following theorem.

Theorem 8.35 *There exists an oracle A such that $NP^A \neq P^A$ and no $NP^A \setminus P^A$-subspaces of $tal(V_\infty)$ exist.*

As a consequence of this theorem it follows that showing the existence of $NP \setminus P$-subspaces is at least as hard as separating $DEXT$ and $NEXT$. On the other hand it is sufficient to separate $DOUBDEXT$ and $DOUBNEXT$ to show the existence of $NP \setminus P$-subspaces.

Theorem 8.36 *([5]) If $DOUBDEXT \neq DOUBNEXT$, then $NP \setminus P$-subspaces of $tal(V_\infty)$ over finite fields exist.*

Sketch of Proof: Let $A \in DOUBNEXT \setminus DOUBDEXT$ and assume the underlying field F has k elements. Define $A_0 = \{0^{k^n} | \exists x \in A[n = 1x]\}$. Since $A \in DOUBNEXT \setminus DOUBDEXT$, it follows that $A_0 \in NP \setminus P$. But clearly $A_0 \subset \{tal(e_1), tal(e_2), \ldots\}$ and and hence A_0 is a height increasing independent subset in $NP \setminus P$. It thus follows from Theorem 8.6 and 8.9 that $space(A_0)$ is in $NP \setminus P$. □

Corollary 8.37 *There exist recursive oracles A such that there are $NP^A \setminus P^A$-subspaces of $tal(V_\infty)$.*

Furthermore Mahaney [45] has shown that the existence of a sparse NP-complete set with respect to $\leq_m^P$ implies $NP = P$. Thus, if $P \neq NP$, then there cannot be a subspace V of $tal(V_\infty)$ which is NP-complete.

Next we turn our attention to the question of whether NP-maximal or NP-simple subspaces exist. We note that Breitbart [7] proved that if R is any infinite recursive set in $\{0,1\}^*$, then there exists a set S in P such that both $S \cap R$ and $R \setminus S$ are infinite. This results shows that there can be no NP-maximal sets since if $M \in NP$ and $R = \{0,1\}^* \setminus M$ is infinite, then certainly R is an infinite recursive set. Thus there is a set $S \in P$ such that both $S \cap R$ and $R \setminus S$ are infinite. But then $W = S \cup M$ is a set in NP such that both $W \setminus M$ and $\{0,1\}^* \setminus M$ are infinite so that M is not NP-maximal. Nerode and Remmel [55] proved that the analogue of Breidbart's splitting theorem holds for recursive subspaces of $tal(V_\infty)$ and $st(V_\infty)$.

Theorem 8.38 *Let V be an infinite dimensional recursive subspace of $tal(V_\infty)$ $(st(V_\infty))$. Then there exist subspaces B_0 and B_1 in P such that $B_0 \cap B_1 = \{\vec{0}\}$, $B_0 + B_1 = tal(V_\infty)$ $(B_0 + B_1 = st(V_\infty))$, and both $B_0 \cap V$ and $B_1 \cap V$ are infinite dimensional.*

We note that unlike the set case, Theorem 8.38 does not exclude the possibility of the existence of NP-maximal sets. That is, suppose V is an infinite and co-infinite dimensional subspace of $tal(V_\infty)$. Then the complementary subspace of V, $space(B_{\overline{V}})$, is certainly recursive so that there exists a pair of polynomial time complementary subspaces, U and W, so that $U \cap space(B_{\overline{V}})$ and $W \cap space(B_{\overline{V}})$ are infinite dimensional. However in this case, we can not make the conclusion that $V + U$ is a NP subspace which witnesses that V is not NP-maximal for two reasons. First there is no guarantee that $V + U$ is co-infinite dimensional and second, in light of Theorem 8.34, there is no guarantee that $U + V$ is in NP. Indeed our next results will show that there are oracles A for which NP^A-maximal sets exists. Similar remarks holds for $st(V_\infty)$.

First we show that the assumption that $NP^X = \text{co-}NP^X$ also eliminates the possibility of the existence of NP^X-simple and NP^X-maximal subspaces of $tal(V_\infty)$.

Theorem 8.39 *Suppose that $NP^X = co\text{-}NP^X$ and V is an NP^X subspace of $tal(V_\infty)$ such that $tal(V_\infty)/V$ is infinite dimensional. Then V is not NP^X-simple and V is not NP^X-maximal.*

Proof: By Theorem 8.10, it follows that $space(B_{\overline{V}}) \in NP^X$ so that V is not NP^X-simple. To see that V is not NP^X maximal, note that by our argument in Theorem 8.10, it follows that for any given $x \in NP^X$, we can nondeterministically from an X oracle find a list of all elements $u_1 < \ldots < u_s$ of height $\leq ht(x)$ which are in B_V and a list of all elements $v_1 < \ldots < v_t$ of height $\leq ht(x)$ which are in $B_{\overline{V}}$. Thus we can form a new NP^X height increasing independent set C where $x \in C$ iff $x = u_i$ for some $i \leq s$ or $x = v_{2k}$ for some $2k \leq t$. It is then easy to see that both $tal(V_\infty)/space(C)$ and $space(C)/V$ are infinite dimensional. It also follows from Theorem 8.10 that $space(C) \in NP^X$ so that C witnesses that V is not NP^X-maximal. $\Box$

Since Baker, Gill and Solovay [4] produced recursive oracles X such that $NP^X \neq P^X$ but $NP^X = \text{co-}NP^X$, we have the following.

Theorem 8.40 *There exists a recursive oracle A such that $NP^A \neq P^A$ and there are no NP^A-simple or NP^A-maximal subspaces of $tal(V_\infty)$.*

We note that the construction of Theorem 8.39 does not construct a P^X-subspace W such that $W \cap V = \{\vec{0}\}$ since it is a priori possible that $space(B_{\overline{V}})$ does not contain an infinite dimensional subspace in P^X. Thus we do not automatically rule out the possibility of the existence of P^X-simple subspaces of $tal(V_\infty)$ with the assumption that $NP^X = \text{co-}NP^X$. We shall see a bit later that there exist oracles A such that no NP^A-simple, P^A-simple, or NP^A-maximal subspaces exists in $tal(V_\infty)$.

It is also the case that if a subspace V of $tal(V_\infty)$ has an infinite height increasing independent subset in P, then V is not P-simple or NP-simple.

Corollary 8.41 *([58]) Let $V \in NP$ be subspace of $tal(V_\infty)$ such that V contains an infinite height increasing independent set C in P. Then V is not NP-simple or P-simple*

Proof: We may assume that V is co-infinite dimensional since otherwise V cannot be NP^A-simple or P^A-simple. We can thus use the proof of Theorem 8.12 to construct a p-time infinite height increasing independent set D such that D is independent over V. It follows by Theorem 8.7, that $space(D)$ is a p-time subspace of $tal(V_\infty)$. Since D is independent over V, $space(D) \cap V = \{\vec{0}\}$ so that V is not NP-simple or P-simple. $\square$

To prove that there exists a recursive oracle B such that $NP^B \neq P^B$ and yet no NP^B-maximal, NP^B-simple, or P^B-simple subspaces exist, we can again use the oracle from Theorem 8.16.

Theorem 8.42 *There is a recursive oracle B such that $P^B \neq NP^B$ and no NP^B-maximal, NP^B-simple, or P^B-simple subspaces of $tal(V_\infty)$ exist.*

Proof: Let B be the recursive oracle of Theorem 8.16. Let V be a NP^B subspace of $tal(V_\infty)$ such that the dimension of $tal(V_\infty)/V$ is infinite. By Theorem 8.6, $B_{\overline{V}}$ is p-time Turing reducible to V so that $B_{\overline{V}}$ contains an infinite subset E in P^B. Thus E is an infinite height increasing independent set in P^B so that by Theorem 8.6, $space(E)$ is an infinite dimensional subspace in P^B. Clearly, $space(E) \cap V = \{\vec{0}\}$ so that $space(E)$ witnesses that V is not P^B-simple or NP^B-simple. Moreover, since we can test whether $tal(e_1), \ldots, tal(e_n)$ are in E in polynomial time in $|tal(e_n)|$, the set $E_2 = \{tal(e_n) \in E : card(E \cap \{tal(e_1), \ldots, tal(e_n)\}) \text{ is even}\}$ is also a p-time height increasing independent set. We claim that $W = space(V \cup E_2)$ is a subspace of $tal(V_\infty)$ which witnesses that V is not NP^B-maximal. Note that $B_V \cup E_2$ is a height increasing basis for W and that $E \setminus E_2 \subseteq B_{\overline{W}}$. Thus $W \supseteq V$ and the dimensions of both $tal(V_\infty)/W$ and W/V are infinite. Because $B_V \cup E_2$ is a height increasing basis for W, it follows that $x \in W$ iff there exists a $b \in V$ and an $e \in space(E_2)$ such that $x = b +_{tal} e$ and $ht(b), ht(e) \leq ht(x)$. Thus given a B-oracle, we can nondeterministically guess b and e of length $\leq k|x|$ and the computation which shows that $b \in V$, and then verify in polynomial time that $x = b +_{tal} e$ and $e \in space(E_2)$. Thus $W \in NP^B$ and hence V is not NP^B-maximal. $\square$

Nerode and Remmel [58] showed that the assumption that
$NP^X =$ co-NP^X also eliminates the possibility of the existence of NP^X-simple and NP^X-maximal sets in $st(V_\infty)/V$.

Theorem 8.43 *Suppose that $NP^X =$ co-NP^X and V is an NP^X subspace of $st(V_\infty)$ such that $st(V_\infty)/V$ is infinite dimensional. Then V is not NP^X-simple and V is not NP^X-maximal.*

As was the case for $tal(V_\infty)$, we can use the Baker-Gill-Solovay results to prove the following.

Theorem 8.44 *There exists a recursive oracle A such that $NP^A \neq P^A$ and there are no NP^A-simple or NP^A-maximal subspaces of $st(V_\infty)$.*

The analogue of Theorem 8.41 for $st(V_\infty)$ is the following.

Theorem 8.45 *([58])*
Let V be a NP co-infinite dimensional subspace of $st(V_\infty)$ such that V contains

an infinite strongly p-time height increasing independent set C. Then V is not NP-simple or P-simple

Proof: Use the proof of Theorem 8.31 to construct a strongly p-time infinite height increasing independent set D such that D is independent over V. It follows by Theorem 8.26, that $space(D)$ is a p-time subspace of (V_∞). Since D is independent over V, $space(D) \cap V = \{\vec{0}\}$ so that V is not NP-simple or P-simple. □

One can again use the oracle of Theorem 8.16 to prove that there is an oracle B where no NP^B-maximal, NP^B-simple, nor P^B-simple subspaces of $st(V_\infty)$ exist.

Theorem 8.46 *There is a recursive oracle B such that $P^B \neq NP^B$ and no NP^B-maximal, NP^B-simple, or P^B-simple subspaces of $st(V_\infty)$ exist.*

In contrast to the set case, there are oracles X for which NP^X-maximal subspaces of $tal(V_\infty)$ and $st(V_\infty)$ exists. The proof requires a priority argument for the construction of the oracle. Such arguments are easier in $tal(V_\infty)$ than in $st(V_\infty)$. In $tal(V_\infty)$, one can naturally follow the usual practice in oracle constructions and make the desired NP^X-maximal subspace V be given by

$$V = \{1^n : (\exists \sigma \in X) : \ |\sigma| = n\}.$$

This is not possible in $st(V_\infty)$. In $st(V_\infty)$, one constructs X so that there is a NP^X independent set which generates the desired NP^X-maximal subspace. To see the difference between these two type of construction, we will give the full argument for $tal(V_\infty)$ and give just the construction for $st(V_\infty)$. We note that similar techniques are used to prove results in the standard and tally representations of the free Boolean algebra which are given in the next section.

Theorem 8.47 *There exists an r.e. oracle Y and a subspace V of $tal(V_\infty)$ which is both P^Y-simple and NP^Y-maximal.*

Proof: We shall construct Y so that

$$M = \{0\} \cup \{1^n : n > 0 \ \& \ (\exists \alpha \in \{0,1\}^*)(|\alpha| = n \text{ and } \alpha \in Y)\}$$

is our desired subspace. Clearly $M \in NP^Y$.

To ensure that M is co-infinite dimensional we must meet the following set of requirements.

$$T_j : card(\{n | \ Y \text{ contains no strings } \alpha \text{ with } k^n \leq |\alpha| < k^{n+1} - 1\}) \geq j$$

Thus T_j says there are at least j heights n so that M contains no strings of height n. So meeting requirement T_j ensures $\dim(V_\infty/M) \geq j$.

To ensure that M is P^Y-simple, we shall meet the following two sets of requirements. Given any subset $V \subseteq tal(V_\infty)$, let $ht(V) = \{n : (\exists x \in V) ht(x) = n\}$

S_j : If N_j^Y is an infinite dimensional subspace of $tal(V_\infty)$ such that $ht(N_j^Y) \setminus ht(M)$ is infinite, then $M \cap N_j^Y \neq \{\vec{0}\}$.

Now suppose that P_i^Y is an infinite dimensional subspace of $tal(V_\infty)$. Note that meeting all the requirements S_j will ensure that either $P_i^Y \cap M \neq \{\vec{0}\}$ or $ht(P_i^Y) \subseteq^* ht(M)$ where for any two sets A and B where we write $A \subseteq^* B$ iff there is a finite set F such that $A \subseteq (B \cup F)$. Now suppose that $ht(P_i^Y) \subseteq^* ht(M)$ and let B_i be the standard height increasing basis for P_i^Y. By Lemma 8.6, B_i is in P^Y. Then clearly we can modify B_i by possibly deleting a finite set of elements to form a new height increasing basis C_i such that $ht(M) \supseteq \{n : (\exists x \in C_i) ht(x) = n\}$. Thus C_i will also be in P^Y and by Lemma 8.6, $space(C_i)$ will also be in P^Y. Hence if $ht(P_i^Y) \subseteq^* ht(M)$, then there exists some j such that P_j^Y is an infinite dimensional subspace of $tal(V_\infty)$ and $ht(P_j^Y) \subseteq ht(M)$. Thus to ensure that M is P^Y simple, it will be enough to ensure that we meet the following set of requirements.

R_i: If P_i^Y is an infinite dimensional subspace of $tal(V_\infty)$, then $ht(P_i^Y) \not\subseteq ht(M)$.

Finally, to ensure M is NP-maximal, we shall meet the following set of requirements.

$Q_{[i,n]}$: If N_i^Y/M is infinite dimensional and $N_i^Y \supseteq M$, then there is an $x \in N_i^Y$ such that $x + tal(e_n) \in M$.

Note that if $N_i^Y \supseteq M$ and $\dim(N_i^Y/M)$ is infinite, then meeting all the requirements $Q_{[i,n]}$ will ensure that $tal(e_n) \in N_i^Y$ for all n so that $N_i^Y = tal(V_\infty)$. Thus in fact, M will be NP^Y-supermaximal.

We shall rank our requirements with those of highest priority coming first as $T_0, S_0, R_0, Q_0, T_1, S_1, R_1, Q_1, \ldots$.

In the construction that follows, we shall let Y_s denote the set of elements enumerated into Y by the end of stage s and

$$M_s = \{0\} \cup \{1^\ell : l > 0 \ \& \ (\exists \alpha \in \{0,1\}^*)(|\alpha| = \ell \ \& \ \alpha \in Y_s)\}.$$

We shall ensure that for each s, M_s is a finite dimensional subspace of $tal(V_\infty)$ and that $ht(M_s)$ is contained in $\{1, \ldots, s\}$. For any stage s, we let $CH_s = \{n_1^s < n_2^s < \ldots\}$ be set of complementary heights for M_s, i.e. the set of all heights n so that there are no elements of $tal(V_\infty)$ of height n in M_s.

At any given stage s, we shall pick out at most one requirement A_j where A_j will be one of the requirements S_j, R_j, or Q_j and take an action to meet that requirement. The fact that the requirements T_j will be satisfied follows from the construction described below. For the other requirements, we shall then say that A_j *received attention* at stage s.

The action that we take to meet the requirement A_j of the form S_j or Q_j will always be of the same form. That is, we shall put some elements into Y

at stage s and possibly restrain some elements from entering Y for the sake of the requirement. We shall let $res(A_j, s)$ denote the set of elements that are restrained from entering Y at stage s for the sake of requirement A_j. We say that requirement A_j of the form S_j or Q_j is **satisfied** at stage s, if there is a stage $s' < s$ such that A_j has received attention at stage s' and $res(A_j, s') \cap Y_s = \emptyset$.

The actions that we take to meet the requirements R_j will be slightly different. First, we shall declare that all R_j are in a *passive state* at the start of our construction. We would like to find an element $x \in P_j^{Y_s}$ of height n such that $n \notin ht(M_s)$. If we can find such an x, then we will restrain all y such that $k^{n-1} \leq |y| \leq k^n - 1$ plus all elements not in Y_s which are queried of the oracle Y_s during the computation of $P_j^{Y_s}(x)$ from entering Y for the sake of requirement R_j. Thus if we ensure that $res(R_j, s) \cap Y = \emptyset$, then M will have no elements of height n and $x \in P_j^Y$ so that $ht(P_j^Y) \not\subseteq ht(M)$. If we take such an action for R_j at stage s, then we will say that R_j has received attention at stage s and declare the state of R_j to be *active*. Then for all $t > s$, we will say that an active R_j is **satisfied** at stage t, if $res(R_j, s) \cap Y_t = \emptyset$. However if R_j is injured at some stage $t > s$ in the sense that $res(R_j, s) \cap Y_t \neq \emptyset$, then R_j will return to a passive state. If we cannot find such an x, we will attempt to force $ht(P_j^Y)$ to be finite. That is, since we will ensure that $ht(M_{s-1}) \subseteq \{0, \ldots, s-1\}$ for all s, M_{s-1} will have no elements of height s. Recall that we are assuming that for $n \geq 0$, the run time of computations of $P_j^X(y)$ for any oracle X is bounded $max(2, n)^j$ for any string of length n. Then for $n \geq 2$, we let b_n be the largest i such that for all $k^{n-1} \leq r \leq k^n - 1$,

$$(k^n)^{(i+2)} < 2^{k^{n-2}}.$$

Note that it is easy to see that $lim_{s \to \infty} b_s = \infty$. Our idea is that elements of height n in $tal(V_\infty)$ are of the form 1^r where $k^{n-1} \leq r \leq k^n - 1$. Our strategy at the end of stage $s-1$ for $s \geq 2$ will be to ensure that for all R_j with $j \leq b_s$ which are in a passive state and have the property that $P_j^{Y_{s-1}}(1^r) = 0$ for all $k^{s-1} \leq r \leq k^s - 1$, we restrain all elements which are not in Y_{s-1} and which are queried of the oracle Y_{s-1} in such computations from entering Y for the sake of R_j. This action will force $ht(P_j^Y)$ to be finite if R_j is in a passive state at stage s for all but finitely many s. For any fixed $j \leq b_s$, the maximum restraint imposed for R_j is if we restrained all elements not in Y_{s-1} which are queried of the oracle Y_{s-1} in some computation $P_j^{Y_{s-1}}(1^r) = 0$ with $1 \leq r \leq k^n - 1$. Since the total number of steps used in all these computations is at most

$$2^j + \sum_{i=2}^{k^s} i^j \leq k^s \cdot (k^s)^j = (k^s)^{(j+1)},$$

then clearly we could have restrained at most $(k^s)^{(j+1)}$ elements from entering Y for the sake of R_j. Thus at stage s, we will have restrained at most

$$\sum_{i=0}^{b_s}(k^n)^{(i+1)} < (k^n)^{(b_s+2)} < 2^{k^{(n-2)}}$$

elements for entering Y for the sake of some passive requirement R_j with $j \leq b_s$ at stage $s-1$. Hence for any given r with $k^{n-1} \leq r \leq k^n - 1$, we will have restrained at most 2^{r-1} elements of length r from entering Y for such R_j's.

CONSTRUCTION.

Stages $0, 1$.
Let $Y_0 = Y_1 = \emptyset$ so that $M_0 = M_1 = \{\vec{0}\}$. Let $res(A_j, 0) = res(A_j, 1) = \emptyset$ for all requirements A_j of the form S_j, R_j, or Q_j.

Stage s with $s \geq 2$.
Let A_j be the highest priority requirement among $S_0, R_0, Q_0, \ldots, S_s, R_s, Q_s$ such that

Case 1. $A_j = S_j$ and S_j is not satisfied at stage $s-1$ and there exists an ℓ with $0 < ht(1^\ell) \leq s$ such that

(a) $1^\ell \in N_j^{Y_{s-1}}$,

(b) $ht(1^\ell) \in CH_{s-1}$ and $ht(1^\ell) > n_j^{s-1}$, and

(c) for each $1^n \in space(\{1^\ell\} \cup M_{s-1}) \setminus M_{s-1}$, there is a string $\alpha_n \in \{0,1\}^*$ such that $|\alpha_n| = |1^n| = n$ and α_n is not restrained from Y by any requirement of higher priority than S_j at stage $s-1$ nor is α_n queried of the oracle in some fixed computation of $N_j^{Y_{s-1}}$ which accepts 1^ℓ.

Case 2. $A_j = R_j$ and R_j is not satisfied at stage $s-1$ and there exists an ℓ with $0 < ht(1^\ell) \leq s$ such that

(i) $1^\ell \in P_j^{Y_{s-1}}$ and

(ii) $ht(1^\ell) \in CH_{s-1}$ and $ht(1^\ell) > n_j^{s-1}$.

Case 3. $A_j = Q_j$ and Q_j is not satisfied at stage $s-1$, and if $j = [e, n]$, there exists an ℓ with $0 \leq ht(1^\ell) < s$ such that

(I) $1^\ell \in N_e^{Y_{s-1}}$,

(II) $ht(1^\ell) \in CH_{s-1}$ and $ht(1^\ell) > max(n, n_j^{s-1})$, and

(III) For each $1^m \notin space(\{1^\ell +_{tal} tal(e_n)\} \cup M_{s-1}) \setminus M_{s-1}$, $ht(1^m) > n_j^{s-1}$ and there is a string α_m of length m in $\{0,1\}^*$ which is not restrained from Y by any requirement of higher priority than Q_j at stage $s-1$ nor is α_m queried in some fixed computation of $N_e^{Y_{s-1}}$ which accepts 1^ℓ.

If there is no such requirement A_j, let $Y_s = Y_{s-1}$. Also for all requirements A_j of the form S_j or Q_j and for all requirements A_j of the form R_j where either R_j is satisfied at stage $s-1$ or $j > b_{s+1}$, let $res(A_j, s) = res(A_j, s-1)$. Declare that a requirement R_j is active at stage s iff R_j is active at stage $s-1$. For any R_j with $j \leq b_{s+1}$ which is currently passive and has the property that $P_j^{Y_s}(1^r) = 0$ for all $k^s \leq r \leq k^{s+1} - 1$, let $res(R_j, s)$ equal $res(R_j, s-1)$ union the set of all $y \notin Y_s$ such that y is queried of the oracle in one of the computations $P_j^{Y_s}(1^r)$ where $k^s \leq r \leq k^{s+1} - 1$.

If there is such a requirement A_j, we have three cases.

Case 1. $A_j = S_{j_s}$.
Let ℓ_s denote the least ℓ corresponding to S_{j_s}. Then for each $1^n \in space(\{1^{\ell_s}\} \cup M_{s-1}) \setminus M_{s-1}$, pick the least string α_n such that $|\alpha_n| = n$, α_n is not restrained from Y by any requirement of higher priority than S_{j_s} at stage $s-1$, nor is α_n queried of the oracle Y_{s-1} in the computation of $N_j^{Y_{s-1}}$ which accepts 1^{ℓ_s}, and put α_n into Y. This will ensure that if M_{s-1} is a finite dimensional subspace of V_∞, then M_s will also be a finite dimensional subspace of V_∞. Note that the assumption that $ht(1^{\ell_s}) \in CH_{s-1}$ ensures that all $1^n \in space(\{1^{\ell_s}\} \cup M_{s-1}) \setminus M_{s-1}$ have the property that $ht(1^n) \geq ht(1^{\ell_s})$. That is, such a 1^n must be of the form $1^n = \lambda \cdot_{tal} 1^{\ell_s} +_{tal} m$ where $m \in M_{s-1}$ and $\lambda \in F$. Then since $ht(m) \neq ht(1^{\ell_s})$, it must be the case that $ht(1^n) \geq ht(1^{\ell_s})$. Thus $ht(M_s) \cap \{n_1^{s-1}, \ldots, n_{j_s}^{s-1}\} = \emptyset$ and hence for all $i \leq j_s$, $n_i^{s-1} = n_i^s$. Let $res(S_{j_s}, s)$ equal the set of all strings not in Y_{s-1} which are queried of the oracle Y_{s-1} in the computation of $N_{j_s}^{Y_{s-1}}$ which accepts 1^{ℓ_s}, and say S_{j_s} receives attention at stage s. Also for all requirements A_j of the form S_j or Q_j and for all requirements A_j of the form R_j where either R_j is satisfied at stage $s-1$ or $j > b_{s+1}$, let $res(A_j, s) = res(A_j, s-1)$ if $Y_s \cap res(A_j, s-1) = \emptyset$ and let $res(A_j, s) = \emptyset$ if $Y_s \cap res(A_j, s-1) \neq \emptyset$. Declare that a requirement R_j is active at stage s iff R_j is active at stage $s-1$ and $Y_s \cap res(R_j, s-1) = \emptyset$. For any R_j with $j \leq b_{s+1}$ which is currently passive and has the property that $P_j^{Y_s}(1^r) = 0$ for all $k^s \leq r \leq k^{s+1} - 1$, let $res(R_j, s)$ equal $res(R_j, s-1)$ union the set of all $y \notin Y_s$ such that y is queried of the oracle Y_s in one of the computations $P_j^{Y_s}(1^r)$, where $k^s \leq r \leq k^{s+1} - 1$.

Case 2. $A_j = R_{j_s}$.
Let ℓ_s denote the least ℓ corresponding to j_s and $n_s = ht(1^{\ell_s})$. We then say that R_{j_s} is active and receives attention at stage s. We let $Y^s = Y^{s-1}$ and $res(R_{j_s}, s)$ consist of all elements y with $k^{n_s - 1} \leq |y| \leq k^{n_s} - 1$ and all elements which are not in Y_{s-1} and which are queried of the oracle Y_{s-1} in the computation $P_{j_{s-1}}^{Y_{s-1}}(1^{\ell_s})$. Note that if $res(R_{j_s}, s) \cap Y = \emptyset$, then M will have no elements of height $n_s = ht(1^{\ell_s})$ but $1^{\ell_s} \in P_{j_s}^Y$. Also for all requirements A_j of the form S_j or Q_j and for all requirements A_j of the form R_j where $j \neq j_s$ and where

either R_j is satisfied at stage $s-1$ or $j > b_{s+1}$, let $res(A_j, s) = res(A_j, s-1)$. For $j \neq j_s$, declare that a requirement R_j is active at stage s iff R_j is active at stage $s-1$. For any R_j with $j \leq b_{s+1}$ which is currently passive and has the property that $P_j^{Y_s}(1^r) = 0$ for all $k^s \leq r \leq k^{s+1} - 1$, let $res(R_j, s)$ equal $res(R_j, s-1)$ union the set of all $y \notin Y_s$ such that y is queried of the oracle Y_s in one of the computations $P_j^{Y_s}(1^r)$ where $k^s \leq r \leq k^{s+1} - 1$.

Case 3. $A_j = Q_{j_s}$.
Let $j_s = [e_s, n_s]$ and ℓ_s denote the least ℓ corresponding to j_s. Then for each $1^m \in space(\{1^{\ell_s} +_{tal} tal(e_{n_s})\} \cup M_{s-1}) \setminus M_{s-1}$, pick the least string α_m such that $|\alpha_m| = m$, and α_m is not restrained from Y by any requirement of higher priority than Q_{j_s} at stage $s-1$ nor is α_m queried in the computation of $N_{e_s}^{Y_{s-1}}$ which accepts 1^{ℓ_s} and put α_m into Y. Once again this will ensure that M_s is a finite dimensional subspace of V_∞. Note that since $ht(1^{\ell_s}) > n_s = ht(tal(e_{n_s}))$, it follows that $ht(1^{\ell_s} +_{tal} tal(e_{n_s})) = ht(1^{\ell_s})$. Thus as in case 1, the assumption that $ht(1^{\ell_s}) \in CH_{s-1}$ ensures that all $1^n \in space(\{1^{\ell_s} +_{tal} tal(e_{n_s})\} \cup M_{s-1}) \setminus M_{s-1}$ have the property that $ht(1^n) \geq ht(1^{\ell_s})$. Let $res(Q_{j_s}, s)$ equal the set of all strings which are not in Y_{s-1} which are queried of the oracle in the computation of $N_{e_s}^{Y_{s-1}}$ which accepts 1^{ℓ_s} and say Q_{j_s} receives attention at stage s. Also for all requirements A_j of the form S_j or Q_j and for all requirements A_j of the form R_j where either R_j is satisfied at stage $s-1$ or $j > b_{s+1}$, let $res(A_j, s) = res(A_j, s-1)$ if $Y_s \cap res(A_j, s-1) = \emptyset$ and let $res(A_j, s) = \emptyset$ if $Y_s \cap res(A_j, s-1) \neq \emptyset$. Declare that a requirement R_j is active at stage s iff R_j is active at stage $s-1$ and $Y_s \cap res(R_j, s-1) = \emptyset$. For any R_j with $j \leq b_{s+1}$ which is currently passive and has the property that $P_j^{Y_s}(1^r) = 0$ for all $k^s \leq r \leq k^{s+1} - 1$, let $res(R_j, s)$ equal $res(R_j, s-1)$ union the set of all $y \notin Y_s$ such that y is queried of the oracle Y_s in one of the computations $P_j^{Y_s}(1^r)$, where $k^s \leq r \leq k^{s+1} - 1$.

This completes the construction of Y.

Lemma 8.48 *Each requirement of the form S_j, R_j, or Q_j receives attention at most finitely often.*

Proof: We proceed by induction on j. Suppose that s_0 is such that there is no stage $s \geq s_0$ such that one of $S_0, R_0, Q_0, \ldots, S_j, R_j, Q_j$ receives attention at stage s. Then if there is a $t > s_0$ such that S_{j+1} receives attention at stage t, then by construction S_{j+1} is satisfied at stage t and $res(S_{j+1}, t) \cap Y_t = \emptyset$. However it is easy to see from our construction that for $s > t$, $res(S_{j+1}, s) = res(S_{j+1}, t)$ and $res(S_{j+1}, s) \cap Y_s = \emptyset$ unless some requirement of higher priority than S_{j+1} receives attention at stage s. Since this never happens by our choice of s_0, S_{j+1} will be satisfied for $s > t$. Thus S_{j+1} can receive attention at most once after stage s_0. Thus there must be a stage s_1 such that there is no stage $s \geq s_1$

such that one of $S_0, R_0, Q_0, \ldots, S_j, R_j, Q_j, S_{j+1}$ receives attention at stage s. A similar argument will show that R_{j+1} can receive attention at most once after stage s_1. Thus there must be a stage s_2 such that there is no stage $s \geq s_2$ such that one of $S_0, R_0, Q_0, \ldots, S_j, R_j, Q_j, S_{j+1}, R_{j+1}$ receives attention at stage s Finally a similar argument will show that Q_{j+1} can receive attention at most once after stage s_2. Thus each of the requirements S_j, R_j, or Q_j can receive attention only finitely often. $\Box$

Lemma 8.49 *$dim(tal(V_\infty)/M)$ is infinite.*

Proof: We prove by induction that $dim(tal(V_\infty)/M) \geq k$ for all k. That is, let t_0 be a stage such that no requirement $S_0, R_0, Q_0, \ldots, S_k, R_k, Q_k$ receives attention at any stage $s \geq t_0$. Since M_{t_0} is finite dimensional, $n_i^{t_0}$ is defined for all i. Hence M_t contains no strings of height n for $n = n_1^{t_0}, \ldots, n_k^{t_0}$. But no requirement S_j, R_j, or Q_j with $j > k$ can force elements of height $n \leq n_k^s$ into M at any stage s. Hence by our choice of t_0, there can be no strings of heights n for $n = n_1^{t_0}, \ldots, n_k^{t_0}$ in M. Thus $dim(tal(V_\infty)/M) \geq k$. $\Box$

Lemma 8.50 *M is P^Y-simple.*

Proof: First we show that if N_j^Y is a subspace of $tal(V_\infty)$ such that $ht(N_j^Y) \setminus ht(M)$ is infinite, then $N_j^Y \cap M \neq \{\vec{0}\}$. For a contradiction assume N_j^Y is such that $ht(N_j^Y) \setminus ht(M)$ is infinite and $N_j^Y \cap M = \{\vec{0}\}$. Note that since M is co-infinite dimensional by Lemma 8.49, it follows that $n_i = lim_{s \to \infty} n_i^s$ exists for all i. Let s_0 be a stage large enough so that $n_i^s = n_i$ for $i \leq j$ and none of the requirements $S_0, R_0, Q_0, \ldots, S_{j-1}, R_{j-1}, Q_{j-1}$ receives attention after stage s_0. Let U_{s_0} denote the set of all 1^n such that there exists a requirement A_i among $S_0, R_0, Q_0, \ldots, S_{j-1}, R_{j-1}, Q_{j-1}$ which is satisfied at stage s_0, such that there exists an $\alpha \in res(A_i, s_0)$ with $|\alpha| = n$. Our choice of s_0 ensures that if $n \notin U_{s_0}$, then no string α of length n is ever restrained from Y by a requirement of higher priority than S_j which is satisfied at some stage $t > s_0$. Also our choice of s_0 ensures that $n_i = n_i^t$ for all $i \leq j$ and $t > s_0$. Next let $t_0 > s_0$ be such that

1. $t_0 > max(\{ht(y) : y \in U_{s_0}\} \cup \{2, s_0, n_j\})$,
2. $b_{t_0} > j$, and
3. $2^{r-1} > r^j$ for all $r > t_0$.

Note that for any $t > t_0$, our construction ensures that the number of strings of length r where $k^{t-1} \leq r \leq k^t - 1$ which are restrained by some requirement R_i with $i < j$ which is passive at stage t is less than 2^{r-1}. Moreover we are assuming that any successful computation of the oracle machine N_j^X for any oracle X on a string of length $r > 2$ takes at most r^j steps. Thus our choice of t_0 ensures that if $t > t_0$ and $1^x \in N_j^{Y_t}$ is string of height $> t_0$, then there is at least one string $\alpha_x \in \{0,1\}^*$ of length x which is not restrained from

Y by any requirement of higher priority than S_j at stage t, nor is queried of the oracle Y_t in some fixed computation which shows that $1^x \in N_j^{Y_t}$. Since $ht(N_k^Y) \setminus ht(M)$ is infinite, there must exist a $1^n \in N_k^Y$ such that $ht(1^n) > t_0$ and $ht(1^n) \notin ht(M)$. Then there must be some stage $s > t_0$ such that $1^n \in N_j^{Y_{s-1}}$. Note that at stage s, each $1^m \in space(\{1^n\} \cup M_{s-1}) \setminus M_{s-1}$ has the property that $ht(1^m) \geq ht(1^n) > t_0$ and thus there is at least one string α_m of length m which is not restrained from Y by any requirement of higher priority than S_j at stage $s-1$, nor is queried of the oracle Y_{s-1} in some fixed computation which shows that $1^n \in N_j^{Y_{s-1}}$. Thus 1^n witnesses that S_j is a candidate to receive attention at stage s. Thus either S_j is satisfied at stage $s-1$ or S_j is highest priority requirement among $S_0, R_0, Q_0, \ldots, S_s, R_s, Q_s$ which can receive attention at stage s. In either case, it follows that S_j will be satisfied at stage s. Thus there will be some $1^n \in (N_j^{Y_s} \cap M_s) \setminus \{\vec{0}\}$ such that all elements which are queried of the oracle Y_s in some computation which shows that $1^n \in N_j^{Y_s}$, and which are not in Y_s, are in $res(S_j, s)$. However our choice of t_0 ensures that we can never put any element of $res(S_j, s)$ into Y after stage s so that 1^n will witness that $N_j^Y \cap M \neq \{\vec{0}\}$.

Remark. We note that the assumption that $ht(N_j^Y) \setminus ht(M)$ is infinite seems to be crucial in this argument. That is, if we merely assume that $dim(N_j^Y/M)$ is infinite, then it may be the case that whenever there exists a $1^n \in N_k^Y$ such that $ht(1^n) > t_0$ and $1^n \notin M$, then at a stage $s > t_0$ where $1^n \in N_k^{Y_{s-1}}$, there is some $1^m \in M_{s-1}$ such that $hy(1^m) = ht(1^n)$. In such a situation it is possible that $ht(1^n +_{tal} 1^m)$ is much less than $ht(1^n)$. That is, it may be possible that some element in $1^x \in space(\{1^n\} \cup M_{s-1}) \setminus M_{s-1}$ has height so small that all strings of length x are queried of the oracle during any computation which shows that $1^n \in N_k^{Y_{s-1}}$. Then it will be impossible to put a string of length x into Y_s so as to ensure that $1^x \in M_s$ while maintaining the computation to ensure that $1^n \in N_k^Y$.

To continue our proof of the lemma, we can now assume that if P_r^Y is an infinite dimensional subspace of $tal(V_\infty)$ such that $P_r^Y \cap M = \{\vec{0}\}$, then $ht(P_r^Y) \setminus ht(M)$ is finite. By our argument preceding the construction, it would then follow that there is some j such that P_j^Y is an infinite dimensional subspace of $tal(V_\infty)$ and $ht(P_j^Y) \subseteq ht(M)$. We shall now show that there can be no such j. For a contradiction, assume that P_j^Y is an infinite dimensional subspace of $tal(V_\infty)$ and $ht(P_j^Y) \subseteq ht(M)$. Let s_1 be a stage large enough so that $n_i^{s_1} = n_i$ for $i \leq j$ and none of the requirements $S_0, R_0, Q_0, \ldots, S_{j-1}, R_{j-1}, Q_{j-1}, S_j$ receives attention after stage s_1. Let U_{s_1} denote the set of all 1^n such that there exists a requirement A_i among $S_0, R_0, Q_0, \ldots, S_{j-1}, R_{j-1}, Q_{j-1}, S_j$ which is satisfied at stage s_1 and there exists an $\alpha \in res(A_i, s_1)$ with $|\alpha| = n$. Our choice of s_1 ensures that if $n \notin U_{s_1}$, then no string α of length n is ever restrained from Y by a requirement of higher priority than R_j which is satisfied at some stage $t > s_0$. Also our choice of s_1 ensures that $n_i = n_i^t$ for all $i \leq j$ and $t > s_0$.

Next let t_1 be such that

1. $t_1 > max(\{ht(y) : y \in U_{s_1}\} \cup \{2, s_1, n_{i-1}\})$,
2. $b_{t_1} > j$, and
3. $2^{r-1} > r^j$ for all $r > t_1$.

Now we claim that there can be no stage $t > t_1$ at which R_j is satisfied at stage t. That is, if R_j is satisfied at stage t, there must be some $s \leq t$ such that R_j receives attention at stage s, and there is a $1^x \in P_j^{Y_{s-1}}$ such that $q = ht(1^x) \in CH_{s-1}$, and $res(R_j, s) = res(R_j, t)$ contains all strings of length r where $k^{q-1} \leq r \leq k^q - 1$, and contains all strings which are not in Y_{s-1} which are queried of the oracle Y_{s-1} in the computation $P_j^{Y_{s-1}}(1^x) = 1$, and $res(R_j, s) \cap Y_t = \emptyset$. But then our choice of $t > t_1$ ensures that $res(R_j, s) \cap Y = \emptyset$, which means that M can have no strings of height q while $1^x \in P_j^Y$. But then 1^x witnesses that $ht(P_j^Y) \not\subseteq ht(M)$ which contradicts our assumption that $ht(P_j^Y) \subseteq ht(M)$. Thus it must be the case that for all stages $t > t_1$, R_j is in a passive state. It follows that for all $t > t_1$, there can be no r with $k^t \leq r \leq k^{t+1} - 1$ such that $P_j^{Y_t}(1^r) = 1$ since otherwise at stage $t+1$, there is some r with $k^t \leq r \leq k^{t+1} - 1$ such that $P_j^{Y_t}(1^r) = 1$. But then at stage $t+1$, 1^r witnesses that R_j is a candidate to receive attention at stage $t+1$. By our choice of $t > t_1$, it would follow that R_j is the highest priority requirement among $S_0, R_0, Q_0, \ldots, S_{t+1}, R_{t+1}, Q_{t+1}$ which could receive attention at stage $t+1$ so that R_j would receive attention at stage $t+1$ which we have already ruled out. Thus it must be the case that for all r with $k^t \leq r \leq k^{t+1} - 1$, $P_j^{Y_t}(1^r) = 0$. But then our choice of $t > t_1$ ensures that $j \leq b_{t+1}$ and hence all elements which are not in Y_t which are queried of the oracle Y_t during one of the computations $P_j^{Y_t}(1^r) = 0$ where $k^t \leq r \leq k^{t+1} - 1$ are put into $res(R_j, t)$. Again the fact that $t > t_1$ ensures that $res(R_j, t) \cap Y = \emptyset$ so that for all r with $k^t \leq r \leq k^{t+1} - 1$, $P_j^Y(1^r) = 0$. That is, P_j^Y has no strings of length $t+1$ for any $t > t_1$ and hence $ht(P_j^Y)$ is finite. Thus there can be no such P_j^Y such that P_j^Y is an infinite dimensional subspace of $tal(V_\infty)$ and $ht(P_j^Y) \subseteq ht(M)$. But this means that there can be no r such that P_r^Y is an infinite dimensional subspace of $tal(V_\infty)$ and $P_r^Y \cap M = \{\vec{0}\}$. Thus M is P^Y-simple as claimed. $\square$

Lemma 8.51 *M is NP^Y-maximal.*

Proof: By our remarks preceding the construction, we need only show that we meet all the requirements $Q_{[e,n]}$. So assume N_e^Y is a subspace of $tal(V_\infty)$ such that (N_e^Y/M) is infinite dimensional and $N_e^Y \supseteq M$. Let $j = [e, n]$ and let s_2 be a stage such that $n_i = n_i^{s_2}$ for $i \leq j$ and none of the requirements $S_0, R_0, Q_0, \ldots, S_{j-1}, R_{j-1}, Q_{j-1}, S_j, R_j$ receive attention after stage s_2. Let U_{s_2} denote the set of all 1^n such that there exists a requirement A_i among $S_0, R_0, Q_0, \ldots, S_{j-1}, R_{j-1}, Q_{j-1}, S_j, R_j$ which is satisfied at stage s_2 and there exists an $\alpha \in res(A_i, s_2)$ with $|\alpha| = n$. Our choice of s_2 ensures that if $n \notin U_{s_2}$, then no string α of length n is ever restrained from Y by a requirement of higher priority than Q_j which is satisfied at some stage $t > s_2$. Also our choice of s_2 ensures that $n_i = n_i^t$ for all $i \leq j$ and $t > s_2$. Next let t_2 be such that

1. $t_2 > max(\{ht(y) : y \in U_{s_2}\} \cup \{2, s_2, n_{i-1}\})$,
2. $b_{t_2} > j$, and
3. $2^{r-1} > r^j$ for all $r > t_2$.

Note that for any $t > t_2$, our construction ensures that the number of strings of length r, where $k^{t-1} \leq r \leq k^t - 1$, which are restrained by some requirement R_i with $i < j$ which is passive at stage t, is less than 2^{r-1}. Moreover we are assuming that any successful computation of the oracle machine N_j^X for any oracle X on a string of length $r \geq 2$ takes at most r^j steps. Thus our choice of t_2 ensures that if $t > t_2$ and $1^x \in N_j^{Y_t}$ is string of height $> t_2$, then there is at least one string $\alpha_x \in \{0,1\}^*$ of length x which is not restrained from Y by any requirement of higher priority than Q_j at stage t, nor is queried of the oracle Y_t in some fixed computation which shows that $1^x \in N_j^{Y_t}$.

Next observe that since $dim(N_e^Y/M)$ is infinite and $N_e^Y \supseteq M$, it must be the case that $ht(N_e^Y) \setminus ht(M)$ is infinite. That is, let $A = \{a_0, a_1, \ldots\}$ be an infinite set of elements of N_e^Y which is independent over M. Then consider some fixed $a_i \in A$ and suppose $a_i = \sum_{i=1}^q \lambda_i \cdot tal(e_{j_i})$ where $\lambda_i \in F$ for $i = 1, \ldots, q$, $\lambda_q \neq 0$, and $j_1 < \ldots < j_q$. Thus $ht(a_i) = j_q$. Now if there exists an $m_1 \in M$ such that $ht(m) = ht(a_i)$, then $m_1 = \sum_{\ell \leq j_q} \beta_\ell \cdot tal(e_\ell)$ where $\beta_\ell \in F$ for all ℓ and $\beta_{j_q} \neq 0$. But then $a_i^1 = a_i -_{tal} \frac{\lambda_{j_q}}{\beta_{j_q}} m_1$ is an element of $N_e^Y \setminus M$ such $ht(a_i^1) < ht(a_i)$. Now if there exists an $m_2 \in M$ such that $ht(a_i^1) = ht(m_2)$, then once again there is some $\gamma \in F$ such that $a_i^2 = a_i^1 -_{tal} \gamma \cdot m_2$ is an element of $N_e^Y \setminus M$ with $ht(a_i^2) < ht(a_i^1) < ht(a_i)$. If we continue in this fashion, we must eventually find some $a_i^k = a_i +_{tal} v_k$ where $v_k \in M$ such that $ht(a_i^k) \notin ht(M)$. That is, we can replace our original independent set A over M by a set $A' = \{a_0', a_1', \ldots\}$ where for all i, $a_i -_{tal} a_i' \in M$ and $ht(a_i') \notin ht(M)$. But then A' is an infinite subset of N_e^Y which is independent over M. Thus there is no finite set F such that $space(M \cup F) \supseteq A'$. This implies that $ht(A') = \{ht(a_i') : i \geq 0\}$ must be infinite, since otherwise there clearly would be a finite set F such that $space(M \cup F) \supseteq A'$. But by construction $ht(A') \subseteq ht(N_e^Y) \setminus ht(M)$ so that $ht(N_e^Y) \setminus ht(M)$ must be infinite.

Since $ht(N_e^Y) \setminus ht(M)$ is infinite, there must exist a $1^q \in N_e^Y$ such that $ht(1^q) > t_2$, $ht(1^q) > n$, and $ht(1^q) \notin ht(M)$. Then there must be some stage $s > t_2$ such that $1^q \in N_e^{Y_{s-1}}$. Note that at stage s, each $1^m \in space(\{1^q +_{tal} tal(e_n)\} \cup M_{s-1}) \setminus M_{s-1}$ has the property that $ht(1^m) \geq ht(1^q +_{tal} tal(e_n)) = ht(1^q) > t_2$, and thus there is at least one string α_m of length m which is not restrained from Y by any requirement of higher priority than Q_j at stage $s-1$,

nor is queried of the oracle Y_{s-1} in some fixed computation which shows that $1^q \in N_j^{Y_{s-1}}$. Thus 1^q witnesses that Q_j is a candidate to receive attention at stage s. Hence either Q_j is satisfied at stage $s-1$ or Q_j is highest priority requirement among $S_0, R_0, Q_0, \ldots, S_s, R_s, Q_s$ which can receive attention at stage s. In either case, it follows that Q_j will be satisfied at stage s. Thus there will be some $1^q \in N_j^{Y_s}$ such that $1^q +_{tal} tal(e_n) \in M_s$ and all elements which are queried of the oracle in some computation which shows that $1^q \in N_j^{Y_s}$ and which are not Y_s, are in $res(Q_j, s)$. However our choice of t_2 ensures we can never put any element of $res(Q_j, s)$ into Y after stage s so that $1^q \in N_j^Y$ and hence requirement Q_j is meet. Thus M will be NP^Y-supermaximal and hence will be NP^Y-maximal. This completes the proof of Lemma 8.51 and of Theorem 8.47
□ □

We note that M constructed in Theorem 8.47 has a number of interesting properties besides being NP^Y-maximal and P^Y-simple. First of all, it is easy to check that in meeting the requirements S_j, we made no use of the fact that N_j^Y was a subspace of $tal(V_\infty)$, but only that N_j^Y was a subset of $tal(V_\infty)$. Similarly, it is easy to check that in meeting the requirements R_j we made no use of the fact that P_j^Y was a subspace of $tal(V_\infty)$, but only that P_j^Y was a subset of V_∞. Thus meeting all the requirements R_j ensures that there is no infinite subset W of $tal(V_\infty)$ in P^Y such that $ht(W) \subseteq ht(M)$. Thus M does not contain any infinite P^Y set and hence M does not have a basis in P^Y. We also claim that $tal(V_\infty) \setminus M$ does not have any infinite subsets in P^Y. That is, suppose that $P_j^Y \subseteq tal(V_\infty) \setminus M$. Now it cannot be that $ht(P_j^Y) \setminus ht(M)$ is infinite, since otherwise there is an i such that $P_j^Y = N_i^Y$ and the fact that we met requirement S_i would mean that $P_j^Y \cap M \neq \{\vec{0}\}$. Thus $ht(P_j^Y) \subseteq^* ht(M)$. Let $Q = ht(space(A)) \setminus ht(P_j^Y)$. Then clearly

$$S = \{x \in P_j^Y : ht(x) \notin Q\}$$

is an infinite set in P^Y such that $ht(S) \subseteq ht(space(A))$. Since meeting all the requirements R_j rules out the existence of such an S, $tal(V_\infty) \setminus M$ does not contain an infinite set in P^Y. Recall that a set of strings S is called P^Y-**immune** if S has no infinite subset in P^Y. Thus both M and $tal(V_\infty) \setminus M$ are P^Y-immune

Note also that by Theorem 8.40, the fact that M is NP^Y-maximal implies that $NP^Y \neq$ co-NP^Y and hence that $P^Y \neq NP^Y$. Thus we have proved the following.

Corollary 8.52 *There exists an r.e. oracle Y and a subspace M of $tal(V_\infty)$ such that*

(1) $P^Y \neq NP^Y$ *and* $NP^Y \neq$ *co-*NP^Y*,*

(2) $M \in NP^Y$*,*

(3) *M is P^Y-immune and hence has no basis in P^Y,*

(4) *$tal(V_\infty) \setminus M$ is P^Y-immune, and*

(5) *M is both P^Y-simple and NP^Y-supermaximal.*

We next give an analogue of Theorem 8.47 for $st(V_\infty)$. Once again we shall think of $st(V_\infty)$ as the k-ary representation $B_k(V_\infty)$ so that for all $x \in st(V_\infty)$, $|x| = ht(x)$.

Theorem 8.53 . *There exists a r.e. oracle D such that there exists an NP^D-supermaximal P^D-simple subspace in $st(V_\infty)$.*

Proof: The construction again proceeds in stages. We let D_s be the set of elements enumerated into D by the end of stage s. For any given $x \in \{0, \ldots, k-1\}^*$ with $|x| \geq 1$, we let C_x denote the set of all strings of length $8|x|+2$ of $\{0, \ldots, k-1\}^*$ of the form $x10^{4|x|}1\sigma$ where σ is any string of length $3|x|$ in $\{0, \ldots, k-1\}^*$. Note that there are $k^{3|x|}$ strings in C_x for any $x \in st(V_\infty)$. Let $C_\emptyset = \{\emptyset\}$. It is then easy to see that if $x \neq y$, then $C_x \cap C_y = \emptyset$

We then define $A = \{x : C_x \cap D \neq \emptyset\}$. Thus A will be in NP^D. Our idea is to define D so that A is a height increasing independent subset of $st(V_\infty)$. Then by the relativized version of Theorem 8.28, $space(A) \in NP^D$. Our construction of D will ensure that $space(A)$ is our desired P^D-simple NP^D-supermaximal space. Let $A_s = \{x : C_x \cap D_s \neq \emptyset\}$. At each stage s, we shall let $B_s = \{st(e_n) :$ A_s has no element of height $n\}$. Our construction will ensure that at each stage s, $A_s \cup B_s$ is a height increasing basis of $st(V_\infty)$. We define b_i^s for all i and s so that $B_s = \{b_0^s, b_1^s, \ldots\}$ where $ht(b_0^s) < ht(b_1^s) < \ldots$.

To ensure that $space(A)$ is co-infinite dimensional we must meet the following set of requirements.

$$T_j : card(\{n : D \text{ contains no strings } \alpha \text{ with } |\alpha| = 8n+2\}) \geq j$$

Thus T_j says there are at least j heights n so that A contains no strings of height n. So meeting requirement T_j ensures $\dim(V_\infty/space(A)) \geq j$.

To ensure that $space(A)$ is P^D-simple, we shall meet the following two sets of requirements.

$$S_j : \text{If } N_i^D \text{ is an infinite dimensional subspace of } st(V_\infty) \text{ such that } ht(N_i^D) \setminus ht(space(A)) \text{ is infinite, then } space(A) \cap N_i^D \neq \{\vec{0}\}.$$

Now suppose that P_j^D generates an infinite dimensional subspace of $st(V_\infty)$ which is in NP^D. Note that meeting all the requirements S_j will ensure that either $space(P_i^D) \cap space(A) \neq \{\vec{0}\}$ or $ht(space(P_i^D)) \subseteq^* ht(space(A))$. Now suppose that $ht(space(P_i^D)) \subseteq^* ht(space(A))$ and let $U = ht(P_i^D) \setminus ht(space(A))$. If $U = \emptyset$, then $ht(P_i^D) \subseteq ht(space(A))$. Otherwise, U is a finite set so let $U = \{n_0, \ldots, n_q\}$ and let $x_0, \ldots, x_q$ be elements of $space(P_i^D)$ such that $ht(x_i) =$

n_i. Note that any $x \in st(V_\infty)$ is a string of the form $x = a_1 \cdots a_{|x|}$ where $a_j \in \{0, \ldots, k-1\}$. Then we define the full height of x, $fh(x) = \{n : 1 \leq n \leq |x| \text{ and } a_n \neq 0\}$. Then it is easy to see that given any $x \in space(P_i^D)$, there exists some $\lambda_1, \ldots, \lambda_q$ in F such that $fh(x -_{st} \sum_{i=1}^{q} \lambda_i x_i) \cap U = \emptyset$. That is, if $x = a_1 \cdots a_{|x|}$ where $|x| \geq n_q$ and $a_{n_q} \neq 0$ and $x_q = a_{1,q} \cdots a_{n_q,q}$ where $a_{n_q,q} \neq 0$, then $x' = x -_{st} \frac{a_q}{a_{n_q,q}} x_q = b_0 \cdots b_{|x|}$ where $b_{n_q} = 0$ so that $n_q \notin fh(x')$. Now if $b_{n_{q-1}} \neq 0$ and $x_{q-1} = a_{1,q-1} \cdots a_{n_{q-1},q-1}$ where $a_{n_{q-1},q-1} \neq 0$, then $x'' = x' -_{st} \frac{b_q}{a_{n_{q-1},q-1}} x_{q-1} = c_0 \cdots c_{|x|}$ where $c_{n_q} = b_{n_q} = 0$ and $c_{n_{q-1}} = 0$ so that neither n_q nor n_{q-1} is in $fh(x'')$. Continuing on in this way we can construct our desired linear combination $\sum_{i=1}^{q} \lambda_i x_i$ such that $fh(x -_{st} \sum_{i=1}^{q} \lambda_i x_i) \cap U = \emptyset$. Now let $Q = \{x \in space(P_i^D) : fh(x) \cap U = \emptyset\}$. It is easy to see that Q is a subspace of P_i^D and our argument above shows that $space(P_i^D) = space(\{x_1, \ldots, x_q\}) \oplus Q$. Thus Q is an infinite dimensional subspace of $st(V_\infty)$ such that $ht(Q) \subseteq ht(space(A))$. Let T be the set of all y such that $fh(y) \cap U = \emptyset$, $|y| > k^{|x_q|}$, and there exists an $x \in P_i^D$ and $z \in space(\{x_1, \ldots, x_q\})$ such that $x +_{st} z = y$. Note that $space(\{x_1, \ldots, x_q\})$ has exactly k^q elements since $\{x_1, \ldots, x_q\}$ is a height increasing basis for $space(\{x_1, \ldots, x_q\})$. Thus given any y with $|y| > k^{|x_q|}$, in polynomial time in $|y|$, we can find all $y +_{st} w$ such that $w \in space(\{x_1, \ldots, x_q\})$. Now for any $w \in space(\{x_1, \ldots, x_q\})$, $ht(w) \leq ht(x_q) = |x_q| < k^{|x_q|}$ so that $ht(y +_{st} w) = ht(y)$. Thus it take at most $k^q(|y|^j)$ steps to test all such $y +_{st} w$ for membership in P_j^D given an oracle D. But then

$$y \in T \text{ iff } \{y +_{st} w : w \in space(\{x_1, \ldots, x_q\})\} \cap P_j^D \neq \emptyset.$$

Thus it follows that T is in P^D and clearly T generates an infinite dimensional subspace of Q. Thus there must be some j such that P_j^D generates an infinite dimensional subspace of $st(V_\infty)$ and $ht(space(P_j^D)) \subseteq ht(space(A))$. Thus to ensure that $space(A)$ is P^D simple, it will be enough to ensure that we meet the following set of requirements.

R_i: If P_i^D generates an infinite dimensional subspace of $st(V_\infty)$, then $ht(P_i^D) \not\subseteq ht(space(A))$.

Finally, to ensure that $space(A)$ is NP-supermaximal, we shall meet the following set of requirements.

$Q_{[i,n]}$: If $N_i^D/space(A)$ is an infinite dimensional and $N_i^D \supseteq space(A)$, then there is an $x \in N_i^D$ such that $x + st(e_n) \in space(A)$.

Note that if $N_i^D \supseteq space(A)$ and $\dim(N_i^D/space(A))$ is infinite, then meeting all the requirements $Q_{[i,n]}$ will ensure that $st(e_n) \in N_i^D$ for all n so that $N_i^D = st(V_\infty)$.

We shall rank our requirements with those of highest priority coming first as $T_0, S_0, R_0, Q_0, T_1, S_1, R_1, Q_1, \ldots$.

As in the construction of Theorem 8.47, at any given stage s, we shall pick out at most one requirement E_j where E_j will be one of the requirements S_j, R_j, or Q_j and take an action to meet that requirement. We shall then say that E_j *received attention* at stage s. The action that we take to meet the requirement E_j of the form S_j or Q_j will always be of the same form. That is, we shall put some elements into D at stage s and possibly restrain some elements from entering D for the sake of the requirement. We shall let $res(E_j, s)$ denote the set of elements that are restrained from entering D at stage s for the sake of requirement E_j. We say that requirement E_j of the form S_j or Q_j is **satisfied** at stage s, if there is a stage $s' < s$ such that E_j has received attention at stage s' and $res(E_j, s') \cap D_s = \emptyset$.

The actions that we take to meet the requirements R_j will essentially the same as in the construction of Theorem 8.47. First, we shall declare that all R_j are in a *passive* state at the start of our construction. We would like to find an element $x \in P_j^{D_s}$ of height n such that $n \notin ht(space(A_s))$. If we can find such an x, then we will restrain all y such that $|y| = 8n + 2$ and $y \in C_x$ for some $x \in st(V_\infty)$ of height n plus all elements not in D_s which are queried of the oracle during the computation of $P_j^{D_s}(x)$ from entering D for the sake of requirement R_j. Then if we ensure that $res(R_j, s) \cap D = \emptyset$, A will have no elements of height n and $x \in P_j^D$ so that $ht(P_j^D) \not\subseteq ht(space(A))$. If we take such an action for R_j at stage s, then we will say that R_j has received attention at stage s and declare the state of R_j to be *active*. Then for all $t > s$, we will say that an active R_j is **satisfied** at stage t, if $res(R_j, s) \cap D_t = \emptyset$. However if R_j is injured at some stage $t > s$ in the sense that $res(R_j, s) \cap D_t \neq \emptyset$, then R_j will return to a passive state. If we cannot find such an x, we will attempt to force $ht(P_j^D)$ to be finite. That is, since we will ensure that $ht(space(A_{s-1})) \subseteq \{0, \ldots, s-1\}$ for all s, A_{s-1} will have no elements of height s. Recall that we are assuming that for $n \geq 2$, the run time of computations of $P_j^X(y)$ for any oracle X is bounded $max(2, n)^j$ for any string of length n. Then for $n \geq 2$, we let d_n be the largest i such that for all r,

$$n^{(i+2)} < k^n.$$

Note that it is easy to see that $lim_{s \to \infty} d_s = \infty$. Our idea is that elements of height n in $st(V_\infty)$ are just the elements of length n. Our strategy at the end of stage $s - 1$ for $s \geq 2$ is that for all R_j with $j \leq d_s$ which are in a passive state and have the property that $P_j^{D_{s-1}}(x) = 0$ for all $x \in st(V_\infty)$ of length s, we will restrain all elements which are not in D_{s-1} and which are queried in such computations from entering D for the sake of R_j. This action will force $ht(P_j^D)$ to be finite if R_j is in a passive state at stage s for all but finitely many s. For any fixed $j \leq d_s$, the maximum restraint imposed for R_j occurs if we restrained all elements not in D_{s-1} which are queried of the oracle D_{s-1} in some computation $P_j^{D_{s-1}}(x) = 0$ with $1 \leq |x| \leq n$ and $x \in st(V_\infty)$. Since the total number of steps used in all these computations is at most

$$2^j + \sum_{i=2}^{s} k^i i^j \leq sk^s \cdot (s)^j = k^s s^{(j+1)},$$

then clearly we could have restrained at most $k^s s^{(j+1)}$ elements from entering D for the sake of R_j. Thus at stage s, we will have restrained at most

$$\sum_{i=0}^{d_s} k^s s^{(i+1)} < k^s s^{(d_s+2)} < k^s k^s = k^{2s}$$

elements from entering D for the sake of some passive requirement R_j with $j \leq b_s$ at stage $s-1$. Hence for any given x with $|x| = n$, we will have restrained less than k^{2s} elements of C_x from entering D for such R_j's.

CONSTRUCTION.

Stages $0, 1$.
Let $D_0 = D_1 = \emptyset$ so that $A_0 = A_1 = \emptyset$. Let $res(E_j, 0) = res(E_j, 1) = \emptyset$ for all requirements E_j of the form S_j, R_j, or Q_j.

Stage s with $s \geq 2$.
Let E_j be the highest priority requirement among $S_0, R_0, Q_0, \ldots, S_s, R_s, Q_s$ such that

Case 1. $E_j = S_j$ and S_j is not satisfied at stage $s-1$ and there exists an $x \in st(V_\infty)$ with $0 < |x| \leq s$ such that

(a) $x \in N_j^{D_{s-1}}$,

(b) $|x| \notin ht(space(A_{s-1}))$ and $|x| > |b_j^{s-1}|$, and

(c) there exists a $y \in C_x$ such that y is not restrained from D by any requirement of higher priority than S_j at stage $s-1$ and y is not queried of the oracle D_{s-1} in some fixed computation which shows that $x \in N_j^{D_{s-1}}$.

Case 2. $E_j = R_j$ and R_j is not satisfied at stage $s-1$ and there exists an $x \in st(V_\infty)$ with $0 \leq |x| \leq s$ such that

(i) $|x| \notin ht(space(A_{s-1})$ and

(ii) $x \in P_j^{D_{s-1}}$.

Case 3. $A_j = Q_j$ and Q_j is not satisfied at stage $s-1$, and if $j = [e, n]$, there exists an x with $0 \leq |x| \leq s$ such that

(I) $x \in N_e^{D_{s-1}}$,

(II) $|x| \notin ht(space(A_{s-1}))$, $|x| > |b_j^{s-1}|$, and $|x| > n$, and

(III) there exists a $y \in C_{x+_{st}st(e_n)}$ such that y is not restrained from D by any requirement of higher priority than S_j at stage $s-1$ and y is not queried of the oracle D_{s-1} in some fixed computation which shows that $N_e^{D_{s-1}}(x)$.

If there is no such requirement E_j, let $D_s = D_{s-1}$. Also for all requirements E_j of the form S_j or Q_j, and for all requirements E_j of the form R_j where either R_j is satisfied at stage $s-1$ or $j > d_{s+1}$, let $res(E_j, s) = res(E_j, s-1)$. Declare that a requirement R_j is active at stage s iff R_j is active at stage $s-1$. For any R_j with $j \leq d_{s+1}$ which is currently passive and has the property that $P_j^{D_s}(x) = 0$ for all $x \in st(V_\infty)$ of length $s+1$, let $res(R_j, s)$ equal $res(R_j, s-1)$ union the set of all $y \notin D_s$ such that y is queried of the oracle D_s in one of the computations $P_j^{D_s}(x)$ where $x \in st(V_\infty)$ of length $s+1$.

If there is such a requirement E_j, we have three cases.

Case 1. $E_j = S_{j_s}$.
Let x_s denote the least x corresponding to S_{j_s}. Then pick the least string $\alpha_{x_s} \in C_{x_s}$ such that α_{x_s} is not restrained from D by any requirement of higher priority than S_{j_s} at stage $s-1$, nor is α_{x_s} queried of the oracle D_{s-1} in the computation of $N_j^{D_{s-1}}$ which accepts x_s, and put α_{x_s} into D. Let $res(S_{j_s}, s)$ equal the set of all strings not in D_{s-1} which are queried of the oracle D_{s-1} in the computation of $N_{j_s}^{D_{s-1}}$ which accepts x_s, and say S_{j_s} receives attention at stage s. Also for all requirements E_j of the form S_j or Q_j, and for all requirements E_j of the form R_j where either R_j is satisfied at stage $s-1$ or $j > d_{s+1}$, let $res(E_j, s) = res(E_j, s-1)$ if $D_s \cap res(E_j, s-1) = \emptyset$ and let $res(E_j, s) = \emptyset$ if $D_s \cap res(E_j, s-1) \neq \emptyset$. Declare that a requirement R_j is active at stage s iff R_j is active at stage $s-1$ and $D_s \cap res(R_j, s-1) = \emptyset$. For any R_j with $j \leq d_{s+1}$ which is currently passive and has the property that $P_j^{D_s}(z) = 0$ for all $z \in st(V_\infty)$ of length $s+1$, let $res(R_j, s)$ equal $res(R_j, s-1)$ union the set of all $y \notin D_s$ such that y is queried of the oracle D_s in one of the computations $P_j^{D_s}(z)$, where $z \in st(V_\infty)$ and $|z| = s+1$.

Case 2. $E_j = R_{j_s}$.
Let x_s denote the least x corresponding to j_s. We then say that R_{j_s} is active and receives attention at stage s. We let $D_s = D_{s-1}$ and $res(R_{j_s}, s)$ consist of all elements y of length $8|x_s|+2$ which are in some C_z such that $z \in st(V_\infty)$ and $|z| = |x_s|$, and all elements which are not in D_{s-1} and which are queried of the oracle D_{s-1} in the computation $P_{j_{s-1}}^{D_{s-1}}(x) = 1$. Note that if $res(R_{j_s}, s) \cap D = \emptyset$, then A will have no elements of height $|x_s|$ but $x_s \in P_{j_s}^D$. Also for all requirements E_j of the form S_j or Q_j, and for all requirements E_j of the form R_j where $j \neq j_s$ and where either R_j is satisfied at stage $s-1$ or $j > d_{s+1}$, let $res(E_j, s) = res(E_j, s-1)$. For $j \neq j_s$, declare that a requirement R_j is active at stage s iff R_j is active at stage $s-1$. For any R_j with $j \leq d_{s+1}$ which is

currently passive and has the property that $P_j^{D_s}(x) = 0$ for all $x \in st(V_\infty)$ of length $s+1$, let $res(R_j, s)$ equal $res(R_j, s-1)$ union the set of all $y \notin D_s$ such that y is queried of the oracle D_s in one of the computations $P_j^{D_s}(x)$, where $x \in st(V_\infty)$ and $|x| = s+1$.

Case 3. $E_j = Q_{j_s}$.
Let $j_s = [e_s, n_s]$ and x_s denote the least x corresponding to j_s. Then pick the least string $\alpha_{x_s} \in C_{x_s}$ such that α_{x_s} is not restrained from D by any requirement of higher priority than Q_{j_s} at stage $s-1$, nor is α_{x_s} queried of the oracle D_{s-1} in the computation of $P_{e_s}^{D_{s-1}}(x_s)$, and put α_{x_s} into D. Let $res(Q_{j_s}, s)$ consists of all strings which are not in D_{s-1} which are queried of the oracle D_{s-1} in the computation of $P_{e_s}^{D_{s-1}}(x_s)$, and say Q_{j_s} receives attention at stage s. Also for all requirements E_j of the form S_j or Q_j and for all requirements E_j of the form R_j, where either R_j is satisfied at stage $s-1$ or $j > d_{s+1}$, let $res(E_j, s) = res(E_j, s-1)$ if $D_s \cap res(E_j, s-1) = \emptyset$ and let $res(E_j, s) = \emptyset$ if $D_s \cap res(E_j, s-1) \neq \emptyset$. Declare that a requirement R_j is active at stage s iff R_j is active at stage $s-1$ and $D_s \cap res(R_j, s-1) = \emptyset$. For any R_j with $j \leq d_{s+1}$ which is currently passive and has the property that $P_j^{D_s}(x) = 0$ for all $x \in st(V_\infty)$ of length $s+1$, let $res(R_j, s)$ equal $res(R_j, s-1)$ union the set of all $y \notin D_s$ such that y is queried of the oracle D_s in one of the computations $P_j^{D_s}(x)$, where $x \in st(V_\infty)$ and $|x| = s+1$.

This completes the construction of D. We note that A is a height increasing independent set in NP^Y since our construction ensures that we can never put two elements of the same height in A. Thus by Theorem 8.28, $space(A) \in NP^Y$. We then have to prove the same sequence of lemmas as in Theorem 8.47 to complete the proof the theorem. The details may be found in [58]. □

Again the $space(A)$ constructed in Theorem 8.53 has a number of interesting properties besides being NP^D-supermaximal and P^D-simple. First of all, meeting all the requirements R_j ensures that $space(A)$ is P^D-immune. That is, if P_i^D is an infinite subset of $space(A)$, then certainly P_i^D generates an infinite dimensional subspace of $st(V_\infty)$ and $ht(P_i^D) \subseteq ht(space(A))$, which would violate requirement R_i. Also as in the construction of Theorem 8.47, it is easy to check that in meeting the requirements S_j we made no use of the fact that N_j^D was a subspace of $st(V_\infty)$, but only that N_j^D was a subset of $st(V_\infty)$. We claim that $st(V_\infty) \setminus space(A)$ does not have any infinite subsets in P^D. That is, suppose that $P_j^D \subseteq st(V_\infty) \setminus space(A)$. Now it cannot be that $ht(P_j^D) \setminus ht(space(A))$ is infinite since otherwise there is an i such that $P_j^D = N_i^D$ and the fact that we met requirement S_i would mean that $P_j^D \cap space(A) \neq \{\vec{0}\}$. Thus $ht(P_j^D) \subseteq^* ht(space(A))$. Let $Q = ht(space(A)) \setminus ht(P_j^D)$. Then clearly

$$S = \{x \in P_j^D : ht(x) \notin Q\}$$

is an infinite set in P^D which generates an infinite dimensional subspace of $st(V_\infty)$ and $ht(S) \subseteq ht(space(A))$. Since meeting all the requirements R_j rules out the existence of such an S, $st(V_\infty) \setminus space(A)$ does not contain an infinite set in P^D. Thus $space(A)$ and $st(V_\infty) \setminus space(A)$ are P^D-immune.

Note also that by Theorem 8.40, the fact that $space(A)$ is NP^D-maximal implies that $NP^D \neq$ co-NP^D and hence that $P^D \neq NP^D$. Thus we have proved the following.

Corollary 8.54 *There exists an r.e. oracle D and a subspace V of $st(V_\infty)$ such that*

(i) $P^D \neq NP^D$ *and* $NP^D \neq$ *co-*NP^D*,*

(ii) $V \in NP^D$*,*

(iii) V *is* P^D*-immune and hence has no basis in* P^D*,*

(iv) $st(V_\infty) - V$ *is* P^D*-immune, and*

(v) V *is both* P^D*-simple and* NP^D*-supermaximal.*

Finally we observe that results about NP and P subspaces of $tal(V_\infty)$ naturally extend to results about $NEXT$ and $DEXT$ subspaces of $st(V_\infty)$ by Lemma 4.3 For a typical example, say that a subspace M of $st(V_\infty)$ is $NEXT^A$-maximal if $M \in NEXT^A$, $dim(st(V_\infty)/M)$ is infinite, and for any subspace W of $st(V_\infty)$ in $NEXT^A$ containing M, either $dim(st(V_\infty)/W)$ is finite or $dim(W/M)$ is finite. Then Theorem 8.46 and Theorem 8.53 show that the question of the existence of $NEXT$-maximal subspaces is oracle dependent.

Theorem 8.55 *There is a recursive oracles A and an r.e. oracle B such that the following hold.*

(i) $NEXT^A \neq DEXT^A$ *and* $NEXT^B \neq DEXT^B$*.*

(ii) *There are no* $NEXT^A$*-maximal subspaces of* $st(V_\infty)$*.*

(iii) *There is an* $NEXT^B$*-maximal subspace* W *of* $st(V_\infty)$*.*

In the same way, all the results in this paper about P^X and NP^X subspaces of $tal(V_\infty)$ can be transfered to results $DEXT^X$ and $NEXT^X$ subspaces of $st(V_\infty)$.

Next we consider some results on splitting theorems for $tal(V_\infty)$ due to Bäuerle. We note a result of Ash and Downey [1] that every r.e. subspace of V_∞ is the direct sum of two decidable spaces. In $tal(V_\infty)$ the property of being a direct sum of two p-time subspaces is equivalent to having a p-time basis.

Theorem 8.56 *([5]) A subspace of V of $tal(V_\infty)$ can be split into two polynomial time subspaces if and only if V has a basis in P.*

Note that by the results on bases and subspaces of $tal(V_\infty)$, we immediately get the following corollaries.

Corollary 8.57 *There is an exponential time subspace W of $tal(V_\infty)$ that cannot be split into two polynomial time subspaces.*

Corollary 8.58 *For all r.e. degrees δ there is an r.e.subspace V of $tal(V_\infty)$ such that deg(V) $= \delta$ and V can be split into two polynomial time subspaces.*

Corollary 8.59 *There exists a recursive oracle A such that every $NP^A \setminus P^A$ subspace V of $tal(V_\infty)$ can be split into two P^A-vector spaces.*

Corollary 8.60 *Let $\mathcal{F}$ be finite. There exists a recursive oracle B such that there is a $NP^B \setminus P^B$ subspace V of $tal(V_\infty)$ that cannot be split into two P^B vector spaces.*

Corollary 8.61 *Arguments valid under relativization are not sufficient to show $NP \neq P \longrightarrow$ every NP-subspace of $tal(V_\infty)$ can be split into two P-time subspaces. $NP \neq P \longrightarrow$ there exists an $NP \setminus P$-subspace of $tal(V_\infty)$ which cannot be split into two P-time subspaces.*

In fact, Báuerle identifies three types of splittings by polynomial time subspaces of $tal(V_\infty)$.

Definition 8.62 *Let V be an r.e. vector space.*

1. *V allows a* **P-splitting** *if there exist P-time spaces W_0 and W_1 such that $W_0 \cap W_1 = \{\vec{0}\}$ and $W_0 + W_1 = V$. We say that W_0 and W_1 P-split V.*

2. *V allows an* **induced P-splitting** *if there exist P-time spaces W_0 and W_1 such that $W_0 \cap W_1 = \{\vec{0}\}$, $W_0 + W_1 = V_\infty$, $dim(V \cap W_1) = dim(V \cap W_0) = \infty$, and $(W_0 \cap V) + (W_1 \cap V) = V$. We say that W_0 and W_1 induce a P-splitting of V.*

3. *V allows an* **induced weak P-splitting** *if there exist r.e. vector spaces W_0 and W_1 such that W_0 and W_1 have bases in P, $W_0 \cap W_1 = \{\vec{0}\}$, $W_0 + W_1 = V_\infty$, $dim(V \cap W_1) = dim(V \cap W_0) = \infty$, and $(W_0 \cap V) + (W_1 \cap V) = V$. We say that W_0 and W_1 induce a weak P-splitting of V.*

Theorem 8.63 *([5]) Let V be a subspace of $tal(V_\infty)$.*

(i) *If V allows a P-splitting, then V allows an induced P-splitting.*

(ii) *If V allows an induced P-splitting, then V allows an induced weak P-splitting.*

Theorem 8.64 *([5])*

(i) *There exists an exponential time subspace V of $tal(V_\infty)$ that allows an induced P-splitting but no P-splitting.*

(ii) *There exists an exponential time subspace V of $tal(V_\infty)$ that does not allow an induced weak P-splitting*

This shows that three notions of Definition 8.62 are increasingly weaker.

We end this section, we some results on Bäuerle [5] on subspaces and superspaces of $NP \setminus P$-subspaces of $tal(V_\infty)$.

Theorem 8.65 *([5]) Every $NP \setminus P$-subspace V of $tal(V_\infty)$ has a non-trivial $NP \setminus P$-subspace W.*

Theorem 8.66 *([5]) Let V be a subspace of $tal(V_\infty)$ such that $V \in NP \setminus P$. If V has a non-trivial superspace in P, then V has a non-trivial superspace in $NP \setminus P$.*

Theorem 8.67 *([5]) There exists a recursive oracle C such that there exists a vector space $V \subset tal(V_\infty)$ that satisfies the following properties:*

1. *$V \in NP^C \setminus P^C$*
2. *V has a non-trivial superspace in $NP^C \setminus P^C$*
3. *V has no non-trivial superspaces in P^C*

Theorem 8.68 *([5]) There is a recursive oracle C such that all non-trivial $NP^C \setminus P^C$-subspaces of $tal(V_\infty)$ have non-trivial superspaces in P^C and in $NP^C \setminus P^C$.*

Finally under the assumption that $NP^A = \text{co-}NP^A$.

Theorem 8.69 *([5]) Let A be an oracle such that $NP^A \setminus P^A$-subspaces of $tal(V_\infty)$ exist and such that $NP^A = co\text{-}NP^A$. Then the following is true.*

1. *For all $NP^A \setminus P^A$-subspaces V of $tal(V_\infty)$ their standard height increasing basis B_V is in $NP^A \setminus P^A$.*
2. *For all $NP^A \setminus P^A$-subspaces V of $tal(V_\infty)$ their standard height increasing complementary basis $B_{\overline{V}}$ and their standard complement $(B_{\overline{V}})^*$ are in $NP^A \setminus P^A$.*
3. *Every $NP^A \setminus P^A$-subspace V of $tal(V_\infty)$ can be split into two disjoint $\leq_T^P$ incomparable $NP^A \setminus P^A$-subspaces.*
4. *The set of $\leq_T^P$ degrees with NP^A-subspaces of $tal(V_\infty)$ is dense.*
5. *There exists a pair V, W of $NP^{\setminus}P^A$-subspaces of $tal(V_\infty)$ such that if $U \leq_T^P V$ and $U \leq_T^P W$, then $U \in P^A$.*
6. *The set of rationals Q can be embedded in the structures of $\leq_m^P$ and $\leq_T^P$-degrees of NP^A-subspaces of $tal(V_\infty)$.*

9 Polynomial Time Boolean Algebras

In this section, we shall survey the results of Nerode and Remmel on the lower semilattice of NP-ideals of a polynomial time presentation of the free Boolean algebra. Again we consider two natural representations of the free Boolean algebra called the tally and standard representation. We start by describing these two representations.

Let $\mathcal{P}([0,1))$ denote the Boolean algebra of all subsets of the rational left-closed right-open interval $[0,1)$ in the rational number $\mathcal{Q}$. The Boolean operations of meet, join, and complementation on $\mathcal{P}([0,1))$ are respectively intersection, union, and relative complement in $[0,1)$. Let $\mathcal{B}([0,1))$ be the subalgebra of $\mathcal{P}([0,1))$ generated by the left-closed right-open intervals of the form

$$[\frac{i}{2^n}, \frac{j}{2^n})$$

with $n \geq 0$ and $0 \leq i < j \leq 2^n$. For any subset $S \subseteq \mathcal{B}([0,1))$, $(S)^*$ denotes the subalgebra of $\mathcal{B}([0,1))$ generated by S and $I(S)$ denotes the ideal generated by S. Given a subalgebra $D \subseteq \mathcal{B}([0,1))$, we let $At(D)$ denote the set of atoms of D.

Next we define a natural generating sequence $a_1, a_2, \ldots$ for $\mathcal{B}([0,1))$ by induction:

$a_1 = [0,1)$

$a_{2^{n-1}+m+1} = [\frac{2m}{2^n}, \frac{2m+1}{2^n})$ if $n \geq 1$ and $0 \leq m < 2^{n-1}$.

Thus

$$a_2 = [0, \frac{1}{2}), \quad a_3 = [0, \frac{1}{4}), \quad a_4 = [\frac{1}{2}, \frac{3}{4}), \quad a_5 = [0, \frac{1}{8}),$$
$$a_6 = [\frac{1}{4}, \frac{3}{8}), \quad a_7 = [\frac{1}{2}, \frac{5}{8}), \quad a_8 = [\frac{3}{4}, \frac{7}{8})$$

Let $A_n = \{a_1, \ldots, a_n\}^*$. Then it is not difficult to see that $A_1, A_2, A_3, \ldots$ is a strictly increasing sequence of subalgebras such that for each $n \geq 1$, there is a unique atom $x_n \in At(A_n)$ such that a_{n+1} splits x_n, i.e. $\emptyset \subset a_{n+1} \subset x_n$. In fact, one can easily show by induction that if k is of the form $2^{n-1} + m$ with $0 \leq m < 2^{n-1}$, then

$$At(A_k) = \{[\frac{i}{2^n}, \frac{i+1}{2^n}) : 0 \leq i < 2m\} \cup \{[\frac{j}{2^{n-1}}, \frac{j+1}{2^{n-1}}) : m \leq j < 2^{n-1}\}.$$

Hence $a_{k+1} = [\frac{2m}{2^n}, \frac{2m+1}{2^n})$ splits the atom $x_k = [\frac{m}{2^{n-1}}, \frac{m+1}{2^{n-1}})$ of A_k. It follows that A_n has exactly n atoms for each $n \geq 1$, so that A_n has exactly 2^n elements.

We use this generating sequence and its corresponding sequence of subalgebras

$$A_1 \subset A_2 \subset A_3 \subset \ldots$$

to define the standard and tally representations of $\mathcal{B}([0,1))$.

The Standard Representation of $\mathcal{B}([0,1))$

First we describe a coding of the elements of $\mathcal{B}([0,1))$ which we call that *standard representation* of $\mathcal{B}([0,1))$. Our idea is to use binary numbers of length n to code the elements of $A_n \setminus A_{n-1}$ for $n > 1$. Formally, we define a 1:1 correspondence $\delta \to s_\delta$ between $Bin(\omega)$ and $\mathcal{B}([0,1))$ by induction.

For the base step, set

$$s_0 = \emptyset, s_1 = [0,1).$$

For the induction step, assume that the correspondence $bin(k) \to s_k$ has been defined between $\{bin(k) : 0 < k < 2^n\}$ and A_n. We then extend our correspondence to A_{n+1} as follows. Given a binary number m of length $n+1$, let $m = k \cdot 2^i$ where k is odd, so that $bin(m) = 0^i \frown bin(k)$, and let

$$s_m = \begin{cases} s_k \cup a_{n+1} & \text{if } s_k \cap a_{n+1} = \emptyset \\ s_k \setminus a_{n+1} & \text{if } s_k \supseteq a_{n+1}. \end{cases} \tag{1}$$

Now let $At(A_n) = \{x_1, \ldots, x_{n-1}, x_n\}$ where x_n is the atom of A_n which is split by a_{n+1}. Then it is easy to see that every element of $A_{n+1} \setminus A_n$ is either of the form

$$a_{n+1} \cup \bigcup_{i \in S} x_i$$

or

$$(x_n \setminus a_{n+1}) \cup \bigcup_{i \in S} x_i$$

for some set $S \subseteq \{1, \ldots, n-1\}$. Thus (1) defines a 1:1 correspondence between $\{k : 2^n \leq k < 2^{n+1}\}$ and $A_{n+1} \setminus A_n$.

Indeed it is quite easy to use (1) to recursively construct s_n. We write $s_{bin(n)}$ for s_n in the following.

Example 9.1 *Suppose $bin(n) = 0101101$. Then s_n can be constructed as follows.*

$s_0 = \emptyset$.

$s_{01} = a_2 = [0, \frac{1}{2})$ *since* $a_2 \cap s_0 = \emptyset$.

$s_{0101} = s_{01} \cup a_4 = [0, \frac{1}{2}) \cup [\frac{1}{2}, \frac{3}{4} = [0, \frac{3}{4})$ *since* $a_4 \cap s_{01} = \emptyset$.

$s_{01011} = s_{0101} \setminus a_5 = [0, \frac{3}{4}) \setminus [0, \frac{1}{8} = [\frac{1}{8}, \frac{3}{4})$ *since* $a_4 \subseteq s_{0101}$.

$s_{0101101} = s_{01011} \setminus a_7 = [0, \frac{3}{4}) \setminus [\frac{1}{2}, \frac{5}{8} = [\frac{1}{8}, \frac{1}{2}) \cup [\frac{5}{8}, \frac{3}{4})$ *since* $a_7 \subseteq s_{01011}$.

It is not difficult to show that given two σ and τ in $Bin(\omega)$ with $|\sigma| \leq |\tau|$, we can find α, β and γ in $Bin(\omega)$ such that

$$s_\alpha = s_\sigma \cup s_\tau, \;\; s_\beta = s_\sigma \cap s_\tau, \;\; s_\gamma = [0,1) \setminus s_\tau$$

in polynomial time in $|\tau|$. Furthermore, note that each of α, β and γ has length $\leq 2|\tau|$, since each of s_α, s_β and s_γ belongs to A_n if $s_\tau \in A_n$. See [54] for details. It follows that if we then define

$$\sigma \wedge_s \tau = \alpha \quad (2)$$

$$\sigma \vee_s \tau = \beta \quad (3)$$

$$\neg_s \tau = \gamma \quad (4)$$

Then $st(\mathcal{B}) = (Bin(\omega), \wedge_s, \vee_s, \neg_s)$ is a polynomial time representation of the countable atomless Boolean algebra $\mathcal{B}([0,1))$ which we call the standard representation of $\mathcal{B}([0,1))$.

The Tally Representation of $\mathcal{B}([0,1))$

The tally representation $tal(\mathcal{B})$ of $\mathcal{B}([0,1))$ can easily be defined from the binary representation $st(\mathcal{B}([0,1))$ to be the isomorphic image under the map taking $bin(n)$ to $tal(n)$ and is therefore a p-time structure by Lemma 4.4 in light of the note above.

Nerode and Remmel [54] studied three basic properties of ideals in a recursive Boolean algebras. Here given a Boolean algebra $\mathcal{B} = (B, \wedge_B, \vee_B, \neg_B)$, we say $I \subseteq B$ is an *ideal* if the zero of B, 0_B is in I, for all $x, y \in I$ imply $x \vee_B y \in B$, and for $x \in I$ and $z \in B$, $x \wedge_B z \in I$. I is a maximal ideal if for all $z \in B$, either $z \in I$ or $\neg_B z \in I$.

Nerode and Remmel studied polynomial time analogues of the following well known results on r.e. ideals in a recursive presentation of $\mathcal{B}([0,1))$.

A. In a recursive Boolean algebra, every r.e. maximal ideal is recursive.

B. Every proper recursive ideal is contained in a recursive maximal ideal.

C. There exists an r.e. ideal of $\mathcal{B}([0,1))$ which is not extendible to a recursive ideal.

We note that **C** is equivalent to the proposition that there is an r.e. axiomatizable theory which is not contained in any decidable theory.

First consider **A**. The fact that every r.e. maximal ideal of a recursive Boolean algebra is based on Kleene's lemma that a set which is r.e. and co-r.e. is automatically recursive. The obvious p-time analogue of **A** is that every NP maximal ideal of p-time Boolean algebra is polynomial time. However in this case, it is a long standing open problem whether $NP \cap \text{co-}NP = P$. Moreover is well known results of Baker-Gill–Solovay [4], there exists recursive oracle X and Y such that $NP^X \cap \text{co-}NP^X \neq P^X$ and $NP^Y \cap \text{co-}NP^Y = P^Y$ but $P^Y \neq NP^Y$. Thus it should come as no surprise that the analogue of **A** is oracle dependent. That is, Nerode and Remmel were able of modify the Baker-Gill-Solovay constructions to prove the following.

Theorem 9.1 *There exists a recursive oracle X such that there exist a maximal ideal I of $tal(\mathcal{B})$ such that $I \in NP^X \setminus P^X$.*

Our next corollary immediately follows from Theorem 9.1 and Lemma 4.3.

Corollary 9.2 *There exists a recursive oracle X such that there exists a recursive oracle X such that there exist a maximal ideal I of $st(\mathcal{B})$ such that $I \in NEXT^X \setminus DEXT^X$.*

Theorem 9.3 *There exists a recursive oracle Y such that there exist a maximal ideal J of $st(\mathcal{B})$ such that $J \in NP^X \setminus P^X$.*

Theorem 9.4 *There exists a recursive oracle E such that $P^E \neq NP^E$ and the following hold.*

(i) *Every maximal ideal I of $tal(\mathcal{B})$ which is in NP^E is in P^E.*

(ii) *Every maximal ideal J of $st(\mathcal{B})$ which is in $NEXT^E$ is in $DEXT^E$.*

(iii) *Every maximal ideal K of $st(\mathcal{B})$ which is in NP^E is in P^E.*

Theorems (9.1-9.4) then yield the following results.

Theorem 9.5 *Arguments which remain valid under relativization to oracles do not suffice to prove any of the following.*

(1) *$P \neq NP$ implies that every NP maximal ideal of $tal(\mathcal{B})$ is in P.*

(2) *$P \neq NP$ implies that there is an NP maximal ideal of $tal(\mathcal{B})$ which is not in P.*

(3) *$P \neq NP$ implies that every NP maximal ideal of $st(\mathcal{B})$ is in P.*

(4) *$P \neq NP$ implies that there is a NP maximal ideal of $st(\mathcal{B})$ which is not in P.*

Next we turn to the analogues of **B**. In this case, Nerode and Remmel proved that the obvious analogues of **B** are true for both $tal(\mathcal{B})$ and $st(\mathcal{B})$ although the argument requires a great deal more care. That is, Nerode and Remmel [54] proved the following.

Theorem 9.6 *Every proper ideal I of $st(\mathcal{B})$ which is in P can be extended to maximal ideal J of $st(\mathcal{B})$ which is in P.*

Theorem 9.7 *Every proper ideal I of $st(\mathcal{B})$ which is in $NP \cap$ co-NP can be extended to maximal ideal J of $st(\mathcal{B})$ which is in $NP \cap$ co-NP.*

Theorem 9.8 *Every proper ideal I of $st(\mathcal{B})$ which is in $DEXT$ can be extended to maximal ideal J of $st(\mathcal{B})$ which is in $DEXT$.*

Theorem 9.9 *Every proper ideal I of $st(\mathcal{B})$ which is in $NEXT \cap$ co-$NEXT$ can be extended to maximal ideal J of $st(\mathcal{B})$ which is in $NEXT \cap$ co-$NEXT$.*

Theorem 9.10 *Every proper ideal I of $tal(\mathcal{B})$ which is in P can be extended to maximal ideal J of $tal(\mathcal{B})$ which is in P.*

Theorem 9.11 *Every proper I of $tal(\mathcal{B})$ which is in $NP \cap$ co-NP can be extended to maximal ideal J of $tal(\mathcal{B})$ which is in $NP \cap$ co-NP.*

Finally we turn to the analogues of **C**. In this case, the analogues are oracle dependent despite Theorems 9.6 - 9.11.

Theorem 9.12 *There exists a recursive oracle A such that $P^A \neq NP^A$ and*

(1) *every proper ideal I_1 of $st(\mathcal{B})$ which is in NP^A is extendible to maximal ideal J_1 of $st(\mathcal{B})$ which is in NP^A,*

(2) *every proper ideal I_2 of $tal(\mathcal{B})$ which is in NP^A is extendible to maximal ideal J_2 of $tal(\mathcal{B})$ which is in NP^A.*

(3) *every proper ideal I_3 of $st(\mathcal{B})$ which is in $NEXT^A$ is extendible to maximal ideal J_3 of $st(\mathcal{B})$ which is in $NEXT^A$.*

Proof: Homer and Maass [36] constructed a recursive oracle A such that $P^A \neq NP^A$ but $NP^A =$ co-NP^A. Thus we can use the relativized version of Theorem 9.7 to prove part (1) and we can use the relativized version Theorem 9.11 to prove part (2). Finally part (3) follows from part (2) and Lemma 4.3. □

The proof of the other direction of the oracle dependence requires a new construction is much more subtle than any of the previous theorems on ideals in our p-time representation of $\mathcal{B}([0, 1))$. The actual proofs can be found in [54].

Theorem 9.13 **(1)** *There exists a recursive oracle C and an ideal J_1 of $tal(\mathcal{B})$ which is in NP^C which is not contained in any maximal ideal of $tal(\mathcal{B})$ which is in NP^C.*

(2) *There exists a recursive oracle B and an ideal J_2 of $st(\mathcal{B})$ which is in NP^B which is not contained in any maximal ideal of $st(\mathcal{B})$ which is in NP^B.*

Of course, we can combine Lemma 4.3 and Theorem 9.13 (1) to prove the following.

Theorem 9.14 *There exists a recursive oracle C and an ideal J_3 of $st(\mathcal{B})$ which is in $NEXT^C$ which is not contained in any maximal ideal of $tal(\mathcal{B})$ which is in $NEXT^C$.*

10 Conclusions and Future Directions

In this survey, we have presented the basic definitions of complexity theoretic algebra and model theory and have attempted to outline the current state of knowledge in the field. There is a great deal more which remains to be done. We will just mention four possible themes for future research.

First, we observe that the results on complexity theoretic algebra were limited to the study of ideals in the free Boolean algebras and and subspaces of an infinite dimensional vector spaces. There are many other algebraic structures that have been studied in recursive algebra including fields, modules, subalgebras of Boolean algebras, subgroups of groups, etc.. Cenzer, Downey and Remmel [9] have recently investigated torsion-free Abelian groups. We have also given complexity theoretic results in combinatorics. Other related areas of mathematics such as geometry and number theory should also provide fruitful basis for investigation.

Second, most of our results concerned the notions of polynomial time complexity, with some results given on linear time and on exponential time complexity. There are many other interesting notions of complexity, including for example, PSPACE and LOGSPACE, which should provide both comparable and contrasting results.

Third, we gave only a few results involving the important complexity hypotheses of theoretical computer science, such as whether $P = NP$ or $NP = PSPACE$. In the complexity theory of real functions, Ko [42] has provided many such results. For example, he gives a condition (not involving complexity) on a real function f and shows that if $P = NP$ and if f is a p-time computable function on the unit interval which satisfies this condition, then all roots of f are p-time computable. There should be similar results in complexity theoretic algebra.

Fourth, we have only begun the study of complexity theoretic model theory with a few results on relational structures and with the general notion of Scott sentences and categoricity. If one studies the recursive model theory survey by Harizanov [33], many problems suggest themselves. For example, the authors have recently investigated complexity theoretic versions of the effective completeness theorem in [18]. Decidability is also of interest in the study of prime and saturated models and in stability theory.

References

[1] Ash, C.J. and Downey, R.G.: Decidable subspaces and recursively enumerable subspaces, J. Symbolic Logic 49, pp 1137-1145, (1984).

[2] Ash, C. and Nerode, A.: Intrinsically recursive relations. *Effective Aspects of Algebra*, editor J.N. Crossley, Upside Down A Book Co., Clayton, Victoria, Australia (1980) 26–41.

[3] Balcázar, J.L, Díaz, J., and Gabarró, J.: *Structural Complexity I.* Springer-Verlag, 1988.

[4] T. Baker, J. Gill, and R. Solovay: Relativisations of the $P = NP$ question. *SIAM J. Comp.* 4, pp. 431-442, 1981.

[5] Bäuerle, F.A.: Ph.D. Thesis, University of California at San Diego, La Jolla, CA 92093 (1995).

[6] Bennett, C.H. and Gill, J.: Relative to a Random Oracle $A, PA \neq NPA \neq$ co-NPA with Probability 1. *SIAM J. Comp.* 10, pp. 96-113, 1981.

[7] Breidbart, S.: On splitting recursive sets. *J. Comput. System Sci.* 17, pp. 56-64, 1978.

[8] Cai, J.: On Some Most Probable Separations of Complexity Classes. *Computer Science Ph.D,. Diss.,* Cornell University , Ithaca, New York, 1986.

[9] Cenzer, D., Downey, R. and Remmel, J.B.: Feasible torsion-free Abelian groups. To appear.

[10] Cenzer, D. and Remmel, J.B.: Polynomial-time versus recursive models. Ann. Pure and Appl. Logic **54** (1991) 17–58.

[11] Cenzer, D. and Remmel, J.B.: Polynomial-time Abelian groups. Ann. Pure and Appl. Logic **56** (1992) 313–363.

[12] Cenzer, D. and Remmel, J.B.: Recursively presented games and strategies. Math. Social Sciences **24** (1992) 117–139.

[13] Cenzer, D. and Remmel, J.B.: Feasible graphs and colorings. Math. Logic Quarterly **41** (1995) 327–352.

[14] Cenzer, D. and Remmel, J.B.: Feasibly categorical abelian groups, in *Feasible Mathematics II*, editors P. Clote and J. Remmel, Prog. in Comp. Science and Appl. Logic **13**, Birkhäuser (1995) 91–154.

[15] Cenzer, D. and Remmel, J.B.: Feasibly categorical models, in *Logic and Computational Complexity*, editor D. Leivant, Lecture Notes in Comp. Science **960**, Springer (1995) 300–312.

[16] Cenzer, D. and Remmel, J.B.: Feasible graphs with standard universe, in *Computability Theory, Oberwolfach 1996*, Special issue of Ann. Pure and Appl. Logic, to appear.

[17] Cenzer, D. and Remmel, J.B.: Π_1^0 classes. This volume.

[18] Cenzer, D. and Remmel, J.B.: Decidability and completeness and complexity. To appear.

[19] Cobham, A.: The Intrinsic Computational Difficulty of Functions. *Proc. Inter. Cong, for Logic Methodology, and Philosophy of Science, North Holland*, pp. 24-30, 1964.

[20] Cook, S.A.: The Complexity of Theorem-Proving Procedures. *ACM Symposium on Theory of Computation,* pp. 151-158, 1971.

[21] Crossley, J.N.: *Aspects of Effective Algebra*, (J.N. Crossley, ed.) Upside Down A Book Company, Yarra Glen, Victoria , Australia, 1981.

[22] Crossley, J.N. and Remmel, J.B.: Polynomial-time combinatorial operators are polynomials. *Feasible Mathematics*, (S. Buss and P. Scott, eds), Birkhauser, pp. 99–130, 1990.

[23] Crossley, J.N. and Remmel, J.B.: Cancellation laws for polynomial-time p-isolated sets. *Annals of Pure and Appl. Logic* **56**, pp. 147-172, 1992.

[24] Dekker, J.C.E.: Two notes on vector spaces with recursive operations. *Notre Dame J. Formal Logic* 12, pp. 329-334, 1971.

[25] Downey, R.G. and Fellows, M.R.: Fixed Parameter Intractability and Completeness III: Some Structural Aspects of the W-Heirarchy. *Complexity Theory* (Ed. K. Ambos-Spies, S. Homer, and U. Schöning), Cambridge Uiversity Press, pp. 166-191, 1993.

[26] Edmonds, J.: Paths, Trees, and Flowers. *Canad. J. Math 17,* pp. 449-467, 1965.

[27] Friedman, H. and Ko, K.: Computational complexity of real functions. *Theoret. Comput. Sci.* 20 (1982), no. 3, 323–352.

[28] Garey, M.R. and Johnson, D.S.: *Computers and Intractability. W.H. Freeman, San Francisco.* 1979.

[29] Goncharov, S.S.: Autostability and computable families of constructivization. Algebra and Logic **14** (1975) 392–409.

[30] Goncharov, S.S.: Some properties of the constructivization of Boolean algebras. Sibirsk Mat. Z. **16** (1975) 264–278.

[31] Goncharov, S.S., Dzgoev, V.D.: Autostability of models. Algebra and Logic **19** (1980) 28–37.

[32] Grigorieff, S.: Every Recursive linear ordering has a copy in DTIME(n). J. Symbolic Logic **55** (1990) 260–276.

[33] Harizanov, V.: Pure recursive model theory. This volume.

[34] Hartmanis, J.: On sparse sets in $NP - P$, *Information Processing Letters* 16, pp. 55-60, (1983).

[35] Hartmanis, J. and Stearns, R.: On the Computational Complexity of Algorithms. *Trans. Amer. Math. Soc. 117,* pp. 285-306, 1965.

[36] Homer, S. and Maass, W.: Oracel Dependent Properties of the Lattice NP-sets. *Theor. Comp. Science, 24,* pp. 279-289, 1983.

[37] Hopcroft, J. and Ullman, J.: "Formal Languages and Their Relations to Automata". Addison Wesley (1969).

[38] Kalantari, I. and Retzlaff, A.: Maximal Vector Spaces under Automorphisms of the Lattice of Recursively Enumerable Vector Spaces. *J. Symb. Logic,* pp. 481-491, 1977.

[39] Karp, R.: Reducibilities Among Combinatorial Problems. *Complexity Computer Computations,* pp. 85-103, 1972.

[40] Khisamiev, N. G.: Constructive Abelian Groups. This volume.

[41] Khoussainov, B. and Nerode, A.: Automatic Presentations of Structures. *Logic and Computational Complexity* editor D. Leivant, Lecture Notes in Comp. Science **960**, Springer (1995), pp. 367–392.

[42] Ko, Ker-I: *Complexity theory of real functions,* Progress in Theoretical Computer Science. Birkhauser (1991).

[43] Ko, Ker-I: Computational complexity of fixed points and intersection points. *J. Complexity* **11.2**, (1995), pp. 265–292.

[44] LaRoche, P.E.: Recursively presented Boolean algebras. Notices Amer. Math. Soc. **24** A-552.

[45] Mahaney, S.R.: Sparse complete sets for NP: solution of a conjecture of Berman and Hartmanis, it Journal of Computer and System Sciences 25, pp. 130-143, (1982).

[46] Martin, D.A.: Classes of Recursively Enumerable Sete and Degrees of Unsolvability. *Zeit. Math. Logik 12*, pp. 295-310, 1966.

[47] Metakides, G. and Nerode, A.: Recursion Theory and Algebra. *Lecture Notes in Mathematics 450, Springer-Verlag, Berlin,* pp. 209-219, 1975.

[48] Metakides, G. and Nerode, A.: Recursively Enumerable Vector Spaces. *Ann. of Math. Logic 11,* pp 141-171, 1979.

[49] Mal'cev, A.I.: On recursive Abelian groups. *Soviet Math.* **3** (1962) 1431–1432.

[50] Nerode, A. and Remmel, J.B.: Recursion Theory on Matroids, II. *Southeast Asian Conference on Logic* (eds. C.T. Chong and M.J. Wicks), North Holland Pub. Co., New York, (1983), 133-184.

[51] Nerode, A. and Remmel, J.B.: A Survey of R.E. Substructures. *Proc. Symposia in Pure Math 42, Amer. Math. Soc.,* pp. 323-373, 1985.

[52] Nerode, A. and Remmel, J.B.: Complexity theoretic algebra I: vector spaces over finite fields. *Logic and Computer Science*, Rendconti del Seminario Matematico: Covegno su Logic and Computer Science, New Trends and Applications (1986), Fascicolo Speciale, Torino Universita e Politecnica, La Precisa, Torino, 1987.

[53] Nerode, A., Remmel, J.B.: Complexity theoretic algebra I, vector spaces over finite fields, *Proceedings of Structure in Complexity, 2d Annual Conference,* Computer Science Press (1987) 218–239.

[54] Nerode, A., Remmel, J.B.: Complexity theoretic algebra II, the free Boolean algebra. Ann. Pure and Applied Logic **44** (1989) 71–99.

[55] Nerode, A., Remmel, J.B.: Complexity theoretic algebra: vector space bases, *Feasible Mathematics*, editors S. Buss and P. Scott, Prog. in Comp. Science and Appl. Logic **9** Birkhäuser (1990) 293–319.

[56] Nerode, A. and Remmel, J.B.: Polynomial time equivalence types. *Logic and Computation: Proc. of a Workshop held at CMU, June 3 - July 2, 1987*, (W. Sieg, ed.), Contemporary Mathematics Series vol. 106, American Mathematical Society, Providence, R. I., 1990.

[57] Nerode, A. and Remmel, J.B.: Polynomially isolated sets. *Recursion Theory Week: Proc. of a Conference at Oberwolfach, FRG, March 19-25, 1989,* (K. Ambos-Spies, G. Müller, G. E. Sacks, eds.), Springer, 1990.

[58] Nerode, A. and Remmel, J.B.: On the lattice of NP-subspaces of a Polynomial Time Vector Space over a Finite Field. *Annals of Pure and Applied Logic*, to appear.

[59] Nerode, A., Remmel, J.B., and Scedrov, A.: Polynomial graded logic: A graded version of Godel's system T, extended abstract with proofs. *Proc. Logic in Computer Science* (LICS), Computer Society of IEEE, pp. 375-384, 1989.

[60] Nurtazin, A.: "Completable classes and algebraic conditions for autostability". Ph. D. thesis, Novosibirsk (1974)

[61] P. Odifreddi: "Classical Recursion Theory". North–Holland, Amsterdam (1989).

[62] J.B. Remmel "Maximal and Cohesive Vector Spaces", *J. Sym. Logic* 42, pp. 400-418, 1977.

[63] Remmel, J.B.: Recursively Enumerable Boolean Algebras. *Ann. Math. Logic* 14, pp. 75-107, 1978,

[64] Remmel, J.B.: Recursion Theory on Algebraic Structures with an Independent Set. *Ann. Math. Logic 18,* pp. 153-191, 1980.

[65] Remmel, J.B.: On Recursively Enumerable and Co-Recursively Enumerable Vector Spaces with Non-Extendible Bases. *J. Symb. Logic 45,* pp. 20-24, 1980.

[66] Remmel, J.B.: Recursively categorical linear orderings. Proc. Amer. Math. Soc. **83** (1981) 387–391.

[67] Remmel, J.B.: Recursive Boolean Algebras, in *Handbook of Boolean Algebras*, ed. J. Monk. North-Holland (1989) 1099–1165.

[68] Remmel, J.B.: When is every recursive linear ordering of type μ recursively isomorphic to a p-time linear order over the binary representation of the natural numbers?, *Feasible Mathematics*, editors S. Buss and P. Scott. Prog. in Comp. Science and Appl. Logic **9** Birkhäuser (1990) 321–341.

[69] Remmel, J.B.: Polynomial-time categoricity and linear orderings, in *Logical Methods*, editors J. Crossley, J. Remmel, R. Shore, and M. Sweedler, Prog. in Comp. Science and Appl. Logic **12** Birkhäuser (1993) 321–341.

[70] Schmerl, J.: Recursive colorings of graphs. *Can. J. Math.* **32** (1980) 821–830.

[71] Schoening, U.: Complexity and Structure. *Lecture Notes in Computer Science* 211, Springer-Verlag, Berlin, 1986.

[72] Shinoda, J. and Slaman, T.: On the theory of PTIME degrees of recursive sets. *J. Comput. System Sci.*, 40 pp. 417-443, 1990.

[73] Shore, R.A. and Slaman, T.A.: The p-T-degrees of the recursive sets: lattice embeddings, extensions of embeddings and the two quantifier theory. *Th. Comp. Sci.*, 97 pp. 263-284, 1992

[74] Smith, R.: Two theorems on autostability in p-groups, it Logic Year 1979-80 (Storrs, Conn.), Lecture Notes in Math. **859** Springer-Verlag (1981) 302–311.

[75] Wilson, C.: *Relativization, Reducibilities, and the Exponential Time Hierarchy*, Ph.D. Thesis, U. Toronto, (1980).

[76] Yao, A.: Separating the Polynomial-Time Hierarchy by Oracles. *IEEE Symp. on Found. of Computer Science [FOCS],* pp. 1-10, 1985.