Π^0_1-Classes

Richard A. Shore ©

January 25, 2012

1 Binary trees

We now return to the basic our basic notion of a tree as a downward closed subset of $\omega^{<\omega}$. In this context we use T_σ to denote the subtree of T consisting of all strings ρ compatible with σ: $T_\sigma = \{\rho | \rho \subseteq \sigma \text{ or } \sigma \subseteq \rho\}$. Recall that the sets of paths in such trees are the closed sets in Baire space ω^ω. In this chapter we are primarily concerned with infinite binary trees, i.e. the infinite downward closed subsets T of $2^{<\omega}$. We endow each binary tree with a left to right partial order as well as the order of extension. It is specified by the lexicographic order on strings so σ is to the left of τ, $\sigma <_L \tau$ if $\sigma(n) < \tau(n)$ for the least n such that $\sigma(n) \neq \tau(n)$ if there is one. (This order extends in the obvious way to one of 2^ω which we also call the left to right or lexicographic order.) The sets of paths $[T] = \{A \in 2^\omega : \forall n (A \restriction n \in T)\}$ through these trees are precisely the nonempty closed subsets of Cantor space, 2^ω.

Exercise 1 *For any binary tree T, $[T]$ is a closed set in Cantor space.*

??Prove??

To see that every closed subset of 2^ω is of the form $[T]$ for some tree T, consider the open sets in 2^ω. They are all unions of basic (cl)open sets of the form $[\sigma] = \{f \in 2^\omega | \sigma \subseteq f\}$ for $\sigma \in 2^{<\omega}$. So given any closed set $\mathcal{C}$ its complement $\bar{\mathcal{C}}$ is a union of such neighborhoods. Let $T = \{\sigma | [\sigma] \not\subseteq \bar{\mathcal{C}}\} = \{\sigma | [\sigma] \cap \mathcal{C} \neq \emptyset\}$. It is clear that T is downward closed. If $f \in \mathcal{C}$ and $\sigma \subseteq f$ then clearly $\sigma \in T$ and so $f \in [T]$. On the other hand if $f \in [T]$ and $\sigma \subseteq f$ then $\sigma \in T$ and so the closed set $[\sigma] \cap \mathcal{C} \neq \emptyset$. As Cantor space is compact $\cap\{[\sigma] \cap \mathcal{C} | \sigma \subseteq f\}$ is nonempty and only f can be in it so $f \in \mathcal{C}$ as required. Note that, by König's lemma (Lemma **??**), $\mathcal{C}$ is nonempty if and only if T is infinite.

??Move this material to Trees section and recall here??

In this chapter we want to investigate the recursive versions of these two notions.

Definition 2 *A class $\mathcal{C} \subseteq 2^\omega$ is* effectively closed *if it is of the form $[T]$ for a recursive binary tree T.*

We can also characterize the effectively closed sets in terms of the complexity of their definition. We use the same notation based on the arithmetic hierarchy for classes of sets or functions as we did for individual sets and functions.??say more now or before Go back and check definitions for Σ_n^A especially Σ_0 and how interpret for σ in place of A ...bounded quantifiers??

Definition 3 *A class $\mathcal{C} \subseteq 2^\omega$ of sets is Σ_n (Π_n) if there is a Σ_n (Π_n) formula $\varphi(X)$ with one free set variable X such that $\mathcal{C} = \{A|\mathbb{N} \vDash \varphi(A)\}$. Similarly for classes $\mathcal{F} \subseteq \omega^\omega$ of functions and formulas with one free function variable.*

The primary connection with trees is the following Proposition.

Proposition 4 *The Π_1^0 classes of sets are precisely the sets of paths through recursive binary trees. Again, the nonempty classes correspond to the infinite recursive binary trees. Moreover, there is a recursive procedure that takes an index for a Π_1^0 formula to one for a recursive tree T such that $[T]$ is the corresponding Π_1^0 class.*

Proof. If T is a recursive binary tree then $[T] = \{A \in 2^\omega : \forall n(A \upharpoonright n \in T)\}$ is clearly a Π_1^0 class. If T is infinite, $[T]$ is nonempty by König's lemma while if T is finite $[T]$ is clearly empty. For the other direction consider any Π_1^0 class $\mathcal{P} = \{A : \forall x R(A, x)\}$ for a Σ_0^A relation R. Let $T = \{\sigma \in 2^{<\omega} | \neg(\exists x < |\sigma|) \neg R(\sigma, x)\}$ where we understand that we are thinking of σ as representing an initial segment of A. Formally we replace $t \in A$ by $\sigma(t) = 1$ and declare the formula $R(\sigma, x)$ false if some term $t > |\sigma|$ occurs in it as described in ??. It is then immediate that $P = [T]$ and that an index for T as recursive function is given uniformly in the index for R as a Σ_0^A formula. If P is nonempty, T has an infinite path and so is itself infinite. Otherwise, T is finite. ■

Exercise 5 *The Π_1^0 classes of functions are precisely the sets of paths through recursive trees (on $\omega^{<\omega}$).*

We can now index the Π_1^0 classes (of sets) by either the indices of the Π_1^0 formulas or of the trees derived from them as in the proof of Proposition 4 as partial recursive functions which are actually total. A natural question then is how hard is to tell if a recursive tree is infinite or a Π_1^0 class is nonempty. It might seem at first that these properties are Π_2^0 and so only recursive in $0''$. If we know that the tree is recursive as we do for the trees derived uniformly from Π_1^0 classes, however, then the question is actually uniformly (on indices) recursive in $0'$. This observation depends on the compactness of Cantor space and plays a crucial role in almost every argument in the rest of this chapter.

Lemma 6 *If T is a recursive binary tree (say with index i so $T = \Phi_i$) then T being finite is a Σ_1 property (of i). Thus we can decide if T is finite or infinite recursively in $0'$. Indeed, T is finite if and only if there is an n such that $\sigma \notin T$ for every σ of length n.*

Proof. Clearly, T is finite if and only if there is an n such that $\sigma \notin T$ for every σ of length n. Clearly this is a Σ_1 property for any recursive binary tree and the associated Σ_1 formula is given uniformly in a recursive index for T. ■

While deciding if a given recursive binary tree is infinite or a Π^0_1 class nonempty requires $0'$, we can actually make a recursive list of the nonempty Π^0_1 classes and so one of corresponding infinite recursive binary trees (up to the set of paths on T).

Exercise 7 *There is a uniformly recursive list of the nonempty Π^0_1 classes in the sense that there is a recursive set Q such that, for each $e \in Q$, Φ_e is (the characteristic function of) an infinite binary tree T_e and for every nonempty Π^0_1 class $\mathcal{C}$ there is an e such that $\mathcal{C} = [\Phi_e] = [T_e]$. Hint: For each e consider the r.e. set W_e viewing its elements as binary strings σ. We now form a recursive tree T_e by putting in the empty string at stage 0 and then at stage $s > 0$ exactly those strings τ of length s with no $\sigma \subseteq \tau$ in $W_{e,s}$ unless there are none (equivalently $\cup\{[\sigma] | \sigma \in W_{e,s}\} = 2^\omega$), in which case we declare all immediate successors of strings in T_e of length $s-1$ to be in T_e as well. Note that T_e is uniformly recursive. For one direction prove that each T_e is infinite (and so $[T_e]$ is a nonempty Π^0_1 class). For the other direction, if $\mathcal{C}$ is a nonempty Π^0_1 class then the set $\{\sigma | [\sigma] \cap \mathcal{C} = \emptyset\}$ is r.e. and so equal to some W_e. Now show that $[T_e] = \mathcal{C}$.*

We now present some important Π^0_1 classes.

Example 8 *$DNR_2 = \{f \in 2^\omega : f$ is $DNR\}$. Recall that DNR means $f(e) \neq \Phi_e(e)$. In other words, $\forall e \forall s \neg(f(e) = \Phi_{e,s}(e))$. Thus, DNR_2 is a Π^0_1 class.*

Example 9 *Let H be any recursively axiomatizable consistent theory. The class $\mathcal{C}_H$=$\{f \in 2^\omega : f$ is a complete extension of $H\}$ is a Π^0_1 class. By the assertion that f "is a complete extension of H" we mean that we have a recursive coding (Gödel numbering) φ_n of the sentences of H such that $T_f = \{\varphi_n | f(n) = 1\}$ is deductively closed, contains all the axioms of H and is consistent in the sense that there is no φ such that f assigns 1 (true) to both φ and $\neg\varphi$. The only point to make about this being a Π^0_1 class is perhaps the requirement that T_f be deductively closed. This says that for all finite sets Φ of sentences and each sentence φ_k and proof p, if p is a proof that $\Phi \vdash \varphi$ and $f(n) = 1$ for every $\varphi_n \in \Phi$ then $f(k) = 1$.*

Example 10 *If A, B are disjoint r.e. sets, then the class $S(A,B) = \{C : C \supset A \ \& \ C \cap B = \emptyset\}$ of* separating sets *C (for the pair (A,B)) is a Π^0_1 class as is obvious from its definition: $S = \{C : \forall n(n \in A \to n \in C \ \& \ n \in B \to n \notin C)\}$. Since $A, B \in \Sigma_1$ this is a Π^0_1 formula.*

We can view a Π^0_1 class as the solution set to the problem of finding an f that satisfies the defining condition for the class. Equivalently, the problem is finding a path f through the corresponding tree T. For the above examples the problems are to construct a DNR_2 function, a complete consistent extension of H and a separating set for A and B, respectively. If we choose our theory

H and our disjoint r.e. sets A and B correctly then the three problems and so the Π^0_1 classes (and the $[T]$ for the corresponding trees) are equivalent in the sense that a solution to (path through) any one of them computes a solution for (path in) each of the others. Suitable choices for H and (A,B) are Peano arithmetic, PA, ??define before?? and (V_0,V_1) where $V_0=\{e:\Phi_e(e)=0\}$ and $V_1=\{e:\Phi_e(e)=1\}$. For these choices, the problems are also universal in the sense that a solution to any one of them computes a path through any infinite recursive binary tree and hence a solution to any problem specified by a nonempty Π^0_1 class.

Theorem 11 *If T is an infinite recursive binary tree and f is a member of any of the three Π^0_1 classes DNR_2, $\mathcal{C}_{PA}$ or $S(V_0,V_1)$ described above then there is a path $g\in[T]$ with $g\leq_T f$.*

Proof. We first prove the theorem for $S(V_0,V_1)$. Suppose T is an infinite recursive binary tree. We begin by defining disjoint r.e. sets A and B such that any $f\in S(A,B)$ computes a path in T. We then show how to compute a path in (any) $S(A,B)$ from one in $S(V_0,V_1)$.

We know that $\{\sigma|T_\sigma$ is finite$\}$ is r.e. so suppose it is W_e. We let $A_0=\{\sigma|\exists s(\sigma\hat{\ }0\in W_{e,s}\ \&\ \sigma\hat{\ }1\in W_{e,s}\}$ (the σ such that we "see" that $T_{\sigma\hat{\ }0}$ is finite before we "see" that $T_{\sigma\hat{\ }1}$ is finite) and $A_1=\{\sigma|\exists s(\sigma\hat{\ }1\in W_{e,s}\ \&\ \sigma\hat{\ }0\in W_{e,s}\}$ (the σ such that we "see" that $T_{\sigma\hat{\ }1}$ is finite before we "see" that $T_{\sigma\hat{\ }0}$ is finite). It is clear that $A_0\cap A_1=\emptyset$. Let $C\in S(A_0,A_1)$ and define D a path in T by recursion. We begin with $\emptyset\in D$. If $\sigma\in D$ then we put $\sigma\hat{\ }C(\sigma)$ into D. We now argue by induction that if $\sigma\in D$ then T_σ is infinite: If T_σ is infinite then at least one of $T_{\sigma\hat{\ }0}$ and $T_{\sigma\hat{\ }1}$ is infinite. If both are infinite there is nothing to prove so suppose that $T_{\sigma\hat{\ }0}$ is finite but $T_{\sigma\hat{\ }1}$ is infinite. In this case, it is clear from the definition that $\sigma\in A_0$ and so $C(\sigma)=1$ and we put $\sigma\hat{\ }1$ into D to verify the induction hypothesis. In the other case, $\sigma\in A_1$, $C(\sigma)=0$ and we put $\sigma\hat{\ }0$ into D with $T_{\sigma\hat{\ }0}$ infinite as required.

Now we see how to compute a $C\in S(A_0,A_1)$ from any $D\in S(V_0,V_1)$. By the $s-m-n$ theorem ?? there is a recursive functions h such that $\forall n(n\in A_i\Leftrightarrow h(n)\in V_i)$. We now let $C(n)=D(h(n)$. It is easy to see that $C\in S(A_0,A_1)$ as required. Thus $S(V_0,V_1)$ is universal in the desired sense.

We now only have to prove that we can compute a member of $S(V_0,V_1)$ from any DNR_2 function f and from any complete extension P of PA. For the first, simply note that if $f\in DNR_2$ then $f\in S(V_0,V_1)$: If $e\in V_0$ then $\Phi_e(e)=0$ and so $f(e)=1$ as required. On the other hand, $e\in V_1$ then $\Phi_e(e)=1$ and so $f(e)=0$ as required.

Finally, suppose P is complete extension of PA. Define $C(n)=1$ if P declares the sentence $\exists s(n\in V_{0,s}\ \&\ \forall t<s(n\notin V_{1,s}))$ to be true and 0 otherwise. Note that if $n\in V_0$ then there is a least $s\in\mathbb{N}$ such that $n\in V_{0,s}$. This fact is then provable in PA (computation is essentially a proof). Similarly, for each $t<s$, $n\notin V_{1,t}$ since $n\in V_0$ and so $C(n)=1$ as required. On the other hand, if $n\in V_1$ then there is a least $s\in\mathbb{N}$ such that $n\in V_{1,s}$ and for each $t<s$, $n\notin V_{0,t}$ since $n\in V_1$ and so PA proves that $\exists s(n\in V_{1,s}\ \&\ \forall t<s(n\notin V_{0,s}))$. As P is a

consistent extension of PA, it cannot then prove that $\exists s(n \in V_{0,s} \ \& \ \forall t < s(n \notin V_{1,s}))$ and so $C(n) = 0$ as required.

Exercise 12 *Show that the degree classes $\mathbf{DNR}_2$, $\mathbf{C}_{PA}$ and $\mathbf{S}(\mathbf{V}_0, \mathbf{V}_1)$ consisting of the degrees in each of the corresponding Π^0_1 classes are all the same.*

■

As every DNR function is obviously nonrecursive (Proposition **??**), none of these three classes have recursive members. So in particular there are no recursive complete extension of PA and there is no recursive separating set for (V_0, V_1).

Thinking of Π^0_1 classes as problems that ask for solutions, the natural question is how complicated must solutions be or how simple can they be. In the (in some sense uninteresting) case that there is only one path in T (or only finitely many) we can say everything about their degrees.

Proposition 13 *If a recursive binary tree T has single path that path is recursive. In fact, any isolated path ??define?? on a recursive tree is recursive.*

In general for arbitrary T one easy answer to the question is that there are always solutions recursive in $0'$.

Exercise 14 *Show that every nonempty Π^0_1 class has a member recursive in $0'$. Hint: it is immediate for the separating classes.*

It is not hard to say a bit more.

Proposition 15 *Every infinite recursive binary tree T has a path of r.e. degree. In fact, the leftmost path P in T has r.e. degree.*

We, in fact, can significantly improve the result of Exercise 14. The Low Basis Theorem below (Theorem 18) gives the best answer with the notion of simplicity of the desired solution measured by its jump class. It is called a *basis theorem* as we say that a class $\mathcal{C}$ is a *basis* for a collection of problems (sets) if every problem (set) in the collection has a solution (member) in $\mathcal{C}$. Theorem 19 gives another basis result in terms of domination properties and Theorem 21 one in terms of solutions not computing given (nonrecursive) sets.

To prove each of these theorems we use the notion of forcing $\mathcal{P}$ whose conditions are basically infinite recursive binary trees T with usual notion of subtree as extension (simply a subset). To make the definition of our required function V recursive, we explicitly specify a stem τ for each tree such that every $\rho \in T$ is compatible with σ. Thus our conditions p are pairs (T, τ) with T an infinite recursive binary tree and $\tau \in T$ such that $(\forall \rho \in T)(\rho \subseteq \tau \text{ or } \tau \subseteq \rho)$. We say that $(T, \tau) \leq_{\mathcal{P}} (S, \sigma)$ if $T \subseteq S$ and $\tau \supseteq \sigma$. Of course, $V((T, \tau)) = \tau$. If $p = (T, \tau)$ and $\sigma \supseteq \tau$, we use p_σ to denote the condition (T_σ, σ).

The complexity of this notion of forcing depends on the representation or indexing used for the infinite recursive binary trees. While,at one end we could use the recursive listing of Exercise 7, it would then be more difficult to describe various operations on trees that determine subtrees in the natural sense but do not obviously produce an index of the type required. In this case we would also want to define the subtree relation $T \subseteq S$ in terms of $[T] \subseteq [S]$ which would then be a Π^0_2 relation (Exercise 16) and so only recursive in $0''$.

Exercise 16 *If e and i are indices for infinite binary recursive trees T and S then the relation $[T] \subseteq [S]$ is Π^0_2, and, in fact,it is Π^0_2 complete.*

At the other end, we can simply use indices for recursive functions that define infinite binary trees. While this set is only recursive in $0''$ (because it takes $0''$ to decide if an index is one for a recursive tree), operations on trees become easy to implement on the indices. On this set of indices, the standard subtree relation $T \subseteq S$ is then Π^0_1 and so recursive in $0'$. We adopt this representation of trees for our notion of forcing. In fact, while the notion of forcing is then only $0''$-recursive, some of what we want to do can be done recursively in $0'$ by analyzing the required density functions. As an example, we have the following Lemma.

Lemma 17 *There is a density function f for the class $V_n = \{(T, \tau)| \ |\tau| \geq n\}$ of dense sets in $\mathcal{P}$ which is recursive in $0'$.*

Proof. Given $p = (T, \tau) \in P$ and $n \in \mathbb{N}$, Lemma 6 tells us that we can find a $\sigma \in T$ ($\sigma \supseteq \tau$) of length $m \geq n$ such that T_σ is infinite. Clearly $p_\sigma = (T_\sigma, \sigma) \in P$ and $V(p_\sigma) \geq n$. ■

Theorem 18 (Low Basis Theorem, Jockusch and Soare) *If T is a recursive infinite binary tree then it has a low path, i.e. there is a $G \in [T]$ with $G' \equiv_T 0'$. Equivalently, if $\mathcal{C}$ is a nonempty Π^0_1 class, then it has a low member. Moreover, we can compute such a path uniformly recursively in $0'$ and the index for T or the class.*

Proof. As usual we want to show that the sets of conditions deciding the jump ($D_n = \{p|\Phi_n^{V(p)}(n) \downarrow$ or $(\forall q \leq_{\mathcal{P}} p)(\Phi_n^{V(q)}(n) \uparrow)\}$) are dense and provide a density function $f \leq_T 0'$ that also tells us in which way $f(p, n)$ is in D_n. By Lemma **??** starting with condition $p_0 = (T, \emptyset)$ we can meet these sets as well as the V_n by a generic sequence recursive in $0'$ and so construct a $G \in [T]$ with $G' \equiv_T 0'$ as required.

Given an $p = (T, \tau) \in P$ and an n, we cannot use our usual strategy of asking for a $\sigma \in T$ ($\sigma \supseteq \tau$) such that $\Phi^\sigma_n(n) \downarrow$ and then taking say p_σ as $f(p, n)$ because T_σ may be finite. Instead we ask if $\hat{T} = \{\sigma \in T|\Phi^\sigma_n(n) \uparrow\}$ is infinite. This question can be answered by $0'$ by Lemma 6. If so, we let $f(p, n) = (\hat{T}, \tau)$ and note that we have satisfied the second clause of the definition of D_n as well as guaranteed that $\Phi^G_n(n) \uparrow$ for every $G \in [\hat{T}]$ including, of course, the generic G we are constructing. If not, then clearly there is a $k \geq |\tau|$ such that $\Phi^\sigma_n(n) \downarrow$ for every $\sigma \in T$ of length k. T_σ must be infinite for one of these σ as T is infinite. Again by Lemma 6, $0'$ can find such a σ and we then set $f(p, n) = p_\sigma$. In this case, it is clear that we have satisfied the first clause of D_n and $\Phi_n^G(n)\downarrow$ for every $G \in [T_\sigma]$.

The assertion about members of the corresponding Π_1^0 classes as well as the uniformity claim in the theorem are now immediate. ■

Note that we cannot make a similar improvement to Proposition 15. Any element of DNR_2, C_{PA} or $S(V_0, V_1)$ of r.e. degree has degree $0'$.??

We next give a different answer to how simple a path we can construct on an arbitrary infinite recursive binary tree. Now the notion of simplicity is specified in terms of domination properties.

Theorem 19 ($\mathbf{0}'$-dominated Basis Theorem) *If T is an infinite recursive binary tree, then there is an $G \in [T]$ such that every $f \leq_T G$ is dominated by some recursive function.*

Proof. We use the same notion of forcing with new dense sets. In place of the D_n we have $E_n = \{(T,\tau) | (\exists x)(\forall \sigma \in T)(\Phi_n^\sigma(x)\uparrow$ or $(\forall x)(\exists k)(\forall \sigma \in T)_{|\sigma|=k}(\Phi_n^\sigma(x)\downarrow)\}$. To see that the E_n are dense consider any condition $p = (T,\tau)$. If there is an x such that $S = \{\sigma \in T | \Phi_n^\sigma(x)\uparrow\}$ is infinite then choose such an x and S. The desired extension of p in E_n is then (S,τ). Note that in this case, $\Phi_n^G(x)\uparrow$ for any $G \in [S]$. If there is no such x, then, by Lemma 6, $p = (T,\tau)$ satisfies the second clause in E_n and is itself the witness to density. Note that in this case Φ_n^G is total for any generic G. Indeed, we can now also define a recursive function h which dominates Φ_e^G for any $G \in [T]$: To compute $h(x)$ find a k such that $(\forall \sigma \in T)_{|\sigma|=k}(\Phi_n^\sigma(x)\downarrow$. This is a recursive procedure since by our case assumption there is always such a k. Now set $h(x) = \max\{\Phi_n^\sigma(x) | \sigma \in T$ and $|\sigma| = k\} + 1$. This function clearly dominates Φ_n^G for any $G \in [T]$. ■

Exercise 20 *Show that we may find a G as in Theorem 19 with $G'' \leq_T 0''$.*

We next turn to finding paths in trees which are simple in the sense that they do not compute some given (nonrecursive) set C or, more generally, any of some countable collection C_i of nonrecursive sets.

Theorem 21 (Cone Avoidance, Jockusch and Soare) *If T is an infinite recursive binary tree and $\{C_i\}$ is a sequence of nonrecursive sets, there is an $A \in [T]$ such that $C_i \not\leq_T A$ for all i.*

Proof. We modify the proof of density of the E_n of Theorem 19 to get $E_{n,m}$ that guarantee that $\Phi_n^G \neq C_m$. We let $E_{n,m} = \{(T,\tau) | (\exists x)(\forall \sigma \in T)(\Phi_n^\sigma(x)\uparrow$ or $(\exists x)(\Phi_n^\tau(x)\downarrow \neq C_m(x))$ or $(\forall x)(\exists k)(\forall \sigma_0, \sigma_1 \in T)_{|\sigma_0|=k=|\sigma_1|}(\Phi_n^{\sigma_0}(x)\downarrow = \Phi_n^{\sigma_1}(x)\downarrow)\}$. Given any condition (T,τ) we first extend it to $q = (S,\sigma) \in E_n$. If we satisfy the first clause of E_n we satisfy the same clause in $E_{n,m}$. Otherwise, we satisfy the second clause of E_n. We now ask if there are ρ_0,$\rho_1 \in S$ with $\rho_i \supseteq \sigma$ and an x such that the S_{ρ_i} are infinite and $\Phi_n^{\rho_0}(x)\downarrow \neq \Phi_n^{\rho_1}(x)\downarrow$. If so, we choose $i \in \{0,1\}$ such that $\Phi_n^{\rho_i}(x) \neq C_m(x)$ and take q_{ρ_i} as our extension of q (and so of p) which gets into $E_{n,m}$ by satisfying the second clause. If not, we claim that q itself satisfies the third clause of $E_{n,m}$ and that there is a recursive function h such that $\Phi_n^G = h$ for every $G \in [S]$. As for q satisfying the third clause of $E_{n,m}$, consider any x and note that it already satisfies the second clause of E_n. If there were infinitely many k such there are $\sigma_0, \sigma_1 \in T$ of length k with $\Phi_n^{\sigma_0}(x) \downarrow \neq \Phi_n^{\sigma_1}(x) \downarrow$ then we would have been in the previous case as there would then be infinitely many $\sigma \in T$ with $\Phi_n^{\sigma}(x) \downarrow \neq C_m(x)$. Thus we may define $h(x)$ by finding a k as in the third clause of $E_{n,m}$ and setting $h(x) = \Phi_n^{\sigma}(x)$ for any σ in S of length k. We then have that $\Phi_n^G = h$ for every $G \in [S]$. As C_m is not recursive,$\Phi_n^G \neq C_m$ for any $G \in [S]$ and so we also satisfy the requirements of the theorem. ■

Exercise 22 *Show that we may construct a G as required in Theorem 21 such that $G \leq_T 0'' \oplus (\oplus_i C_i)$ and indeed uniformly.*

Exercise 23 *For one nonrecursive C instead of a countable set of C_i show that we may construct a G as required in Theorem 21 such that $G \leq_T 0''$ (but without the uniformity). Hint: use the following exercise.*

Exercise 24 *Prove that for any infinite recursive binary tree T there are $G_0, G_1 \in [T]$ such that any $C \leq_T G_0, G_1$ is recursive. Moreover, we may find such G_i with $G_i'' \equiv_T 0''$.*

Exercise 25 *Nonempty Π^0_1 classes such as DNR_2 that have no recursive member are called* special *Π^0_1 classes. Prove that any such class has $2^{\aleph_0}$ many members.*

Exercise 26 *Strengthen some of previous theorems producing a path in T with some property to producing $2^{\aleph_0}$ many if T is special.*

2 Finitely branching trees

Also trees recursive in A (f). Relativizations.

Finitely branching trees, f-bounded, (recursively bounded) essentially the same as binary (recursive) binary trees relativize results to f.

Given a recursive recursively bounded tree can get recursive binary tree which has same paths up to degree by padding.

The sets of paths through infinitely branching trees $T \subseteq \omega^{<\omega}$ correspond to closed sets in Baire space. Even for recursive trees finding paths is much more complicated in this setting. Whether such trees even have paths is a Π^1_1 complete question. As for a basis theorem, one says that if there is a path then there is one recursive in the complete Π^1_1 set $\mathcal{O}$.??

Reference for low basis theorem: Jockusch, Soare " Degrees of Members of Π^0_1 Classes" Pacific J. Math 40(1972) 605-616
Pseudo jump operators: Jockusch, Shore " Pseudo jump operators I: the r.e. case" Trans. Amer. Math. Soc. 275 (1983) 599-609; " Pseudo-jump operators II: Transfinite iterations, hierarchies, and minimal covers" JSL 49 (1984) 1205-1236