Generating Sets for the Recursively Enumerable Turing Degrees

Klaus Ambos-Spies, Steffen Lempp,[*] *and Theodore A. Slaman*[†]

Department of Mathematics
and Computer Science
University of Heidelberg
D-69120 Heidelberg, Germany
ambos@math.uni-heidelberg.de

Department of Mathematics
University of Wisconsin–Madison
Madison, WI 53706-1388, USA
lempp@math.wisc.edu

Department of Mathematics
University of California–Berkeley
Berkeley, CA 94720-3840, USA
slaman@math.berkeley.edu

Abstract

We give an example of a subset of the recursively enumerable Turing degrees which generates the recursively enumerable degrees using meet and join but does not generate them using join alone.

1 Introduction

One of the recurrent themes in the area of the recursively enumerable (r.e.) degrees has been the study of the *meet operator*. While, trivially, the partial ordering of the

[*]Lempp was partially supported by NSF grant DMS-0140120 and a Mercator Guest Professorship of the Deutsche Forschungsgemeinschaft.

[†]Slaman was partially supported by the Alexander von Humboldt Foundation and by National Science Foundation Grant DMS-9988644.

r.e. degrees is an upper semi-lattice, i.e., the join operator is total, the meet of two incomparable r.e. degrees may or may not exist (Lachlan (1966), Yates (1966)). The asymmetry between joins and meets is further illustrated by the fact that, by Sacks' splitting theorem (Sacks (1963)), every nonzero r.e. degree is join-reducible, i.e., is the join of two lesser degrees, whereas there are both, meet-reducible (branching) and meet-irreducible (nonbranching), incomplete r.e. degrees (Lachlan (1966)).

The existence of meets and the failure of meets are densely distributed in the partial ordering of the r.e. degrees. So Fejer (1983) showed that the nonbranching degrees are dense while Slaman (1991) showed that the branching degrees are dense. Similarly, every interval of the r.e. degrees contains an incomparable pair of degrees without meet (Ambos-Spies (1984)) and an incomparable pair of degrees with meet (Slaman (1991)). That, actually, the lack of meets is more common than the existence of meets has been demonstrated by Ambos-Spies (1984) and, independently, by Harrington (unpublished) who showed that, for any nonzero, incomplete r.e. degree $\mathbf{a}$, there is an incomparable degree $\mathbf{b}$ such that the meet of $\mathbf{a}$ and $\mathbf{b}$ does not exist, but also that there is such a degree $\mathbf{a}$ such that, for any incomparable degree $\mathbf{b}$, the meet of $\mathbf{a}$ and $\mathbf{b}$ does not exist. More evidence, that the failure of meets is more typical than their existence, was given by Jockusch (1985) who showed that, given r.e. degrees $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ such that $\mathbf{a}$ and $\mathbf{b}$ are incomparable and $\mathbf{c}$ is the meet of $\mathbf{a}$ and $\mathbf{b}$, none of these degrees is e-generic.

Another way to look at the join and meet operators in the r.e. degrees is to study *generating sets*, i.e., sets of r.e. degrees which generate all the recursively enumerable degrees under (finitely many applications of) join and meet. The question now arises naturally whether both the join operation and the meet operation are needed here. As observed in Ambos-Spies (1985), the above results on nonbranching degrees easily imply that the join operation is indeed necessary, namely there is a subset of the recursively enumerable degrees which generates all recursively enumerable degrees using join and meet but not using meet alone. Ambos-Spies, however, left open the question of whether the meet operation is necessary (see Ambos-Spies (1985), Problem 1). The above mentioned negative results on meets by Fejer (1983), Ambos-Spies (1984) and Jockusch (1985) may suggest a negative answer to this question. More evidence in this direction has been obtained by Ambos-Spies (1985) who showed that any generating set intersects any notrivial initial segment of the r.e. degrees and, more recently, by Ambos-Spies, Ding and Fejer (unpublished) who showed that any generating set generates the high r.e. degrees using join alone. Despite this negative evidence, in this paper, we answer Ambos-Spies' question affirmatively by the following

Theorem 1.1 *There exists a subset* $\mathbf{G}$ *of the recursively enumerable Turing degrees which generates the recursively enumerable Turing degrees using meet and join but does not generate them using join alone.*

Proof. Our theorem follows by our technical result, Theorem 2.1, below, using a nonconstructive definition of the set $\mathbf{G}$. Fix the recursively enumerable degree $\mathbf{a}$ from Theorem 2.1. Let $\{\mathbf{x}_n\}_{n\in\omega}$ be a (noneffective) enumeration of all recursively enumerable degrees $\leq \mathbf{a}$. We now define a (noneffective) sequence of recursively enumerable degrees $\mathbf{0} = \mathbf{y}_0 \leq \mathbf{y}_1 \leq \mathbf{y}_2 \leq \cdots < \mathbf{a}$ as follows: Set $\mathbf{y}_0 = \mathbf{0}$. Given $\mathbf{y}_n < \mathbf{a}$, check whether $\mathbf{y}_n \cup \mathbf{x}_n = \mathbf{a}$. If not, then set $\mathbf{y}_{n+1} = \mathbf{y}_n \cup \mathbf{x}_n$. Otherwise, let $\mathbf{b}$ be the recursively enumerable degree given by Theorem 2.1 using $\mathbf{x} = \mathbf{x}_n$ and $\mathbf{y} = \mathbf{y}_n$, and set $\mathbf{y}_{n+1} = \mathbf{y}_n \cup \mathbf{b}$. Finally, we define

$$\mathbf{G} = \{\mathbf{x} \mid \mathbf{x} \not\leq \mathbf{a} \text{ or } \exists n\, (\mathbf{x} \leq \mathbf{y}_n)\}.$$

By Theorem 2.1, the degree $\mathbf{a}$ is clearly not the join of any finite set of degrees in $\mathbf{G}$. On the other hand, fix any recursively enumerable degree $\mathbf{x}$ and assume $\mathbf{x} \notin \mathbf{G}$. Then $\mathbf{x} \leq \mathbf{a}$, and so $\mathbf{x} = \mathbf{x}_n$ for some $n \in \omega$. Since $\mathbf{x} \notin \mathbf{G}$, we have $\mathbf{x} \not\leq \mathbf{y}_{n+1}$ and so $\mathbf{x} \cup \mathbf{y} = \mathbf{a}$ for $\mathbf{y} = \mathbf{y}_n$. Fix $\mathbf{b}$, $\mathbf{c}$, $\mathbf{d}$, and $\mathbf{e}$ as in Theorem 2.1. Then $\mathbf{x} = \mathbf{b} \cup (\mathbf{d} \cap \mathbf{e})$ where all of $\mathbf{b}$, $\mathbf{d}$, and $\mathbf{e}$ are in $\mathbf{G}$ since $\mathbf{b} \leq \mathbf{y}_{n+1}$ and $\mathbf{d}, \mathbf{e} \not\leq \mathbf{a}$. $\square$

2 The technical theorem and some intuition for its proof

Starting with this section, we will prove the technical theorem needed to establish Theorem 1.1:

Theorem 2.1 *There is a nonrecursive, recursively enumerable set A such that for every pair of recursively enumerable sets X and Y, if X and Y are recursive in A and A is recursive in XY then one of the following conditions holds.*

1. *A is recursive in Y.*
2. *There are recursively enumerable sets B, C, D, and E such that*
   (a) *X has the same Turing degree as BC,*
   (b) *D and E are not recursive in A and the degree of C is the infimum of the degrees of DC and EC, and*
   (c) *A is not recursive in BY.*

2.1 Requirements and simple strategies

We disassemble the statement of Theorem 2.1 into requirements as follows. First, A must be nonrecursive and so we must satisfy all the requirements $\Theta \neq A$, where Θ is a recursive function.

Second, for each X, Y, $\Lambda_{a,x}$, $\Lambda_{a,y}$, and $\Lambda_{xy,a}$, we associate the principal equations $\Lambda_{a,x}(A) = X$, $\Lambda_{a,y}(A) = Y$, and $\Lambda_{xy,a}(XY) = A$. We can satisfy our requirement on X, Y, $\Lambda_{a,x}$, $\Lambda_{a,y}$, and $\Lambda_{xy,a}$ in any of several ways. If the principal equations are not valid then our requirement is satisfied.

Anticipating that the principal equations actually are valid, we enumerate the sets B, C, D, and E and recursive functionals $\Gamma_{x,b}$, $\Gamma_{x,c}$, and $\Gamma_{bc,x}$. We ensure that $\Gamma_{x,b}(X) = B$, $\Gamma_{x,c}(X) = C$, and $\Gamma_{bc,x}(BC) = X$. Now, our requirement is satisfied in one of two ways.

For every recursive functional Θ_{by}, if $\Theta_{by}(BY) = A$ then there is a $\Delta_{y,a}$, which we enumerate during our construction, such that $\Delta_{y,a}(Y) = A$. If there is a $\Delta_{y,a}$ such that $\Delta_{y,a}(Y) = A$, then again our requirement is satisfied.

Otherwise, we ensure that every instance of the following family of requirements is satisfied.

1. For all Θ_a, $\Theta_a(A) \neq D$ and $\Theta_a(A) \neq E$.
2. For all Ψ_{cd} and Ψ_{ce}, if $\Psi_{cd}(CD) = \Psi_{ce}(CE)$ then there is a Ξ_c such that $\Xi_c(C) = \Psi_{cd}(CD) = \Psi_{ce}(CE)$.

2.2 Strategies

2.2.1 Making $\Theta \neq A$: $\sigma_0(\Theta)$

We ensure that $\Theta \neq A$ by choosing a number n, keeping n out of A until seeing $\Theta(n) = 0$, and then enumerating n into A. This strategy σ_0 is one of the standard methods to satisfy requirements of this form.

2.2.2 Measuring whether the equations hold: $\sigma_1(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a})$

Now, we consider the more complicated requirements. Suppose that X, Y, $\Lambda_{a,x}$, $\Lambda_{a,y}$, and $\Lambda_{xy,a}$ are given.

Our strategy $\sigma_1(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a})$ approximates if the principal equations hold for X, Y, $\Lambda_{a,x}$, $\Lambda_{a,y}$, and $\Lambda_{xy,a}$. We will abbreviate by σ_1 the strategy $\sigma_1(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a})$ and use similar conventions throughout this section. Essentially, σ_1 measures expansionary stages in the approximation to these equalities. For technical reasons, explained below, σ_1 waits for something more than simple expansion. In the following, a_1 and a_2 are variables of the strategy which enumerates pairs (a_1, a_2) into a list of pairs of witnesses.

1. If a_1 is undefined and it is possible to do so, choose a value for a_1 that is larger than $\lambda_{a,x}(A, x)[s]$ for every x previously mentioned in the construction during a σ_1-expansionary stage. Let x_1 be the smallest number x such that $\lambda_{a,x}(A, x)[s]$ is greater than a_1. Suspend the enumeration of any functionals associated with B, C, D or E. (We may assume that we have not enumerated any computations from BC of X at arguments greater than or equal to x_1.)

   Wait until the first stage s such that $(\Lambda_{xy,a}(XY) \upharpoonright a_1 + 1 = A \upharpoonright a_1 + 1)[s]$, and $(\Lambda_{a,x}(A) = X)[s]$ and $(\Lambda_{a,y}(A) = Y)[s]$ on all numbers less than or equal to the maximum of $\lambda_{xy,a}(XY)[s] \upharpoonright a_1 + 1$. At this stage, we let a_2 equal the supremum of $(\lambda_{a,x}(A) = X)[s]$ and $(\lambda_{a,y}(A) = Y)[s]$ on all numbers less than or equal to the maximum of $\lambda_{xy,a}(XY)[s] \upharpoonright a_1 + 1$. We enumerate the pair (a_1, a_2) into our list and let the strategies of lower priority resume the enumeration of any functionals associated with B, C, D or E. (The (a_1, a_2) notation will be convenient below.) Go to Step 2.
2. At the next stage when σ_1 is active, we say that a_1 is undefined, and go to Step 1.

Consider the possibilities. The strategy σ_1 could reach a limit in Step 1. In this case, one of the principal equations fails and the requirement is satisfied.

If σ_1 does not reach a limit in Step 1 then it enumerates infinitely many stable pairs and has no other effect on the construction.

For the remainder of this section, we assume that σ_1 does not reach a finite limit and that all subsequent strategies act during the stages when σ_1 enumerates a new pair. We call such stages σ_1-expansionary.

2.2.3 Computations between B, C, and X: $\sigma_2(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a})$

Our strategy σ_2 builds functionals $\Gamma_{x,b}$ and $\Gamma_{x,c}$ and ensures that if the principal equations are valid then for each n there are infinitely many s such that $(\Gamma_{x,b}(X, n) = B(n))[s]$ and $(\Gamma_{x,c}(X, n) = C(n))[s]$. This, combined with our preserving A, B, and C, will be sufficient to conclude that B and C are recursive in X.

We ensure their correctness by imposing the constraint on all lower priority strategies τ that if $\Gamma_{x,b}(X, n)[s]$ or $\Gamma_{x,c}(X, n)[s]$ is defined while τ acts then τ cannot enumerate n into B or C, respectively, during that stage.

Similarly, we ensure that X is recursive in BC by enumerating a functional $\Gamma_{bc,x}$ and ensuring that if the principal equalities hold then for all n, $\Gamma_{bc,x}(BC, n) = X(n)$ during infinitely many stages of the construction.

We have complete freedom to define the uses of these functionals, but the construction does not require a subtle decision. During σ_1-expansionary stages, we enumerate new computations into $\Gamma_{bc,x}$. If n enters X during stage s and $\Gamma_{bc,x}(BC, n) = 0[s]$ then we must enumerate a number less than or equal to $\gamma_{bc,x}(BC, n)[s]$ into either B or C. We set the uses of these functions to be larger than any number previously used in the construction.

In the case of maintaining $\Gamma_{bc} = X$, we also have the freedom to decide which of B and C to change when recording a change in X. The choice made is irrelevant to σ_2. In our construction, we will leave the decision to the highest priority strategy for which it is relevant. See the discussion of the strategies of type σ_6.

2.2.4 Making C the infimum of CD and CE: $\sigma_3(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Psi_{cd}, \Psi_{ce})$

We will use the branching strategies from Fejer (1982) and attempt to make the degree of C equal to the infimum of the degrees of CD and CE. Suppose that Ψ_{cd} and Ψ_{ce} are given and let σ_3 denote our branching strategy associated with this pair. Then, σ_3 enumerates a functional Ξ_c. Say that s is σ_3-*expansionary* if and only if the least number n such that $(\Psi_{cd}(CD, n) \neq \Psi_{ce}(CE, n))[s]$ is larger than at any earlier stage.

First, during stage s, if there is an n such that $\Xi_c(C, n)[s]$ is defined and a strategy of priority less than or equal to that of σ_3 enumerates numbers into C, D, or E so that neither $(\Psi_{cd}(CD, n) = \Xi_c(C, n))[s]$ nor $(\Psi_{ce}(CE, n) = \Xi_c(C, n))[s]$, then σ_3 must enumerate a number less than or equal to $\xi_c(C, n)[s]$ into C. (We will have to argue that this enumeration is compatible with C's being recursive relative to X.)

Second, if s is σ_3-expansionary then for the least n such that $\xi_c(C,n)$ is not defined, we choose a value for $\xi_c(C,n)[s]$ which is larger than any number previously mentioned in the construction and enumerate a computation into Ξ_c setting $\Xi_c(C[s],n)=\Psi_{cd}(CD[s],n)$ with use $\xi_c(C,n)[s]$.

If $\Psi_{cd}(CD)=\Psi_{ce}(CE)$ then there will be infinitely many σ_3-expansionary stages. Since we will be preserving computations from CD and CE, the converse will also be true. So, if $\Psi_{cd}(CD)\neq\Psi_{ce}(CE)$, then σ_3 will act finitely often. Otherwise, it produces a functional Ξ_c from C which is defined infinitely often to agree with the common value of $\Psi_{cd}(CD)$ and $\Psi_{ce}(CE)$. Again, since we are preserving the sets that we construct, this will be sufficient to ensure that $\Xi_c(C)$ is equal to this common value.

We will assume that there are infinitely many σ_3-expansionary stages and describe the appropriate strategies to follow. These strategies act only during σ_3-expansionary stages.

Instability in C and compatibility between σ_2 and σ_3. The strategies σ_3 introduce an instability to the initial segments of C. Namely, suppose that a strategy τ enumerates a number c into C. Then, c enters both CD and CE and could change the common value of $\Psi_{cd}(CD,c_1)$ and $\Psi_{ce}(CE,c_1)$. In response, τ enumerates $\xi_c(C,c_1)$ into C, possibly changing C at a number less than c, and the effect can propagate. We call the set of numbers that enter C in this way the cascade initiated by c.

When combined with the strategy to ensure that C is recursive in X, the branching strategies make it difficult to enumerate any number at all into C. If $\Gamma_{x,c}(X,m)$ is defined then we cannot enumerate any c into C unless we can be sure that the instability in C will not propagate to the point of requiring that m enter C. We will use some of the ideas of Slaman (1991) to work within this constraint.

Definition 2.2 A number c is *σ_3-stable at stage s* if for all m, if $\xi_c(C,m)[s]<c$ then either $\psi_{cd}(CD,m)<c$ or $\psi_{ce}(CE,m)<c$.

We note that if c is σ_3-stable at stage s then any cascade initiated by a number greater than or equal to c during stage s does not include any number less than c. To prove this claim, consider the recursive propagation of a cascade initiated by a number greater than or equal to c, and let m be the first number less than c to appear in the cascade. Earlier in the propagation of the cascade, C would have to change below the minimum of $\psi_{cd}(CD,m)[s]$ and $\psi_{ce}(CE,m)[s]$. By the stability of c, this minimum is less than c and we have contradicted m's being first.

Thus, if c is stable and $\Gamma_{x,c}(X,c)[s]$ is not defined, then we can enumerate c into C and respect both σ_2 and σ_3. We will design all the strategies to follow so that they enumerate only stable numbers into C. Of course, there is no such constraint on B, since B is not constructed to be branching.

2.2.5 Making $\Theta_a(A) \neq D$ and $\Theta_a(A) \neq E$: $\sigma_4(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Theta_a)$ and $\sigma_5(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Theta_a)$

We use a variation, σ_4, on the basic Friedberg strategy to ensure that $\Theta_a(A) \neq D$. (The strategy σ_5 for E is similar.) We choose n larger than any number previously mentioned in the construction and constrain n from entering D. We wait for a stage s such that $(\Theta_a(A, n) = 0)[s]$. By our assumption, s will be σ_1 and σ_3-expansionary.

Then, we enumerate n into D and constrain any number less than s from entering any set under construction other than D. We note that these actions are consistent. Since we did not enumerate anything into A and A's computation of X exists on a longer interval than ever before, X cannot change at any number m such that $\Gamma_{bc,x}(BC, m)[s]$ is defined. So, σ_2 will not require any change in B or C. Since s is σ_3-expansionary, both $\Psi_{cd}(CD)[s]$ and $\Psi_{ce}(CE)[s]$ were defined (before we changed D), agreed on a longer interval than ever before, and agreed with $\Xi_c(C)[s]$ where $\Xi_c(C)[s]$ was defined. Since we did not change E, $\Psi_{ce}(CE)[s]$ is still equal to $\Xi_c(C)[s]$ where the latter is defined and σ_3 does not require any change in C.

2.2.6 If $\Theta_{by}(BY) = A$ then $\Delta_{y,a}(Y) = A$: $\sigma_6(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Theta_{by})$

Now we come to the crux of the proof of Theorem 2.1. Our strategy σ_6 must either diagonalize $\Theta_{by}(BY)$ against A, or it must determine that A is recursive in Y. Since Y is an arbitrary set below A, both cases are possible.

In the context of the construction, σ_6 can assume that every stage is σ_1- and σ_2-expansionary. Now, this implies that if we enumerate a number a into A during stage s, there is a later stage t during which A recomputes X and Y and either $X[s] \neq X[t]$ or $Y[s] \neq Y[t]$ and the least m at which the inequality occurs is less than $\lambda_{xy,a}(XY, a)[s]$. In other words, if we change A then one of X or Y must change in order to correct $\Lambda_{xy,a}(XY)$.

To establish $\Theta_{by}(BY) \neq A$, at least once we would have to change A without having to change B and without Y's having changed. This could happen, since the change in A could be recorded in X and we could record the change in X (for the sake of σ_3) in C. If, on the other hand, this diagonalization is not possible then we must conclude that A is recursive in Y. The conclusion is not unreasonable since every change in A results in a change in Y. But, if even one change in A results in a change in X, then we must be able to record that change in C.

We have reached the technical problem to be solved to prove the theorem. For every A-change allowed by σ_6, if it results in a change in X, then we must be able to record that change in X and in C. Now remember that we are only able to enumerate numbers into C which are σ_3-stable during their stage of enumeration. So we must ensure that changes in X can be recorded in C by the enumeration of such numbers. This is the purpose in our strategy σ_6.

Configurations. Consider a possible stage-s situation as depicted in Figure 1. In this picture, we illustrate a number a_1 which we intend to enumerate into A; x_1

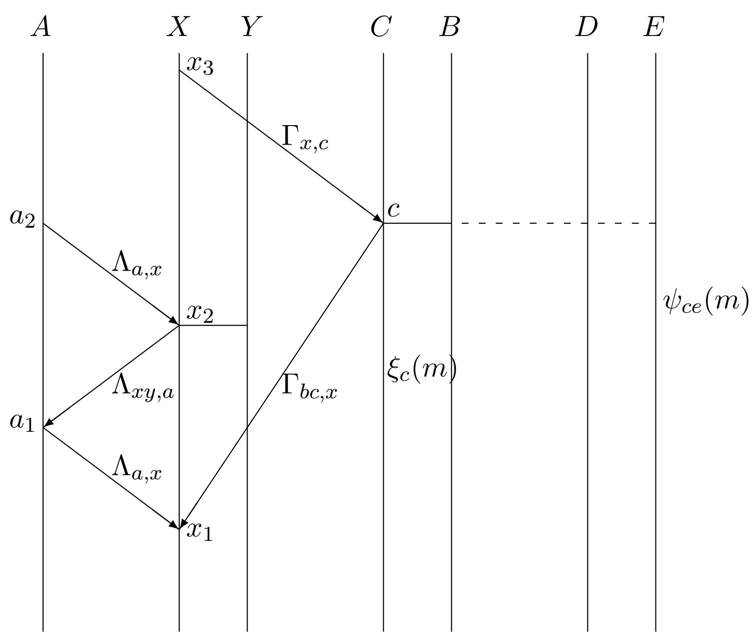

Fig. 1: Configuration for a_1

is the least number x such that a_1 is less than $\lambda_{a,x}(A,x)[s]$ and hence the least number which might enter X when a_1 enters A; x_2 is equal to $\lambda_{xy,a}(XY,a_1)[s]$ and a_1's entering A would cause a change in XY below x_2; a_2 is the supremum of $\lambda_{a,x}(A,x_2)[s]$ and $\lambda_{a,y}(A,x_2)[s]$, so our preserving A on numbers less than a_2 will preserve the relationship between a_1, x_1, and x_2; c is $\gamma_{bc,x}(BC,x_1)[s]$ and so enumerating c into C would correct the computation of X from BC on every argument at which X might change; we intend that c be σ_3-stable at stage s and so, for any m, if $\xi_c(m)$ is less than c then one of $\psi_{cd}(CD,m)[s]$ or $\psi_{ce}(CE,m)[s]$ is also less than c; finally, x_3 is $\gamma_{x,c}(X,c)$, the use of X's computation of C at argument c. Note that we can preserve the relationships between these numbers by preserving A up to a_2, and B,C,D, and E up to c. (We can keep x_3 above x_2 by enumerating our functions so that the uses of new computations are at least as great as the uses of earlier computations at the same argument.)

Suppose that, in this situation, we were to enumerate a_1 into A and Y did not change below x_2 to record that fact (for example, if $Y \not\geq A$). Then X would change below x_2, allowing us to enumerate the σ_3-stable number c into C and thereby correct BC's computations for any change in X allowed by a_1's entering A.

We say that the situation depicted in Figure 1 is a *σ_1-configuration for a_1*. Anticipating that $\Theta_{by}(BY) = A$, we must ensure that for all but a recursive set of numbers a, if a enters A then it does so in the role of a_1 with a configuration as above.

Generating configurations. Though configurations seem artificial at first, they are very common. In fact, a new configuration can be produced during every σ_1-expansionary stage.

Note that, by the constraint imposed by σ_3, at the beginning of every stage t, for every m, if $\Xi_c(C,m)[t]$ is defined then one of $\Psi_{cd}(CD,n)[t]$ and $\Psi_{ce}(CE,n)[t]$ is also defined with the same value.

Now consider a σ_1-expansionary stage s. Let c be the least strict upper bound on the range of $\xi_c(C)[s]$. By the observation above, c is σ_3-stable. Since s is σ_1-expansionary, σ_1 enumerated a pair (a_1,a_2) related as in Figure 1. Further, by the choice of a_1 and a_2, a_1 is greater than $\lambda_{a,x}(A,x)[s]$ for every x such that $\Gamma_{bc,x}(BC,x)$ has ever been defined. Consequently, there is no computation in $\Gamma_{bc,x}[s]$ which applies to the argument x_1. Then, we can use σ_2 to enumerate computations into $\Gamma_{x,b}$, $\Gamma_{x,c}$, and $\Gamma_{bc,x}$ so that $\gamma_{bc,x}(BC,x_1) > c$, and $\gamma_{x,c}(X,c) > \lambda_{xy,a}(X,a_1)$. In Figure 1, $\lambda_{xy,a}(X,a_1)$ would be x_2 and $\gamma_{x,c}(X,c)$ would be x_3. In short, at the beginning of each stage s, the whole of $\xi_c[s]$ is stable and at a σ_2-expansionary stage, we can use the pair (a_1,a_2) enumerated by σ_1 and enumerate new computations into our functionals to extend $\xi_c[s]$ to a configuration for a_1.

Restricting to configured numbers. We satisfy the requirement

$$\Theta_{by}(BY) = A \implies \Delta_{y,a}(Y) = A$$

as follows.

1. If a is undefined and we are not preserving an inequality between $\Theta_{by}(BY)$ and A, then choose a value for a such that a is larger than any value previously chosen and such that there is a configuration for a. We restrain a from entering A and preserve the configuration for a until we find a stage s such that $(\Theta_{by}(BY, a) = 0)[s]$ and that $\Lambda_{a,y}(A)[s]$ is equal to $Y[s]$ on all numbers less than or equal to $\theta_{by}(BY, a)[s]$. At stage s, we go to Step 2.

2. Let a_0 be the largest number that we have previously enumerated as *allowed to enter A with a certified configuration* (or 0 if there is no such number). For each number n between a_0 and a, if $n \notin A[s]$ then we restrain n from ever entering A. We enumerate a into the set of numbers still allowed to enter A and we say that the current configuration for a together with the current computation $(\Theta_{by}(BY, a) = 0)[s]$ is the certified configuration associated with a during stage s.

   For all strategies τ of lower priority, require that if τ enumerates a into A during stage t then the certified configuration associated with a during stage s must also exist during stage t. That is, the initial segments of the sets involved in the configuration for a and the computation from BY must not have changed. Further, if during the next σ_1-expansionary stage u it happens that $Y \restriction \theta_{by}(BY)[t]$ is equal to $Y \restriction \theta_{by}(BY)[u]$ (i.e., Y did not change) then the change in X is recorded in C and we preserve the inequality $\Theta_{by}(BY, a) \neq A(a)$ by preserving the appropriate initial segments of the sets under construction.

   We say that a is now undefined and go to Step 1.

Clearly, if for every Θ_{by} we can conclude that $\Theta_{by}(BY) \neq A$ then we have satisfied our requirement. Further, each of the strategies will act at most finitely often and cause little trouble to the rest of the construction.

Suppose this is not the case and consider the effects of the above strategy σ_6 when $\Theta_{by}(BY) = A$. Assume that the strategy is never injured (or be willing to accept finitely many exceptions). Then σ_6 enumerates an infinite increasing sequence of numbers a as still being allowed to enter A. Call this sequence the σ_6-stream of numbers. For each number n, if n is not an element of the σ_6-stream then n is an element of A if and only if n is enumerated into A before any number greater than n is enumerated into the σ_6-stream. Thus, the restriction of A to the numbers not in the σ_6-stream is recursive. Now, consider a number a which is enumerated into the σ_6-stream, say at stage s_a.

By the action of σ_6, for every number a in the σ_6-stream, if a enters A during a stage s greater than or equal to the one during which a was enumerated in the sequence, then the configuration existent when a was enumerated into the stream by σ_6 is still available during stage s. Since $\Theta_{by}(BY) = A$ and there are infinitely many σ_1-expansionary stages, it must be the case that during the interval from stage t to the next σ_1-expansionary stage after t, Y changed below $\lambda_{xy,a}(XY, a)[s]$. Thus, if a is an element of the σ_6-stream and is enumerated into the stream at stage s, then a enters A no later than the first stage u after stage s such that

$Y[u] \upharpoonright \lambda_{xy,a}(XY, a)[s] = Y \upharpoonright \lambda_{xy,a}(XY, a)[s]$. It follows that A is recursive relative to Y.

2.2.7 One sequence $(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a})$

If we were to work only with one sequence $(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a})$ or equivalently have only one strategy of type σ_1, then our construction would be particularly simple. We would start with the strategies of type σ_1 and σ_2 and follow them with the strategies of type σ_3, σ_4, σ_5, and σ_6 (as well as nonrecursiveness strategies σ_0). In the simplest case, one of these strategies could have a finite outcome and we could conclude that our requirement is satisfied. In the simplest case, the σ_1-strategy could have a finite outcome and we could conclude that our requirement is satisfied. If not, then either each of the σ_6-strategies would have a finite outcome and we would have satisfied all the necessary requirements $\Theta_{by}(BY) \neq A$, or one of our strategies would have an infinite outcome and we would conclude that our requirement is satisfied by virtue of A's being recursive relative to Y.

In this last case, how can we conclude that A is not recursive? The σ_6-strategy that generates an infinite stream associates with these numbers an infinite stream of certified configurations. The strategy to ensure $\Theta \neq A$ chooses a number a, preserves its configuration and preserves its certification by preserving B and preserving enough of A to ensure that Y cannot change on any relevant number. If at a later stage t it happens that $\Theta(a)[t] = 0$ then the diagonalization strategy enumerates a into A.

3 The global construction

In the previous section, we analyzed the combinations of the strategies associated with a single sequence $(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a})$. We now combine the strategies for all possible such sequences and thereby present a proof of Theorem 2.1.

3.1 Interactions between σ-strategies

In fact, there is very little interaction between the strategies associated with different sequences. For the most part, their constraints apply to different B's, C's, D's, and E's and so are mutually compatible. The only set that they have in common is A and the only constraints that they put upon A are the finite ones associated with successful diagonalization and the infinite one constraining the enumeration of new elements of A to the conditions of a σ_6-stream.

Consider a strategy τ constrained to work within an infinite σ_6-stream. The new constraint on τ is that at stage t, τ can enumerate element a into A if and only if a was enumerated into the σ_6-stream at a stage $s < t$ and the configuration associated with a during the stage s still exists during stage t.

If τ is associated with a σ_0-strategy ensuring $\Theta \neq A$ then when τ chooses its number with which to diagonalize, τ chooses that number a from the σ_6-stream.

While τ is waiting for $\Theta(a) = 0$, τ preserves enough of A to preserve the σ_6-configuration associated with a. If $\Theta(a)$ is seen to be equal to 0 then τ can enumerate a into A consistently with the constraint of σ_6.

By inspection of the strategies, this is the only way by which numbers enter A and so we need not make many internal changes within our families of strategies.

3.2 The tree of strategies

We fix recursive enumerations $(\Theta^i : i \in \omega)$ of all recursive functionals relative to the empty set, $((X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}) : i \in \omega)$ of all sequences as described in σ_1, and, for each i, $(\Psi^{i,j} : j \in \omega)$, $(\Theta^{i,j}_a : j \in \omega)$, and $(\Theta^{i,j}_{by} : j \in \omega)$ of all recursive functionals with one set argument. Of course, the enumerations $(\Psi^{i,j} : j \in \omega)$, $(\Theta^{i,j}_a : j \in \omega)$, and $(\Theta^{i,j}_{by} : j \in \omega)$ need not depend on i, but the notation will be convenient below. Let $((i,j) : i,j \in \omega)$ be a recursive ordering of $\omega \times \omega$ of order type ω. We will assume that for all j, i is less than or equal to the position of (i,j) in this ordering.

We define a tree T of sequences of pairs of strategies and outcomes using recursion. We will also order the immediate extensions of each node from left to right. Ordering by first difference, we have a left to right ordering for all incompatible sequences in T. As usual, shorter nodes or nodes to the left will be assigned higher priority than those below or to the right. For $\eta \in T$, we will speak of the extensions of η as being below η in T. We start with the empty sequence as an element of T.

Suppose that $\eta = ((\tau_k, o_k) : k < \ell)$ is an element of T.

Definition 3.1 Suppose $k < \ell$ and τ_k is of the form $\sigma_1(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a})$. Then

1. τ_k is *in effect at* η if and only if o_k is Π_2, and
2. τ_k is *unresolved at* η if and only if for all j,

$$(\sigma_6(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_{by}), \Pi_2) \notin \eta.$$

Definition 3.2 Suppose $k < \ell$ and τ_k is of the form $\sigma_6(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Theta_{by})$. Then, τ_k is *in effect at* η if and only if o_k is Π_2.

Strategies below η in T are based on the assumption that there will be infinitely many expansionary stages for the σ_1- and σ_6-strategies in effect at η. If τ_k is unresolved at η then no strategy in η has determined that A is recursive relative to Y.

Let i_{max} be the largest i such that there is a $k < \ell$ such that τ_k is equal to $\sigma_1(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a})$. (If there is no such i, let $i_{max} = -1$.)

Case 1. If there is a pair (i^*, j^*) among the first i_{max} many such pairs such that

A. $\sigma_1(X^{i^*}, Y^{i^*}, \Lambda^{i^*}_{a,x}, \Lambda^{i^*}_{a,y}, \Lambda^{i^*}_{xy,a})$ is in effect at η, unresolved at η, and

B. one of $\sigma_2(X^{i^*}, Y^{i^*}, \Lambda^{i^*}_{a,x}, \Lambda^{i^*}_{a,y}, \Lambda^{i^*}_{xy,a})$, $\sigma_3(X^{i^*}, Y^{i^*}, \Lambda^{i^*}_{a,x}, \Lambda^{i^*}_{a,y}, \Lambda^{i^*}_{xy,a}, \Psi^{i^*,j^*})$, $\sigma_4(X^{i^*}, Y^{i^*}, \Lambda^{i^*}_{a,x}, \Lambda^{i^*}_{a,y}, \Lambda^{i^*}_{xy,a}, \Theta^{i^*,j^*}_{a})$, $\sigma_5(X^{i^*}, Y^{i^*}, \Lambda^{i^*}_{a,x}, \Lambda^{i^*}_{a,y}, \Lambda^{i^*}_{xy,a}, \Theta^{i^*,j^*}_{a})$, or $\sigma_6(X^{i^*}, Y^{i^*}, \Lambda^{i^*}_{a,x}, \Lambda^{i^*}_{a,y}, \Lambda^{i^*}_{xy,a}, \Theta^{i^*,j^*}_{by})$ does not appear in (the first coordinate of an element of) η,

then let (i, j) be the least such (i^*, j^*). We determine the immediate successor of η in T by the first of the following conditions which applies.

1. If $\sigma_2(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a})$ does not appear in η then
$$\eta^\frown(\sigma_2(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}), \Pi_1) \in T.$$
2. If $\sigma_3(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Psi^{i,j})$ does not appear in η then
$$\eta^\frown(\sigma_3(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Psi^{i,j}), \Sigma_2) \in T,$$
$$\eta^\frown(\sigma_3(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Psi^{i,j}), \Pi_2) \in T,$$
and the Σ_2-extension of η is to the right of the Π_2-extension.
3. If $\sigma_4(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_a)$ does not appear in η then
$$\eta^\frown(\sigma_4(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_a), \Sigma_1) \in T,$$
$$\eta^\frown(\sigma_4(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_a), \Pi_1) \in T,$$
and the Π_1-extension of η is to the right of the Σ_1-extension.
4. If $\sigma_5(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_a)$ does not appear in η then
$$\eta^\frown(\sigma_5(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_a), \Sigma_1) \in T,$$
$$\eta^\frown(\sigma_5(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_a), \Pi_1) \in T,$$
and the Π_1-extension of η is to the right of the Σ_1-extension.
5. If $\sigma_6(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_{by})$ does not appear in η then
$$\eta^\frown(\sigma_6(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_{by}), \Sigma_2) \in T,$$
$$\eta^\frown(\sigma_6(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a}, \Theta^{i,j}_{by}), \Pi_2) \in T,$$
and the Σ_2-extension of η is to the right of the Π_2-extension.

Case 2. If there is no such pair (i, j) as above then we set $i = i_{max} + 1$ and determine the immediate successor of η in T as follows.

1. If $\sigma_0(\Theta^i)$ does not appear in η then
$$\eta^\frown(\sigma_0(\Theta^i), \Sigma_1) \in T,$$
$$\eta^\frown(\sigma_0(\Theta^i), \Pi_1) \in T,$$
and the Σ_1-extension of η is to the left of the Π_1-extension.

2. Otherwise,

$$\eta^\frown(\sigma_1(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}), \Sigma_2) \in T,$$
$$\eta^\frown(\sigma_1(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}), \Pi_2) \in T,$$

and the Σ_2-extension of η is to the right of the Π_2-extension.

3.2.1 η-configurations

Notice that if $\eta \in T$ then there is a unique strategy σ such that σ appears as the first component in the last element of the immediate successors of η.

Definition 3.3 Suppose that η is an element of T.

1. Let

$$\{\sigma_1(X^{i_j}, Y^{i_j}, \Lambda^{i_j}_{a,x}, \Lambda^{i_j}_{a,y}, \Lambda^{i_j}_{xy,a}) : j < \ell_1\}$$

be the sequence of σ_1-strategies σ in effect at η. Then an *η-configuration for a_1* is a finite initial segment A and the sets associated with these strategies such that for each $j < \ell_1$, there is a $\sigma_1(X^{i_j}, Y^{i_j}, \Lambda^{i_j}_{a,x}, \Lambda^{i_j}_{a,y}, \Lambda^{i_j}_{xy,a})$-configuration for a_1 within this initial segment.

2. We say that an η-configuration is *certified* if in addition to the above, for every σ_6-strategy τ_k which is in effect at η, the computation setting $\Theta_{by}(BY, a) = 0$ has not changed since the stage during which τ_k enumerated the configuration for a as certified.

For example, if η has only one σ_1-strategy $\sigma_1(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a})$ with a Π_2-outcome, then an η-configuration for a_1 is the same as a $\sigma_1(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a})$-configuration for a_1, as described in Figure 1. With n such strategies, an η-configuration is described by n copies of Figure 1, one for each strategy and involving the sets associated with that strategy, sharing a common value for a_1.

The i_jth component of an η-configuration for a_1 is the initial segment of A, B^{i_j}, C^{i_j}, D^{i_j}, and E^{i_j} which makes up the $\sigma_1(X^{i_j}, Y^{i_j}, \Lambda^{i_j}_{a,x}, \Lambda^{i_j}_{a,y}, \Lambda^{i_j}_{xy,a})$-configuration for a_1.

3.3 The construction

We organize our construction by stages s, where s is greater than or equal to 1. Each stage s is divided into at most s many substages t, where t is also greater than or equal to 1. We proceed as follows during stage s.

Let $\eta[s, 0]$ equal the empty sequence.

Given $\eta[s, t-1]$ with t less than or equal to s, let σ be the strategy which appears in the first component of the immediate successors of $\eta[s, t-1]$. We may assume that σ has been assigned a number a_1 and an $\eta[s, t-1]$-configuration for a_1 during an earlier stage and that no component of that configuration has changed since the stage during which it was assigned. (Otherwise, a strategy simply ends the stage since by its hypothesis, it will eventually be assigned such a number a_1 by a σ_2- or σ_5-strategy as described below.)

We follow the instructions of σ, which depend on its type as described below. At the end of its action, either σ ends stage s and we go to stage $s+1$ with $t=0$ or σ determines a value for $\eta[s,t]$. In the second case, if t is less than s then we continue with substage $t+1$ of stage s.

We adapt the pure strategies described in the previous section to work within the full construction as follows.

3.3.1 Adding an A-diagonalization strategy σ_0

Suppose that σ is a diagonalization strategy $\sigma_0(\Theta^i)$ to ensure that $\Theta^i \neq A$.

If $\Theta^i(a_1)[s]$ is not equal to 0 then we restrain any number from entering any set involved in our η-configuration for a_1. We let $\eta[s,t]$ be $\eta[s,t-1]^\frown(\sigma,\Pi_1)$.

If $\Theta^i(a_1)[s]$ is equal to 0 and a_1 is not an element of $A[s]$, then we enumerate a_1 into A and end stage s.

If $\Theta^i(a_1)[s]$ is equal to 0 and we have already enumerated a_1 into A, then we let $\eta[s,t]$ be $\eta[s,t-1]^\frown(\sigma,\Sigma_1)$.

3.3.2 Adding a $\sigma_1(X,Y,\Lambda_{a,x},\Lambda_{a,y},\Lambda_{xy,a})$

Suppose that σ is $\sigma_1(X^i,Y^i,\Lambda^i_{a,x},\Lambda^i_{a,y},\Lambda^i_{xy,a})$. We alter the pure σ_1-strategy (described in the previous section) in the following way.

First, we measure σ-expansions in terms of stages when $\eta[s,t-1]$ is active in the construction. Second, while waiting for a σ-expansionary stage, we preserve the $\eta[s,t-1]$-configuration for a_1.

If s is not σ-expansionary in the above sense then let $\eta[s,t]$ be $\eta[s,t-1]^\frown(\sigma,\Sigma_2)$.

Otherwise, we enumerate the pair (a_1,a_2) as in the pure σ_1-strategy, we let $\eta[s,t]$ be $\eta[s,t-1]^\frown(\sigma,\Pi_2)$, and we cancel all strategies on nodes to the right of $\eta[s,t]$.

3.3.3 Adding a $\sigma_2(X,Y,\Lambda_{a,x},\Lambda_{a,y},\Lambda_{xy,a})$

Suppose that σ is $\sigma_2(X^i,Y^i,\Lambda^i_{a,x},\Lambda^i_{a,y},\Lambda^i_{xy,a})$. By the definition of T and the previous paragraph, we may assume that the last strategy mentioned in $\eta[s,t-1]$ is of the form $\sigma_1(X^i,Y^i,\Lambda^i_{a,x},\Lambda^i_{a,y},\Lambda^i_{xy,a})$ and that s is expansionary for that strategy (i.e., the sequence $\eta[s,t-1]$ ends with the pair $(\sigma_1(X^i,Y^i,\Lambda^i_{a,x},\Lambda^i_{a,y},\Lambda^i_{xy,a}),\Pi_2)$).

We let $\eta[s,t]$ be $\eta[s,t-1]^\frown(\sigma,\Pi_1)$, and we alter the pure σ_2-strategy in the following way.

First, we may need to change BC to record a change in X. If during the previous stage s' during which σ_2 acted, some strategy associated with an extension of η enumerated a number a into A, then let μ be the node in T associated with that strategy. There are two cases to consider. In the first case, there is no strategy $\sigma_6(X^i,Y^i,\Lambda^i_{a,x},\Lambda^i_{a,y},\Lambda^i_{xy,a},\Theta_{by})$ above μ with $B^i=B$ and $C^i=C$ which certified a. In this case, we record the change in X by changing B accordingly.

Otherwise, if Y did not change below $\lambda^i_{xy,a}(a)$ between stage s' and the current stage, then X must have changed there. This allows us to record all changes in X by enumerating c into C, where c is the number depicted in Figure 1, and we do so without changing B.

Next, let (a_1, a_2) be the pair just enumerated by $\sigma_1(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a})$. We extend the definitions of the functionals $\Gamma_{x,b}$, $\Gamma_{x,c}$, and $\Gamma_{bc,x}$ so that we have a $\sigma_1(X^i, Y^i, \Lambda^i_{a,x}, \Lambda^i_{a,y}, \Lambda^i_{xy,a})$-configuration for a_1 and thus an $\eta[s,t]$-configuration for a_1. If there is a μ such that $\eta[s,t] \subseteq \mu$, μ was active during a previous stage, the set of strategies in effect at μ is equal to the set of strategies in effect at $\eta[s,t]$, and μ does not currently have an $\eta[s,t]$-configuration assigned to it, then we assign the configuration for a_1 to the leftmost and shortest such μ (that is, the one of highest priority). We cancel all strategies to the right of μ and end stage s.

3.3.4 Adding a $\sigma_3(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Psi_{cd}, \Psi_{ce})$

Our only alteration to the pure σ_3-strategy is to make it measure expansionary stages taking into account only those stages during which it is active. For $\sigma = \sigma_3(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Psi_{cd}, \Psi_{ce})$ we let $\eta[s,t]$ be $\eta[s,t-1]^\frown(\sigma, \Pi_2)$ if s is σ-expansionary and $\eta[s,t-1]^\frown(\sigma, \Sigma_2)$ otherwise.

3.3.5 Adding a $\sigma_4(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Theta_a)$ or a $\sigma_5(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Theta_a)$

We use the pure σ_4- and σ_5-strategies without change. For a σ_4-strategy σ, we let $\eta[s,t]$ be $\eta[s,t-1]^\frown(\sigma, \Sigma_1)$ if the diagonalization witness n has been enumerated into D and $\eta[s,t-1]^\frown(\sigma, \Sigma_2)$ otherwise. (For a σ_5-strategy σ, D is replaced by E.)

3.3.6 Adding a $\sigma_6(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Theta_{by})$

We do not alter the first phase of the pure σ_6-strategy. We start with a number a for which we have a certified $\eta[s,t-1]$-configuration. We restrain a from entering A and preserve its configuration. We wait for a stage s such that $(\Theta_{by}(BY, a) = 0)[s]$. If the current s is not such a stage then, for $\sigma = \sigma_6(X, Y, \Lambda_{a,x}, \Lambda_{a,y}, \Lambda_{xy,a}, \Theta_{by})$ we set

$$\eta[s,t] = \eta[s,t-1]^\frown(\sigma, \Sigma_2).$$

Otherwise, we set

$$\eta[s,t] = \eta[s,t-1]^\frown(\sigma, \Pi_2).$$

If there is an a such that we wait forever for an s such that $(\Theta_{by}(BY, a) = 0)[s]$, then $\Theta_{by} \neq A$ and the requirement is satisfied. Otherwise, according to the pure σ_6-strategy, we should restrict the enumeration of numbers into A to those which appear in the stream it generates.

We must describe how the numbers in that stream are distributed to the strategies associated with nodes extending $\eta[s,t] = \eta[s,t-1]^\frown(\sigma, \Pi_2)$. For this, we make the same alteration which we made for the σ_2-strategies.

We can assume that σ has been assigned a certified $\eta[s, t-1]$-configuration for a number a_1. When σ finds a computation setting $\Theta_{by}(BY, a_1) = 0$ and for which there is are $\Lambda_{a,y}(A)$ computations agreeing with Y below $\theta_{by}(BY, a_1)$, then we say that these computations certify the $\eta[s, t-1]$-configuration for a_1 with respect to σ. Thus, a_1 now has a certified $\eta[s, t]$-configuration.

If there is a μ such that $\eta[s, t] \subseteq \mu$, μ was active during a previous stage, the set of strategies in effect at μ is equal to the set of strategies in effect at η, and μ does not currently have an $\eta[s, t]$-configuration assigned to it, then we assign the configuration for a_1 to the leftmost and shortest such μ (that is, the one of highest priority). We cancel all strategies η' to the right of μ and end stage s. If there is no such μ then we cancel all strategies to the right of $\eta[s, t]$.

3.4 Analyzing the construction

Let η^∞ be the path through T such that

1. for infinitely stages s and infinitely many substages t, $\eta[s, t]$ is a subsequence of η^∞, and
2. for at most finitely many stages s and substages t, $\eta[s, t]$ is to the left of η^∞.

Following convention, we say that η^∞ is the true path of the construction.

Lemma 3.4 *The true path η^∞ is an infinite path in T.*

Proof. Suppose that η is a finite initial segment of η^∞. We will argue that there is a proper extension of η which is also contained in η^∞.

Note that T is a finite branching tree. Consequently, if there are infinitely many s during which η acts and does not end the stage, then there is a leftmost proper extension which acts infinitely often and the claim is proven.

There are four cases in which η peremptorily ends a stage during which it acts. Firstly η might end with a strategy which ends the stage since it is not assigned a number, which can happen at most finitely often in a row by the strategy's assumption on outcomes of strategies above it. Next, η might end with a strategy for making $\Theta \neq A$, in which case this strategy can end the stage at most once without being initialized. Otherwise, either η ends with a σ_2-strategy, or it ends with a σ_6-strategy, and, in both cases, η allocates an η-certified configuration to a strategy below it. But, if μ is eligible to be assigned a configuration by η then μ must have been active during an earlier stage of the construction. If η were to end all but finitely many of the stages during which it finds a new certified η-configuration, then there can only be finitely many such μ's. Eventually, every such μ will have a certified η-configuration assigned to it. But then the next time that η finds a new certified η-configuration it will not end the stage and some proper extension of η will be active.

Lemma 3.5 *For each finite $\eta \subset \eta^\infty$, the following conditions hold.*

1. *We cancel η during the last stage s_η during which there is a t such that $\eta[s_\eta, t]$ is to the left of η.*
2. *We let η act infinitely often.*
3. *During every stage greater than or equal to s_η, we respect all of the constraints imposed by η during any earlier stage.*

Proof. Routine. □

Lemma 3.6 *Our construction satisfies all of the requirements of Section 2.1.*

Proof. This follows as in the analysis of the individual strategies in Section 2.2. □

Theorem 2.1 follows directly from Lemma 3.6.

References

Ambos-Spies, K. (1984). On pairs of recursively enumerable degrees. *Trans. Amer. Math. Soc. 283*(2), 507–531.

Ambos-Spies, K. (1985). Generators of the recursively enumerable degrees. In *Recursion theory week (Oberwolfach, 1984)*, Volume 1141 of *Lecture Notes in Math.*, pp. 1–28. Berlin: Springer.

Fejer, P. A. (1982). Branching degrees above low degrees. *Trans. Amer. Math. Soc. 273*(1), 157–180.

Fejer, P. A. (1983). The density of the nonbranching degrees. *Ann. Pure Appl. Logic 24*(2), 113–130.

Jockusch, Jr., C. G. (1985). Genericity for recursively enumerable sets. In *Recursion theory week (Oberwolfach, 1984)*, Volume 1141 of *Lecture Notes in Math.*, pp. 203–232. Berlin: Springer.

Lachlan, A. H. (1966). Lower bounds for pairs of recursively enumerable degrees. *Proc. London Math. Soc. (3) 16*, 537–569.

Sacks, G. E. (1963). *Degrees of unsolvability.* Princeton, N.J.: Princeton University Press.

Slaman, T. A. (1991). The density of infima in the recursively enumerable degrees. *Ann. Pure Appl. Logic 52*(1-2), 155–179. International Symposium on Mathematical Logic and its Applications (Nagoya, 1988).

Yates, C. E. M. (1966). A minimal pair of recursively enumerable degrees. *J. Symbolic Logic 31*, 159–168.