

Extension of Embeddings in the Computably Enumerable Degrees

Theodore A. Slaman
and
Robert I. Soare *

1 Introduction

Gödel's Incompleteness Theorem [Göd31] and his subsequent work on computable functions [Göd34] exhibited undecidability in the most familiar mathematical settings, even in elementary number theory. Following Gödel, there has been an intensive study of noncomputable sets arising in ordinary mathematics. The *computably enumerable (c.e.)* sets (those which can be listed by a computable method) are of particular interest because they appear naturally in most branches of mathematics and there is a noncomputable c.e. set. Thus, they play a key role in well-known results such as Gödel's incompleteness theorem, the unsolvability of the word problem for finitely presented groups, and the unsolvability of Hilbert's tenth problem on Diophantine equations.

Turing [Tur39] introduced the notion of a set of natural numbers A being computable *relative* to set B , written $A \leq_T B$. The equivalence class of A under \equiv_T is the (*Turing degree*) of A . Friedberg [Fri57] and independently Mučnik [Muc56] solved Post's problem by producing a noncomputable incomplete c.e. set, and in doing so introduced the finite injury priority method. Let $\mathbf{0}$ denote the degree of the computable sets and $\mathbf{0}'$ denote the degree of K , the complete c.e. set, and let $\mathcal{R} = (\mathbf{R}, <, \mathbf{0}, \mathbf{0}')$. Lower case boldface letters $\mathbf{a}, \mathbf{b}, \mathbf{c}, \dots, \mathbf{x}, \mathbf{y}, \mathbf{z}$ denote c.e. degrees. The Friedberg-Mučnik theorem may be regarded as the first and easiest extension of embeddings result for \mathcal{R} . It asserts that $(\exists \mathbf{x})[\mathbf{0} < \mathbf{x} < \mathbf{0}']$.

*The first author was supported by National Science Foundation Grants DMS 89-02437 and DMS-9212022-A01, and SERC Visiting Fellowship Research Grant ("Leeds Recursion Theory Year 1993/94") No. GR/H 91213, and the second author by National Science Foundation Grants DMS 91-06714, and DMS 94-00825.

Let $\mathbf{a} \vee \mathbf{b}$ denote the join of degrees \mathbf{a} and \mathbf{b} which always exists in \mathbf{R} . Sacks [Sac63] introduced a stronger form of the finite injury method to prove his splitting theorem, $(\forall \mathbf{a} > \mathbf{0})(\exists \mathbf{x}, \mathbf{y})[\mathbf{x} \mid \mathbf{y} \ \& \ \mathbf{x} \vee \mathbf{y} = \mathbf{a}]$. Sacks [Sac64] then developed a version of the infinite injury priority method to prove the density theorem, $(\forall \mathbf{a}, \mathbf{b})[\mathbf{a} < \mathbf{b} \rightarrow (\exists \mathbf{x})[\mathbf{a} < \mathbf{x} < \mathbf{b}]]$. Shoenfield [Sho65] then conjectured that for all finite upper semi-lattices $P \subseteq Q$ with least and greatest elements, any embedding of P into the upper semi-lattice $(\mathbf{R}, <, \vee, \mathbf{0}, \mathbf{0}')$ can be extended to an embedding of Q into the same. However, Lachlan [Lac66] and independently Yates [Yat66] refuted Shoenfield's conjecture by constructing a *minimal pair*, namely a pair of nonzero c.e. degrees $\langle \mathbf{a}, \mathbf{b} \rangle$ such that $\mathbf{a} \wedge \mathbf{b} = \mathbf{0}$.

The next significant advance was the Lachlan nonsplitting theorem [Lac75] which asserts that the Sacks splitting and density theorems cannot be combined simultaneously, $\neg(\forall \mathbf{a}, \mathbf{b})[\mathbf{b} < \mathbf{a} \rightarrow (\exists \mathbf{x}, \mathbf{y})[\mathbf{b} < \mathbf{x}, \mathbf{y} < \mathbf{a} \ \& \ \mathbf{a} = \mathbf{x} \vee \mathbf{y}]]$. Lachlan's result was significant first because it demonstrated a new kind of nonextension phenomenon, and second because its proof introduced a new and powerful technology (the $\mathbf{0}'''$ -priority method) for constructing c.e. sets. During the 1970's and 1980's this method was further developed and applied to prove a number of deep results about \mathcal{R} such as the Harrington and Shelah theorem [HS82] that the elementary theory of \mathcal{R} is undecidable. It remained evident that \mathcal{R} admits a considerable level of algebraic analysis. In this vein, attention was directed toward embedding and extension of embeddings problems for \mathcal{R} . Virtually all the major algebraic results about \mathcal{R} can be viewed as embedding, extension of embeddings results, or nonextension of embeddings results for \mathcal{R} viewed as either a partial ordering or sometimes with partial lattice structure.

The extension of embeddings question was solved for certain related structures. For example, Fejer and Shore [FS85] solved it for the c.e. tt-degrees and wtt-degrees. Shore and Slaman [SS91] calculated the extension of embeddings theory which is common to all principal ideals in \mathcal{R} for which the top point is low_2 (i.e. $\mathbf{a}'' = \mathbf{0}''$). The method of proof in these results is to rule out certain extensions using the minimal pair method and then to realize those remaining extensions using the methods of the Sacks density theorem. However, the minimal pair method, with its associated theorems about meet and join, is not sufficient to capture all the nonextension properties of \mathcal{R} .

We solve the full extension of embeddings problem for the structure of the c.e. degrees \mathcal{R} . In §2, we state the extension criterion in terms of two relatively simple algebraic conditions on P and Q . If Condition (1) holds, then in §3 we embed P into \mathcal{R} so that there is no extension to an embedding of Q ; the proof uses a combination of elements from the proofs of the minimal pair and splitting theorems. If Condition (2) holds, then in §5 we construct

a similar counterexample to extension of embedding; the proof uses a version of the Lachlan nonsplitting technology and the $\mathbf{0}'''$ -method. If Conditions (1) and (2) both fail, then in §4 we construct the required extension; the proof uses a strengthening of the Sacks density method generalizing [HS92].

Theorem 2.2 unifies the algebraic extension and nonextension results about \mathcal{R} . (Note that several of the theorems above are corollaries of it.) Its proof unifies the principal proof methods for c.e. degrees: it shows that the minimal pair method and Lachlan nonsplitting method are complementary to the Sacks density method, when all are suitably generalized.

2 The decision procedure

We use these conventions: P and Q are finite partial orders with 0 and 1 with P contained in Q ; lower case Roman letters denote elements of P and Q ; a, b, c , and d denote elements of P ; and x, y, z, u , and v denote elements of $Q - P$.

Definition 2.1 For S a subset of Q , we define

$$\begin{aligned}\mathcal{A}(S) &= \{a \mid a \in P \ \& \ \text{for all } x \text{ in } S, a \geq x\} \\ \mathcal{B}(S) &= \{b \mid b \in P \ \& \ \text{for all } x \text{ in } S, x \geq b\}\end{aligned}$$

$\mathcal{A}(S)$ and $\mathcal{B}(S)$ are the filter and ideal in P determined by S . We write $\mathcal{A}(x)$ for $\mathcal{A}(\{x\})$ and $\mathcal{B}(x)$ for $\mathcal{B}(\{x\})$. We also write $x \geq \mathcal{B}$ if $x \geq b$ for all $b \in \mathcal{B}$ and similarly $x \leq \mathcal{A}$ if $x \leq a$ for all $a \in \mathcal{A}$.

Theorem 2.2 *Suppose that P and Q are finite partially ordered sets with 0 and 1, and that P is a sub-partial-order of Q . If P and Q satisfy either of the following two conditions (1) or (2) then there is an embedding of P into \mathcal{R} which cannot be extended to an embedding of Q into \mathcal{R} :*

$$(1) \quad (\exists x, y \in Q)[x \not\geq y \ \& \ \mathcal{B}(\mathcal{A}(y)) \subseteq \mathcal{B}(\mathcal{A}(\mathcal{B}(x)))],$$

$$(2) \quad (\exists x \in Q - P)[\mathcal{Z}(x) \neq \emptyset \ \& \ \mathcal{B}(\mathcal{A}(\mathcal{Z}(x) \cup \mathcal{B}(x))) \not\subseteq \mathcal{B}(x)], \text{ where}$$

$$\mathcal{Z}(x) = \{z \mid z \in Q - P \ \& \ x > z \ \& \ \mathcal{B}(x) \not\subseteq \mathcal{B}(\mathcal{A}(z))\}.$$

If neither of these two conditions applies then every embedding of P into \mathcal{R} can be extended to an embedding of Q into \mathcal{R} .

Corollary 2.3 *Given $\langle P, Q \rangle$, a pair of finite partially ordered sets with 0 and 1 such that $P \subseteq Q$, it is a uniformly computable condition whether every embedding of P into \mathcal{R} extends to an embedding of Q into \mathcal{R} .*

We present our proofs as priority constructions using the heuristics of the tree method. See Soare [Soa87] for this material and for notation and definitions. Upper case letters A, B, X and Y denote computably enumerable sets, ostensibly the images of a, b, x and y in some embedding of Q into \mathcal{R} . Upper case Greek letters such as Φ, Ψ or Θ denote Turing functionals. When $\Phi(A, x) = y$, we let $\varphi(A, x)$ be the largest number which is mentioned in the computation of $\Phi(A)$ at x . We say that $\varphi(A, x)$ is the *use* of $\Phi(A)$ at x . Thus, if $\Phi(A, x) = y$ and A agrees with B on all numbers less than or equal to $\varphi(A, x)$ then $\Phi(B, x) = y$. We let $f \upharpoonright x$ ($f \upharpoonright s$) denote the restriction of f to arguments $y < x$ (respectively $(y \leq x)$). If α and β are sequences, then $\alpha \hat{\ } \beta$ is the sequence with the elements of α followed by those of β , and $|\alpha|$ denotes the length of α . In addition to computably enumerable sets, we also enumerate computable functionals Γ . When we define $\Gamma(A, x) = y$ with use $\gamma(A, x)$, we enumerate a computation into the functional Γ which states that if X extends $A \upharpoonright \gamma(A, x)$ then $\Gamma(X, x) = y$.

We approximate the satisfaction of equations of the form $\Phi(U) = V$. During stage s of our construction, our approximation depends on the finite subsets $\Phi[s]$, $U[s]$ and $V[s]$ of Φ , U and V as enumerated by stage s , and we use $p[s]$ or p_s to denote the value of parameter p at the end of stage s . We say that s is $\Phi(U) = V$ -*expansionary* if the stage s approximation to the initial segment of numbers on which $\Phi(U) = V$ is larger than at any earlier stage.

3 The first nonextension theorem

Ruling out extensions by infima and suprema. In this section we prove that the first condition (1) of Theorem 2.2 leads to nonextendibility.

Theorem 3.1 *Suppose that P and Q are finite partially ordered sets with 0 and 1, and that P is a sub-partial-order of Q , and satisfies Condition (1). Then there is an embedding of P into \mathcal{R} which cannot be extended to an embedding of Q into \mathcal{R} .*

The second conjunct in (1) is equivalent to, the assertion that it is consistent to extend P to a lattice L such that

$$(3) \quad \bigvee \mathcal{B}(x) \geq \bigwedge \mathcal{A}(y),$$

where $\bigvee \mathcal{B}(x)$ represents the supremum in L of the elements $b \in \mathcal{B}(x)$ and $\bigwedge \mathcal{A}(y)$ the infimum of the elements $a \in \mathcal{A}(y)$. The details are similar to those in [SS91], using the Lachlan, Lerman and Thomason result [Soa87, p. 157] that

any finite distributive lattice can be embedded into \mathcal{R} preserving least element, and the dual result by Ambos-Spies [AS80] for embeddings preserving greatest element.

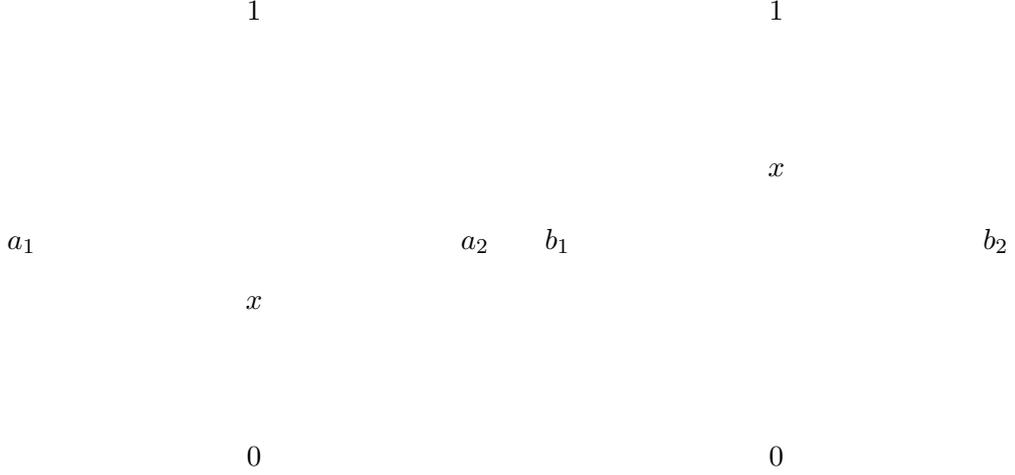


Figure 1: Two extensions ruled out by Condition (1)

Some examples. Consider the left-side example, ruled out by the Lachlan-Yates existence of a minimal pair. Here $x \not\leq 0$, *i.e.* $u = 0$ and $v = x$, $\mathcal{A}(x) = \{a_1, a_2, 1\}$, and $\mathcal{B}(\mathcal{A}(x)) = \{0\}$, $\mathcal{B}(\mathcal{A}(\mathcal{B}(0))) = \mathcal{B}(0)$, and $\mathcal{B}(0) = \{0\}$. Consequently, $\mathcal{B}(\mathcal{A}(x)) \subseteq \mathcal{B}(\mathcal{A}(\mathcal{B}(0)))$ and the extension is ruled out by Condition (1). Now consider the right-side example, ruled out by the Sacks splitting theorem. Here $x \not\leq 1$, *i.e.* $u = x$ and $v = 1$, $\mathcal{A}(1)$ is $\{1\}$ and $\mathcal{B}(\mathcal{A}(1)) = \{0, b_1, b_2, 1\}$, $\mathcal{B}(x) = \{0, b_1, b_2\}$, $\mathcal{A}(\mathcal{B}(x)) = \{1\}$, and $\mathcal{B}(\mathcal{A}(\mathcal{B}(x))) = \{0, b_1, b_2, 1\}$, so the extension is ruled out by Condition (1).

4 The extension theorem

We now restate the Extension Theorem in a more convenient form using conditions equivalent to the former ones.

Theorem 4.1 (Extension Theorem) *Suppose that P and Q are finite partially ordered sets with 0 and 1 , and that P is a sub-partial-order of Q which*

satisfy the following two conditions:

$$(4) \quad (\forall x, y \in Q)[x \not\geq y \rightarrow (\exists c \in P)(\exists d \in P)[c \geq \mathcal{B}(x) \ \& \ d \leq \mathcal{A}(y) \ \& \ c \not\geq d]], \text{ and}$$

$$(5) \quad (\forall x \in Q - P)[\mathcal{Z}(x) \neq \emptyset \rightarrow \mathcal{B}(\mathcal{A}(\mathcal{Z}(x) \cup \mathcal{B}(x))) \subseteq \mathcal{B}(x)], \text{ where}$$

$$\mathcal{Z}(x) = \{z \mid z \in Q - P \ \& \ x > z \ \& \ \mathcal{B}(x) \not\subseteq \mathcal{B}(\mathcal{A}(z))\}.$$

Then any embedding of P into \mathcal{R} extends to an embedding of Q into \mathcal{R} .

Proof. We first remark that conditions (4) and (5) are equivalent to the conjunction of the negations of (1) and (2) from Theorem 2.2. Hence, Theorem 4.1 is equivalent to the extension half of Theorem 2.2.

Intuition. To better understand (5), the reason for it, and the construction in this section, consider the adjacent diagram shown in Figure 2. (We use this diagram because it is the easiest one to illustrate the difficulties for coding Z into W and X , and the need for the partition of X into 1X and 2X . However, it is not extendible because it does not satisfy condition (5) as explained in §5.1 just after the Figure 5.1.)

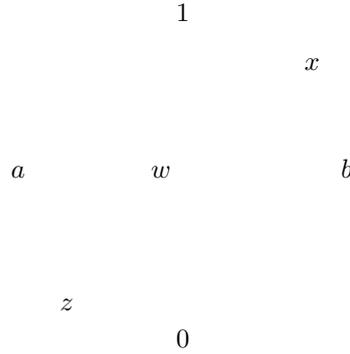


Figure 2: Diagram for coding Z into W but not into X .

Let the upper case letters represent the c.e. sets realizing the points with the corresponding lower case letters. We cannot simply code Z directly into X because then any restraint on (elements entering) X produces implied restraint on Z . An element n entering B can lift the X -restraint and hence the implied Z -restraint also. However, in this diagram we do not have $A \geq B$, so A cannot compute when the Z -restraint will drop, and thus A cannot compute

Z . Abstractly, $\mathcal{Z}(X)$ is the set of points Z where we encounter this difficulty relative to X . For those points $W \notin \mathcal{Z}(X)$ the problem does not occur, so we do code W into X . This key distinction lies at the heart of our proof of the Recovery Lemma 4.8.

To overcome this problem, we code Z into W because $Z \notin \mathcal{Z}(W)$, but we cannot and *do not* code Z into X because $Z \in \mathcal{Z}(X)$. (Therefore, the coding described below will not be transitive.) In general, this would be a fatal shortcoming and would prevent us from extending the embedding, as is the case for the diagram in Figure 3 of section 5.1. However, in the present case condition (5) saves us and says that we do not *need* to explicitly code Z into X because by (5) we can already compute Z from the join of $\mathcal{B}(X)$ (and hence from X itself) as we prove in the Recovery Lemma 4.8 and the Comparability Lemma 4.9.

This brings yet another *key difficulty*. Since $W \notin \mathcal{Z}(X)$ we *do* wish to directly code W into X . But $Z < W$, and Z was coded into W as above, and Z must *not* be coded into X . Therefore, Z and every $U \in Q - P$ will be decomposed into the disjoint union of two pieces: (i) the *indirect* coding set 2U which will code the contributions of the incomparability requirements for all α such that $U = Y_\alpha$ in (9) of requirement R_α ; and (ii) the *direct* coding set 1U which will code, for $(V, U) \in \mathcal{C}$ of (6), either V itself if $V \in P$, or merely 2V if $V \in Q - P$. Thus, both 2U and 1U will consist of finitely many disjoint components, corresponding to all such α (V respectively). For the sets in our diagram, the set 2Z will be coded into 1W but not into X , which is not allowed because $Z \in \mathcal{Z}(X)$. The set 2W will be coded into 1X because $W \notin \mathcal{Z}(X)$. We do not code 1W into X because this would inadvertently code 2Z into X .

The strategies for the requirements. Fix an embedding π of P into \mathcal{R} . For each $a \in P$ fix a c.e. set $A \in \pi(a)$. We identify P with $\pi(P)$, identify the elements of Q with the c.e. sets in P and those we are constructing for $Q - P$. We let A, B, C, D represent members of P and X, Y, Z, U, V represent members of Q (which may also be members of P). For $\mathcal{S} \subseteq P$ let $\mathbb{W}(\mathcal{S})$ be $\bigoplus\{X \mid X \in \mathcal{S}\}$, where \bigoplus is the usual join operation. Our task is to assign to each member of $Q - P$ a c.e. set X so that $X \leq_T Y$ iff $Q \models X \leq Y$. The requirements on the sets we are building can be divided into the *comparability* requirements $V \leq U$, and the *incomparability* requirements $V \not\leq U$. We mostly write $X \leq Y$ instead of $X \leq_T Y$ for $X, Y \in Q$, but after constructing the c.e. sets in $Q - P$ we sometimes use $X \leq_T Y$, particularly to compare X to a set such as $\mathbb{W}\mathcal{S}$ which may not be in Q .

Comparability requirements $V \leq U$ of type 1 and direct coding. The *comparability requirements of type 1* are those corresponding to the pairs,

$$(6) \quad \mathcal{C} = \left\{ (V, U) \left| \begin{array}{l} U \in Q - P \ \& \ Q \models V < U \\ \& \ [V \in P \vee V \notin \mathcal{Z}(U)] \end{array} \right. \right\}.$$

The remaining comparability requirements are considered in the Comparability Lemma 4.9. For $S \subseteq \omega$ let $S^{[n]} = \{ \langle x, y \rangle : y = n \}$. We will define a *tree* $T \subset \omega^{<\omega}$, whose definition begins here and is completed just before Definition 4.2. We adopt an effective indexing of all $\alpha \in \omega^{<\omega}$, identify α with its index n , and let $\omega^{[\alpha]} = \omega^{[n]}$. Let c be the cardinality of \mathcal{C} . Each new set W constructed will be decomposed into two disjoint sets 1W and 2W , where we enumerate elements in 1W for direct coding (to be explained below), and in 2W for the Sacks coding to meet an incomparability requirement of the form (9) with $W = Y_\alpha$, to be explained as we proceed.

For each i , $0 \leq i < c$, we put node $\alpha = 0^i$ in T , and assign to α the requirement,

$$(7) \quad P_\alpha : \quad [V_\alpha \in P \rightarrow V_\alpha \leq_T U_\alpha] \quad \& \quad [V_\alpha \notin P \rightarrow {}^2V_\alpha \leq_T U_\alpha],$$

where (V_α, U_α) is the i^{th} pair in \mathcal{C} . If $(V_\alpha, U_\alpha) \in \mathcal{C}$, and $V_\alpha \notin P$, then we *directly code* ${}^2V_\alpha$ into ${}^1U_\alpha$ by putting $\langle x, \alpha \rangle$ into ${}^1U_\alpha$ exactly when x is enumerated in ${}^2V_\alpha$. This action will not be subject to restraint by any other requirement. Hence, ${}^2V_\alpha \leq_T {}^1U_\alpha^{[\alpha]} = U_\alpha^{[\alpha]} \leq_T U_\alpha$. Likewise, if $V_\alpha \in P$, and $(V_\alpha, U_\alpha) \in \mathcal{C}$, we directly code V_α into ${}^1U_\alpha$. We call ${}^2V_\alpha$ (in case $V_\alpha \notin P$) and V_α (in case $V_\alpha \in P$) the *direct coder* for P_α . (Note that the pairs in \mathcal{C} produce finitely many direct coding requirements, and these take precedence over all the incomparability requirements below.)

Incomparability requirements $Y \not\leq X$. For the *incomparability* requirements $Y \not\leq X$, let (Φ_e, X_e, Y_e) be an enumeration of all triples (Φ, X, Y) such that Φ is a partial computable functional, $Q \models Y \not\leq X$, and at least one of X and Y is in $Q - P$. (Here we assume that *both* X and Y are in $Q - P$ so we restrain elements from X and code elements into Y . If one of X and Y is in P then we modify the construction to eliminate one of these parts.) For each $\alpha \in T$, $|\alpha| \geq c$, let $(\Phi_\alpha, X_\alpha, Y_\alpha)$ be (Φ_e, X_e, Y_e) where $|\alpha| = c + e$. By hypothesis (4) we have,

$$(8) \quad (\exists C_\alpha, D_\alpha \in P)[C_\alpha \geq \mathcal{B}(X_\alpha) \ \& \ D_\alpha \leq \mathcal{A}(Y_\alpha) \ \& \ C_\alpha \not\leq D_\alpha].$$

We call D_α an *indirect coder* for Y_α , and we note that $D_\alpha \in \mathcal{B}(\mathcal{A}(Y_\alpha))$. For each $\alpha \in T$, $|\alpha| \geq c$, we assign to α the requirement,

$$(9) \quad R_\alpha : \quad \Phi_\alpha^{X_\alpha} \neq Y_\alpha.$$

We meet R_α by building a functional Ψ_α such that

$$(10) \quad \Phi_\alpha^{X_\alpha} = Y_\alpha \rightarrow \Psi_\alpha^{X_\alpha \oplus Y_\alpha} = D_\alpha.$$

Let φ_α and ψ_α be the use functions of Φ_α and Ψ_α . Each functional has one intended oracle. We omit that oracle, and write $\psi_{\alpha,s}$ in stead of $\psi_{\alpha,s}^{X_{\alpha,s} \oplus Y_{\alpha,s}}$. We describe the α -*strategy* and drop the subscript α . Define the length functions and restraint functions (recalling that $X \upharpoonright x$ denotes the restriction of X to arguments $y \leq x$):

$$\begin{aligned} \ell(s) &= \max\{x \mid (\forall y < x)[\Phi_s^{X_s}(y) = Y_s(y)]\} \\ m(s) &= \max\{x \mid (\exists v < s)[x \leq \ell(v) \ \& \ (\forall y \leq x)[X_s \upharpoonright \varphi_v^{X_v}(y) = X_v \upharpoonright \varphi_v^{X_v}(y)]]\} \\ r(s) &= \max\{0, \max\{\varphi_s^{X_s}(x) \mid x \leq m(s)\}\}. \end{aligned}$$

At stage $s + 1$ restrain with priority α any $x \leq r(s)$ from entering X or entering ${}^2V_\beta$ for any $V_\beta \in Q - P$ such that $(V_\beta, U_\beta) \in \mathcal{C}$, $U_\beta = X$, because ${}^2V_\beta \leq_T {}^1U_\beta$ by direct coding as above. This is the *implied restraint* on ${}^2V_\beta$ by restraint on U_β . Note that there is no restraint on ${}^1V_\beta$. (See the precise implementation of this direct coding and implied restraint in the Construction, Steps 1–2.)

In addition, if $x < \ell(s)$, $\psi_s(x) \uparrow$, and $\varphi_s(x) \downarrow$, then at stage $s + 1$ define $\Psi(x) = D_s(x)$ and define $\psi(x)$ to be the least $z \in \omega^{|\alpha|}$ such that $z > \varphi_s(x)$, s , $\psi_s(y)$, for all $y < x$. If later some element $y \leq \varphi(x)$ enters X causing $\varphi(x)$ to become undefined then we also make $\psi(x)$ undefined using $y \leq \varphi(x) < \psi(x)$. Hence,

$$(11) \quad (\forall x)(\forall s)[\psi_s(x) \downarrow \rightarrow [\varphi_s(x) \downarrow \ \& \ \varphi_s(x) < \psi_s(x)]].$$

If at some later stage t , $x \in D_t$, $x < \ell(t)$, and $\Psi_t(x) = 0$, then at stage $t + 1$ we enumerate $\psi_t(x)$ in Y , and redefine $\Psi_{t+1}(x) = 1$.

If $\Phi^X = Y$ then clearly $\Psi^{X \oplus Y} = D$. Hence, $X \geq_T X \oplus Y \geq_T D$. But $\Phi^X = Y$ implies $\lim_s \ell(s) = \infty$, so $\lim_s r(s) = \infty$. Since $\lim_s r(s) = \infty$ each element $x \notin X$ is eventually permanently restrained from X . The use of $m(s)$ and the quantifier there $(\exists v < s)$ rather than $(\exists v \leq s)$ causes a delay in believing a computation. Using $m(s)$ rather than $\ell(s)$ prevents the restraint from dropping (even if there is a Y change) unless there is an X change below

$r(s)$. (We return to this point when we define $\widehat{\Phi}_\alpha$, and in Lemma 4.5.) Only β -requirements for $\beta < \alpha$ can lift an α -restraint. In Lemma 4.5, we will show that we can tell whether a given restraint is permanent computably in $\mathbb{W}\mathcal{B}(X)$. Hence, $X \leq_T \mathbb{W}\mathcal{B}(X)$. From this and (8) we have, $C \geq_T \mathbb{W}\mathcal{B}(X) \geq_T X$, so $C \geq_T D$, contrary to (8).

(The Sacks Density Theorem, which is a very special case of our Extension Theorem 4.1, asserts that if $D > C$ are c.e. sets then there is a c.e. set Y , $D > Y > C$. To meet the condition $Y \not\leq C$, use the above strategy for requirements R_α with C and coder D .)

The tree T of strategies. For each i , $0 \leq i < c$, we have already put node $\alpha = 0^i$ in T , and have assigned to α the requirement P_α as in (7). If $\alpha \in T$, $|\alpha| \geq c$, put $\alpha \widehat{\ } \langle d, r, k \rangle$ in T where $d, r \in \omega$, $k \in \{0, 1\}$, and associate with α the requirement R_α as in (9). We order these triples $\langle d, r, k \rangle$ lexicographically by first differences. Then we order all the nodes $\alpha \in T$ as in [Soa87, p. 301], with the usual notation such as $\alpha <_L \beta$, and $\alpha \leq \beta$.

Definition 4.2 We define the *true path* $f \in [T]$ as follows by induction on $|\alpha|$. For $n < c$, define $f(n) = 0$. Let $\alpha = f \upharpoonright n$ for some $n \geq c$. Define $f(n) = \langle d_\alpha, r_\alpha, k_\alpha \rangle$ as follows. Define

$$d_\alpha = (\mu x)[\Phi_\alpha^{X_\alpha}(x) \neq Y_\alpha(x)], \text{ and } r_\alpha = \liminf_s r_\alpha(s),$$

and let k_α be defined as below.

We define a computable sequence $\{f_s\}_{s \in \omega}$ such that $\liminf_s f_s = f$. We define s to be an α -stage if $s = 0$ or if $\alpha \subseteq f_s$ and $|\alpha| < s$. Define

$$d_{\alpha,s} = (\mu x)[\widetilde{\Phi}_{\alpha,s}^{X_{\alpha,s}}(x) \neq Y_{\alpha,s}(x)],$$

where $\widetilde{\Phi}_{\alpha,s}^{X_{\alpha,s}}(x) = y$ if $\Phi_{\alpha,s}^{X_{\alpha,s}}(x) \downarrow = y$ with use u and $X_{\alpha,s} \upharpoonright u = X_{\alpha,s'} \upharpoonright u$ where s' is the greatest α -stage $t < s$, and $\widetilde{\Phi}_{\alpha,s}^{X_{\alpha,s}}(x)$ diverges otherwise.

To modify the α -strategy we define the following two functions:

$$g(\alpha, s) = \begin{cases} \psi_{\alpha,s}(d_{\alpha,s}) & \text{if } \psi_{\alpha,s}(d_{\alpha,s}) \downarrow \\ \max\{s \cup \{g(\alpha, t) \mid t < s\}\} & \text{otherwise.} \end{cases}$$

From the definition of $g(\alpha, s)$ here, and the definition of $\psi_s(\alpha, i)$ in Step 6 below, it will follow that $g(\alpha, s) \leq g(\alpha, t)$ for all $s \leq t$. Hence, either $\lim_s g(\alpha, s) = \infty$ or $\lim_s g(\alpha, s) \downarrow < \infty$. Define

$$k_\alpha = \begin{cases} 0 & \text{if } \lim_s g(\alpha, s) = \infty \\ 1 & \text{if } \lim_s g(\alpha, s) \downarrow < \infty; \end{cases}$$

$$G(\alpha, s) = \min\{g(\beta, s) \mid \beta \hat{<} d, r, k \hat{>} \subseteq \alpha \ \& \ k = 0\}.$$

(The point is that if $\alpha \subset f$, $\beta \hat{<} d, r, k \hat{>} \subseteq \alpha$, and $k = 0$, then β may enumerate infinitely many elements of the form $\psi_{\beta, s}(i)$ all of which ignore any restraint $r_{\alpha, s}$ by α . However, to neutralize this effect, α can wait for $g(\beta, s) > z$ for some z which α wants to restrain. In the case $k = 1$, node β will act only finitely often, and α will eventually cease to be injured by β .)

Let s' be the greatest α -stage $< s$. Define the following functions,

$$k_{\alpha, s} = \begin{cases} 0 & \text{if } g(\alpha, s') < g(\alpha, s) \\ 1 & \text{otherwise;} \end{cases}$$

$$\widehat{\Phi}_{\alpha, s}^{X_{\alpha, s}}(x) = \begin{cases} \widetilde{\Phi}_{\alpha, s'}^{X_{\alpha, s}}(x) & \text{if } \varphi_{\alpha, s'}(x) \downarrow = y < G(\alpha, s) \ \& \ X_{\alpha, s} \parallel y = X_{\alpha, s'} \parallel y \\ \uparrow & \text{otherwise.} \end{cases}$$

Define $\widehat{\ell}(\alpha, s)$, $\widehat{m}(\alpha, s)$, and $\widehat{r}(\alpha, s)$ as above but with $\widehat{\Phi}_{\alpha, s}(x)$ in place of $\Phi_{\alpha, s}(x)$. Hence,

$$(12) \quad \widehat{r}(\alpha, s) < G(\alpha, s).$$

Define

$$(13) \quad \mathcal{F}(\alpha) = \{\beta \mid \beta < \alpha \ \& \ |\beta| \geq c \ \& \ [X_\beta = Y_\alpha \vee (Y_\alpha, X_\beta) \in \mathcal{C}]\},$$

The members β of $\mathcal{F}(\alpha)$ are exactly those nodes on T which have the power to restrain elements from ${}^2Y_\alpha$, because $\beta < \alpha$ and either $X_\beta = Y_\alpha$ or there is a direct coding P -requirement to ensure that ${}^2Y_\alpha \leq_T {}^1X_\beta$, so X_β -restraint becomes implied ${}^2Y_\alpha$ -restraint. Define

$$(14) \quad F(\alpha) = \max\{r_\beta \mid \beta \in \mathcal{F}(\alpha)\}, \text{ and } F(\alpha, s) = \max\{\widehat{r}(\beta, s) \mid \beta \in \mathcal{F}(\alpha)\}.$$

The *modified α -strategy* is roughly the same as above except that α can define $\psi_{\alpha, s+1}(i) = z$ only if $\alpha \subseteq f_{s+1}$ and $z > \max\{s, \varphi_{\alpha, s+1}(i), g(\alpha, s)\}$, but if $i \in D_{\alpha, s}$ and $\psi_{\alpha, s}(i) = z > F(\alpha, s)$ then α will enumerate z in $Y_{\alpha, s+1}$ immediately (whether or not $\alpha \subseteq f_{s+1}$) and will *initialize* every $\gamma \in T$ (i.e. cause $\psi_{\gamma, s+1}$ to become undefined) such that $\alpha <_L \gamma$. Such enumeration when $\alpha \not\subseteq f_{s+1}$ is called *residual action* for α . We also initialize γ at stage $s+1$ if $f_{s+1} <_L \gamma$. If $\alpha \subset f$ then $\lim_s G(\alpha, s) = \infty$ and $\liminf_s F(\alpha, s) = F(\alpha) < \infty$, so the modified α -strategy succeeds in satisfying requirement R_α . (See Lemma 4.7.)

The construction. To *initialize* node α at stage s means to let $\psi_{\alpha,s}(i)$ be undefined for all i .

Stage $s = 0$. For all $\alpha \in T$ and $i \in \omega$ let $\psi_{\alpha,0}(i)$ be undefined, and let $Z_0 = \emptyset$ for all $Z \in Q$. Let $P = \{C_1, \dots, C_p\}$. For all $1 \leq j \leq p$, choose e_j such that $W_{e_j} = C_j$.

Stage $s+1$. Perform these steps in order.

Step 1. Representing P . If $x \in W_{e_j,s} - C_{j,s}$, and $x < s$, then enumerate x in C_j and enumerate $\langle x, \beta \rangle$ in U_β (necessarily in ${}^1U_\beta$) for each β such that $(V_\beta, U_\beta) \in \mathcal{C}$ with $V_\beta = C_j$.

Step 2. Enumerating $\psi_\alpha(i)$. Choose the $<$ -least α (if such α exists) such that $c \leq |\alpha| < s$, and:

1. $i \in D_{\alpha,s}$;
 2. $\psi_{\alpha,s}(i) \downarrow < s$ and $\psi_{\alpha,s}(i) \notin Y_{\alpha,s}$;
 3. $\psi_{\alpha,s}(i) > F(\alpha, s)$;
- (a) For each such i enumerate $x = \psi_{\alpha,s}(i)$ in Y_α , and enumerate $\langle x, \gamma \rangle$ in U_γ (necessarily in ${}^1U_\gamma$) for each γ such that $(V_\gamma, U_\gamma) \in \mathcal{C}$ with $V_\gamma = Y_\alpha$.
- (b) Initialize every $\gamma \in T$ such that $\alpha <_L \gamma$.

Now choose the next least α satisfying the above, and perform the indicated action. Iterate until there are no more α .

Step 3. Canceling $\psi_\alpha(i)$. For each α and i if there exists $z \leq \varphi_{\alpha,s}(i)$ such that $z \in X_{\alpha,s+1} - X_{\alpha,s}$ then let $\psi_{\alpha,s+1}(i)$ be undefined.

Step 4. Defining f_{s+1} . Define f_{s+1} , the approximation to the true path at stage $s+1$, as follows. For $n < c$, define $f_{s+1}(n) = 0$. Let $\alpha = f_{s+1} \upharpoonright n$ for some n , $c \leq n < s$. Define $f_{s+1}(n) = \langle d_{\alpha,s+1}, \hat{r}(\alpha, s+1), k_{\alpha,s+1} \rangle$. Hence, $f_{s+1} \in T$ and $|f_{s+1}| = s$. Initialize all $\gamma \in T$, such that $f_{s+1} <_L \gamma$.

Step 5. Defining $\psi_\alpha(i)$. Suppose that Steps 1–4 do not cause $\psi_{\alpha,s+1}(i)$ to become undefined, and

$$\alpha \subseteq f_{s+1} \ \& \ \varphi_{\alpha,s+1}(i) \downarrow \ \& \ \psi_{\alpha,s}(i) \uparrow \ \& \ i < \ell(\alpha, s), \text{ and}$$

$$(\exists z > s)[z \in \omega^{[\alpha]} \ \& \ z > \max\{s, \varphi_{\alpha,s+1}(i), g(\alpha, s), \psi_{\alpha,t}(i) : t \leq s\}].$$

For each such α choose the least such i and z , and define $\psi_{\alpha,s+1}(i) = z$, and $\Psi^{X_{\alpha,s+1} \oplus Y_{\alpha,s+1}}(i) \downarrow = D_{\alpha,s}(i)$.

The verification. By construction, $\alpha \subset f$ if and only if there are only finitely many stages during which α is initialized and there are infinitely many stages s during which $\alpha \subset f_s$.

Lemma 4.3 (i) $(\forall \alpha)(\forall s)(\forall t)[t < s \rightarrow g(\alpha, t) \leq g(\alpha, s)]$.

(ii) $(\forall \alpha \subset f)[\alpha \subset f \rightarrow \lim_s G(\alpha, s) = \infty]$.

(iii) $(\forall \alpha \subset f)(\exists v_\alpha)(\forall s > v_\alpha)(\forall \beta < \alpha)(\forall z < G(\alpha, s))$

[β does not cause z to be enumerated in Y_β at $s + 1$.]

Proof. (i) Suppose $g(\alpha, s+1) \neq g(\alpha, s)$. If $\psi_{\alpha, s+1}(d_\alpha) \uparrow$ then the second clause in the definition of $g(\alpha, s+1)$ ensures that $g(\alpha, s+1) \geq g(\alpha, s)$. If $\psi_{\alpha, s+1}(d_\alpha)$ is newly defined at stage $s+1$ then $\psi_{\alpha, s+1}(d_\alpha) > g(\alpha, s)$ by Step 5, and $g(\alpha, s+1) = \psi_{\alpha, s+1}(d_\alpha)$. In the remaining case where $\psi_{\alpha, s+1}(d_\alpha) = \psi_{\alpha, s}(d_\alpha) \downarrow$ we have $g(\alpha, s+1) = g(\alpha, s)$ by the definition of g .

(ii) Assume that $\alpha \subset f$. If $\beta^\wedge \langle d, r, k \rangle \subseteq \alpha$ and $k = 0$ then $\lim_s g(\beta, s) = \infty$. Hence, $\lim_s G(\alpha, s) = \infty$.

(iii) Fix $\alpha \subset f$. Fix stage u_α such that for all $s \geq u_\alpha$ $f_s \not\subseteq_L \alpha$, and no $\beta < \alpha$ has $\psi_{\beta, s}$ enumerated in Y_α (necessarily under Step 2).

Fix $\beta \subseteq \alpha$. If $\Phi_\beta^{X_\beta}(d_\beta) \uparrow$ then $\varphi_{\beta, s}(d_\beta)$, and hence $\psi_{\beta, s}(d_\beta)$, diverge for infinitely many s by Step 3. Hence, $\lim_s g(\beta, s) = \infty$ by Step 5 and by the second clause in the definition of $g(\beta, s)$.

If $\Phi_\beta^{X_\beta}(d_\beta) \downarrow \neq Y_\beta(d_\beta)$ then choose t_β such that for all $i < d_\beta$, and $s \geq t_\beta$, $\varphi_{\beta, s}(i) = \varphi_\beta(i)$, $Y_\beta \parallel d_\beta = Y_{\beta, s} \parallel d_\beta$, and therefore $\ell(\beta, s) = d_\beta$. Our use of “ $i < \ell(\beta, s)$ ” in defining $\psi_\beta(i)$ in Step 5 means that $\psi_\beta(j)$ is never redefined at any stage $s \geq t_\beta$ for any $j \geq d_\beta$. Hence, β contributes at most finitely many elements to Y_β . Choose v_α large enough to exceed: u_α , the supremum of the stages t_β for all $\beta \subset \alpha$; and the stages at which there are contributions of $\psi_{\beta, s}(i)$ for some $i \leq d_{\beta, s}$ for β such that $\beta^\wedge \langle d, r, k \rangle \subset \alpha$ and $k = 1$.

After stage v_α no $\beta < \alpha$ can contribute an element to Y_β unless $\beta^\wedge \langle d, r, k \rangle \subseteq \alpha$ and $k = 0$. In this case the contribution must be of the form $\psi_{\beta, s}(i)$ for $i \geq d_{\beta, s}$. But $\psi_{\beta, s}(i) \geq \psi_{\beta, s}(d_{\beta, s}) = g(\beta, s) \geq G(\alpha, s)$. ■

Definition 4.4 An α -restraint $\hat{r}(\alpha, s)$ is $\mathcal{B}(X_\alpha)$ -correct at s if $s > v_\alpha$, as defined in Lemma 4.3, and if

$$(\forall B \in \mathcal{B}(X_\alpha)) (\forall z < \hat{r}(\alpha, s)) [z \in B_s \iff z \in B].$$

Lemma 4.5 (Correctness Lemma) Suppose that $\hat{r}(\alpha, s)$ is $\mathcal{B}(X_\alpha)$ -correct at s , $z \leq \hat{r}(\alpha, s)$, and $z \notin X_{\alpha, s}$. Then either $z \leq \hat{r}(\alpha, t)$ for all $t \geq s$, in which case $z \notin X_\alpha$, or at some stage $t > s$ we initialized α (and hence all $\gamma \geq \alpha$).

Proof. By the definitions of $\hat{m}(\alpha, s)$ and $\hat{r}(\alpha, s)$ the restraint $\hat{r}(\alpha, s)$ cannot decrease unless some element $x \leq \hat{r}(\alpha, s)$ enters X_α or else α is initialized. Fix $z, s, \hat{r}(\alpha, s)$, and $X_{\alpha, s}$ as in statement of the Lemma. Suppose that α is not initialized after stage s . Choose the least $t \geq s$ such that some $z \leq \hat{r}(\alpha, s)$ enters X_α at stage $t + 1$. Note that $\hat{r}(\alpha, s) \leq \hat{r}(\alpha, t)$.

Case 1. z enters X_α because of Step 1 at stage $t + 1$ for $C \in P$. Then either $C = X_\alpha$ or $C = V_\beta$, $(V_\beta, U_\beta) \in \mathcal{C}$, and $U_\beta = X_\alpha$. In either case, $C \in \mathcal{B}(X_\alpha)$, contrary to our assumption that $\hat{r}(\alpha, s)$ is $\mathcal{B}(X_\alpha)$ -correct at s .

Case 2. z enters X_α because of Step 2 at stage $t + 1$. Then either: $X_\alpha = Y_\gamma$; or $Y_\gamma = V_\beta$, $(V_\beta, U_\beta) \in \mathcal{C}$, and $U_\beta = X_\alpha$, and Step 2 acted to enumerate $z = \psi(\gamma, t)$ in Y_γ . If $\gamma <_L \alpha$ or $\gamma \hat{\ } \langle d, r, 1 \rangle \subseteq \alpha$ then γ cannot act at any stage $w \geq v_\alpha$. If $\gamma \hat{\ } \langle d, r, 0 \rangle \subseteq \alpha$ then $G(\alpha, t) < \psi(\gamma, t)$ by the definitions of $G(\alpha, t)$ and $\hat{r}(\alpha, t)$, so $z < \hat{r}(\alpha, t) < \psi_{\gamma, t}(i)$ and $z \neq \psi_{\gamma, t}(i)$. If $\alpha < \gamma$, then by the definition of $F(\alpha, s)$ in (13) and (14), and by clause 3 of Step 2 of the construction, the restraint by α prevents γ from enumerating x into X_α at stage $t + 1$. (Note that $x < z = \langle x, \beta \rangle$ and hence restraint on z is restraint on x .) ■

Lemma 4.6 (True Computation Lemma) *Fix $\alpha \subset f$, the true path. Take v_α as in Lemma 4.3. Assume that $\psi_{\alpha, s}(i)$ is defined for some $s > v_\alpha$, $\psi_{\alpha, s}(i) < \hat{r}(\alpha, s)$, and $\hat{r}(\alpha, s)$ is $\mathcal{B}(X_\alpha)$ -correct at s . Then,*

$$(\forall t > s)[\Phi_{\alpha, t}(i) = \Phi_{\alpha, s}(i) \ \& \ \varphi_{\alpha, t}(i) = \varphi_{\alpha, s}(i) \ \& \ \psi_{\alpha, t}(i) = \psi_{\alpha, s}(i)].$$

Proof. No β such that $\beta < \alpha$ or $\beta \hat{\ } \langle d, r, 1 \rangle \subseteq \alpha$ can act at any stage $s \geq w_\alpha$, and α cannot be initialized at such a stage. By the definitions of $G(\alpha, s)$ and $\hat{\Phi}_{\alpha, s}^{X_{\alpha, s}}$ no $\beta \hat{\ } \langle d, r, 0 \rangle \subseteq \alpha$ can contribute an element $z < \hat{r}(\alpha, s)$ to X_α . Hence, by Lemma 4.5 the computations of φ_α and Φ_α remain unchanged at all $t > s$. By Step 3 of the construction, the value of ψ_α also remains unchanged. ■

Lemma 4.7 (Incomparability Lemma) *The incomparability requirements $Y \not\leq X$ in Q are all satisfied.*

Proof. We discuss the most complicated case where both P and Q are in $Q - P$. Suppose towards a contradiction that $Q \models Y \not\leq X$, but $\Phi_e^X = Y$ for some e . Choose some $\alpha \subset f$ such that $(\Phi_\alpha, X_\alpha, Y_\alpha)$ is (Φ_e, X, Y) . Choose an α -stage $s_0 \geq v_\alpha$ of Lemma 4.3 such that for all $s \geq s_0$: α is not initialized at stage s ; for all β and i if $\beta <_L \alpha$, or if $\beta \subset \alpha$ and $i < d_\beta$, then $\psi_{\beta, s}(i) = \psi_\beta(i)$ (which may diverge for some i if $\beta <_L \alpha$), and $\psi_\beta(i)$ is not enumerated at stage s .

Now $\alpha \subset f$ implies that $\liminf_s F(\alpha, s) = F(\alpha)$, and $\lim_s G(\alpha, s) = \infty$ by Lemma 4.3. By assumption $\Phi_\alpha^{X_\alpha} = Y_\alpha$, and hence $\lim_s \ell(\alpha, s) = \infty$. We claim that $\Psi_\alpha^{X_\alpha \oplus Y_\alpha} = D_\alpha$. For each i choose $t_i \geq s_0$ such that for all $t \geq t_i$, $\varphi_{\alpha, t}(i) = \varphi_\alpha(i)$ and no $x \leq \varphi_\alpha(i)$ is enumerated in $X_\alpha \cup Y_\alpha$ at stage t . Thus, at some α -stage $u_i \geq t_i$ there exists a z satisfying Step 5 and we define $\psi_{\alpha, u_i}(i) = z$. Hence, $\psi_{\alpha, s}(i) = \psi_\alpha(i)$ for all $s \geq u_i$ because after stage u_i , α is never initialized and Step 3 does not apply to $\psi_\alpha(i)$. If $i \in D_{\alpha, v}$, for $v \geq u_i$, then choose the least $s \geq v$ such that $F(\alpha, s) < z$. At stage $s+1$ if $\psi_{\alpha, s}(i) \notin Y_{\alpha, s}$ we enumerate $\psi_{\alpha, s}(i)$ in $Y_{\alpha, s+1}$, and redefine $\Psi_{\alpha, s+1}(i) = D_{\alpha, s+1}(i)$ by Step 2. Hence, $\Psi^{X_\alpha \oplus Y_\alpha} =^* D_\alpha$, so $X_\alpha \geq_T (X_\alpha \oplus Y_\alpha) \geq_T D_\alpha$.

Furthermore, $\lim_s \ell(\alpha, s) = \infty$ implies $\lim_s \hat{r}(\alpha, s) = \infty$. Fix any z . Choose $s > z$ such that $z \leq \hat{r}(\alpha, s)$ and $\hat{r}(\alpha, s)$ is $\mathcal{B}(X_\alpha)$ -correct. Now by Lemma 4.6 if $z \notin X_{\alpha, s}$ then $z \notin X_\alpha$. Hence, $\mathbb{W} \mathcal{B}(X_\alpha) \geq_T X_\alpha$. From this and the first clause of (8) we have, $C_\alpha \geq_T \mathbb{W} \mathcal{B}(X_\alpha) \geq_T X_\alpha$, and therefore $C_\alpha \geq D_\alpha$, contrary to the third clause of (8). \blacksquare

Lemma 4.8 (Recovery Lemma) *For every $Z \in Q$, $Z \leq_T \mathbb{W} \mathcal{B}(\mathcal{A}(Z))$.*

Proof. If $Z \in P$ then clearly $\mathbb{W} \mathcal{B}(\mathcal{A}(Z)) \equiv_T Z$. Now assume $Z \in Q - P$. We will prove that $Z^{[\beta]} \leq_T \mathbb{W} \mathcal{B}(\mathcal{A}(Z))$ uniformly in β . Fix $\beta \in T$.

Case 1. $Z^{[\beta]} \subseteq {}^1 Z$, $|\beta| < c$, $Z = U_\beta$ where $(V_\beta, U_\beta) \in \mathcal{C}$, and $V_\beta = C \in \mathcal{P}$. Let $z = \langle x, \beta \rangle$. Then by Step 1, $x \in C$ iff $z \in Z$. Now $C \in \mathcal{B}(Z)$ by definition of $\mathcal{B}(Z)$. Hence, $Z^{[\beta]} \leq_T \mathbb{W} \mathcal{B}(\mathcal{A}(Z))$.

Case 2. $Z^{[\beta]} \subseteq {}^1 Z$ and there is some $(V_\beta, U_\beta) \in \mathcal{C}$ with $U_\beta = Z$, and $V_\beta = Y_\alpha$ for some Sacks incomparability requirement R_α . We may assume by induction the rank of $<$ in Q (measured by the length of chains) that $V_\beta \leq_T \mathbb{W} \mathcal{B}(\mathcal{A}(V_\beta))$. Now by (5) we know that $\mathcal{B}(\mathcal{A}(V_\beta)) \subseteq \mathcal{B}(U_\beta)$ and $x \in V_\beta$ iff $\langle x, \beta \rangle \in Z$. Hence, $Z^{[\beta]} \leq_T \mathbb{W} \mathcal{B}(\mathcal{A}(Z))$.

Case 3. $Z^{[\beta]} \subseteq {}^2 Z$. Then $|\alpha| \geq c$, and $Z = Y_\alpha$, for some Sacks incomparability requirement R_α for $\alpha = \beta$. Fix z . Choose $u > z$. Suppose $\psi_{\alpha, u}(i) = z$. (If not then $z \neq \psi_{\gamma, s}(j)$ for all $s > u$ and all γ and j , because of the restriction $z > s$ in Step 5. In this case $z \in Z$ iff $z \in Z_u$.) Let D_α be the indirect coder for Y_α used by the α -strategy, and recall that $D_\alpha \in \mathcal{B}(\mathcal{A}(Y_\alpha))$ by (8). Hence, effectively in $\mathcal{B}(\mathcal{A}(Z))$ we compute whether $i \in D_\alpha$, and if so, compute the least $v \geq u$ such that $i \in D_{\alpha, v}$. Suppose $z \notin Z_v$.

It remains to determine computably in $\mathbb{W} \mathcal{B}(\mathcal{A}(Z))$ the least $t \geq v$ such that either: z is permanently restrained for all $s \geq t$ by some $\beta < \alpha$; or z is

canceled as a position for $\psi_\alpha(i)$; or z enters Z . For each s define

$$\mathcal{S}_s = \{\beta \mid \beta \in \mathcal{F}(\alpha) \ \& \ z \leq \hat{r}(\beta, s)\}.$$

These are exactly the nodes β which at stage s are restraining z from entering Z according to Step 2 clause (iii). From the definition of $\mathcal{F}(\alpha)$ we know that for each such β , either $Y_\alpha = X_\beta$ or $(Y_\alpha, X_\beta) \in \mathcal{C}$. In the latter case by the definition (6) of \mathcal{C} we know $Y_\alpha \notin \mathcal{Z}(X_\beta)$. Therefore, by the definition (5) of $\mathcal{Z}(X)$, we have

$$(15) \quad \mathcal{B}(X_\beta) \subseteq \mathcal{B}(\mathcal{A}(Y_\alpha)) = \mathcal{B}(\mathcal{A}(Z)).$$

Thus, in either case $\mathbb{W} \mathcal{B}(X_\beta) \leq_T \mathbb{W} \mathcal{B}(\mathcal{A}(Z))$. Hence, by Lemma 4.6 computably in $\mathbb{W} \mathcal{B}(\mathcal{A}(Z))$ we can search for the first stage $t \geq v$ such that: (i) there is a $\beta \in \mathcal{S}_t$ such that the β -restraint is permanent thereafter, i.e. $\hat{r}(\beta, s) \geq z$ for all $s \geq t$; or (ii) $\psi_{\alpha,t}(i) \neq z$; or (iii) for all $\beta \in \mathcal{S}_t$, $\hat{r}(\beta, t) < z$. If (iii) occurs, then z enters Z at stage $t + 1$ by Step 2. Otherwise, z will never enter Z .

We now verify that the above search will terminate. If $z \leq F(\alpha)$, then $z \leq r_\beta$ for some $\beta \in \mathcal{F}(\alpha)$, so (i) holds for β at almost every stage s . If $z > F(\alpha)$, then let γ be the $>$ -greatest (shortest) node on f which is incompatible with α . Let t_0 be so large that for every $t \geq t_0$ and every $\beta \in T$, if $\beta <_L \gamma$ then $\beta <_L f_t$. Since $z > F(\alpha)$, there must be a stage t_1 greater than or equal to t_0 such that for each $\beta \in \mathcal{F}(\alpha)$, if $\beta <_L \gamma$ then $\hat{r}(\beta, t)$ is less than z , by virtue of a change in X_β since the most recent β stage prior to t_1 . Since $t_1 \geq t_0$, for such β there will be no β stages after stage t_1 , and so if $t \geq t_1$ then $\hat{r}(\beta, t) < z$. For each of the other elements β of $\mathcal{F}(\alpha)$, z is greater than or equal to r_β (by the enumeration of Ψ_α in Step 2), and for each γ stage t , $\hat{r}(\beta, t) = r_\beta$. Consequently, $F(\alpha, t) < z$ during the first γ stage t greater than t_1 . ■

Lemma 4.9 (Comparability Lemma) *All the comparability requirements $V \leq U$ are satisfied.*

Proof. Suppose $V \leq U$. We must show that $V \leq_T U$. The proof is by induction on the rank of V , where the *rank* of V is defined as the length of the longest chain of elements of Q below V .

Case 1. $V \in \mathcal{Z}(U)$. In this case by (5) we have, $\mathcal{B}(\mathcal{A}(\mathcal{Z}(U))) \subseteq \mathcal{B}(U)$, and thus in particular, $\mathcal{B}(\mathcal{A}(V)) \subseteq \mathcal{B}(U)$. But by Lemma 4.8, we have $V \leq_T \mathbb{W} \mathcal{B}(\mathcal{A}(V))$, and clearly $\mathbb{W} \mathcal{B}(U) \leq_T U$, so $V \leq_T U$.

Case 2. $V \notin \mathcal{Z}(U)$. If $V \in P$ then V is coded into 1U under the direct coding in Step 1. Assume $V \notin P$. Then V is decomposed into the disjoint union of

sets 1V and 2V . Now ${}^1V \equiv_T \oplus\{{}^2W_j : 1 \leq j \leq j_0\}$ where $W_j < V$. Each W_j has lower rank than V and hence $W_j \leq_T U$ by the inductive hypothesis. Therefore, ${}^1V \leq_T U$. The remaining part of V is 2V which is coded into 1U (and hence into U) under Step 2 because $V \notin \mathcal{Z}(U)$. ■

This completes the proof of the Lemma and of Theorem 4.1. We turn to some applications and examples of the Extension Theorem 4.1. ■

Some further examples. The left hand example of Figure 3 illustrates some of the features of our Extension Theorem 4.1.

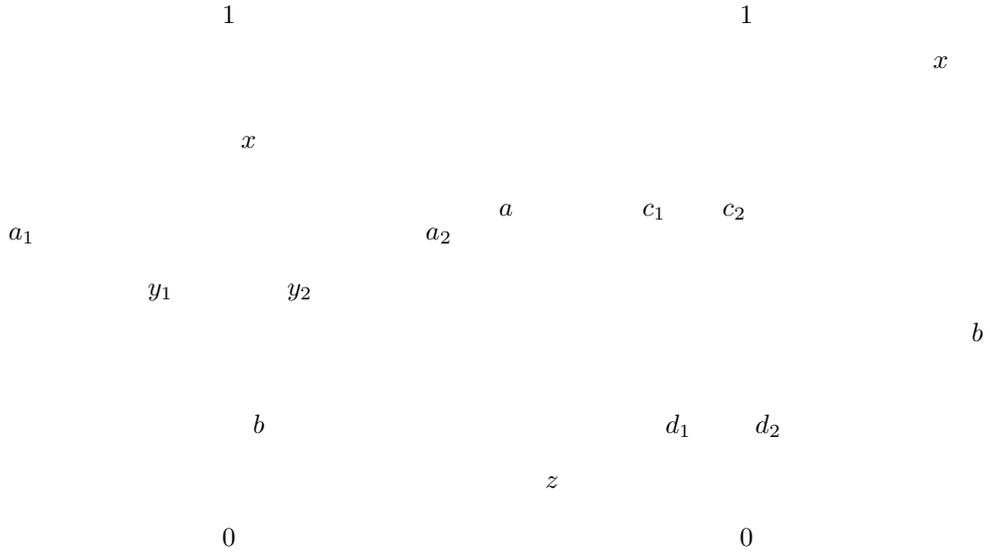


Figure 3: Two extensions guaranteed by the Extension Theorem

This example requires our modified coding method because to meet the requirement $Y_1 \not\leq_T A_2$ we use A_1 as the indirect coder and ensure $\Phi^{A_2} = Y_1$ implies $\Psi^{Y_1 \oplus A_2} = A_1$. A further apparent complication in this example is that since we directly code Y_1 into X , any X restraint causes implicit Y_1 restraint. However, the Sacks coding of A_2 into Y_2 (to achieve $Y_2 \not\leq_T B$), may cause the X restraint to drop, allowing elements to enter Y_1 . Since $A_2 \not\leq_T A_1$ this appears to jeopardize the recovery procedure to prove $Y_1 \leq_T A_1$. Nevertheless, by putting the construction on a tree we have allowed each requirement R_α , $\alpha \subset f$, to correctly guess at the computable set contributed by each $\beta < \alpha$ so this is not an obstacle. It is easy to check that conditions (4) and (5) are both satisfied in this example.

In the previous example we have $\mathcal{Z}(w) = \emptyset$ for every $w \in Q - P$. To produce an extendible example where $\mathcal{Z}(w) \neq \emptyset$ for some w consider the second example in Figure 3. Note that $\mathcal{B}(x) = \{d_1, d_2, b, 0\}$, $\mathcal{A}(z) = \{a, c_1, c_2, 1\}$, and $\mathcal{B}(\mathcal{A}(z)) = \{d_1, d_2, 0\}$ so $\mathcal{Z}(x) = \{z\}$. But $\mathcal{B}(\mathcal{A}(z)) \subseteq \mathcal{B}(x)$ so (5) is still satisfied and the Extension Theorem 4.1 applies. An even simpler example where $\mathcal{Z}(w) \neq \emptyset$ is given by the following description of Q , $Q = \{0 < z < c < b < x < a < 1\}$. Now $\mathcal{B}(X) = \{0, c, b\} \not\subseteq \{c\} = \mathcal{B}(\mathcal{A}(z))$ so $\mathcal{Z}(x) = \{z\}$. However, $\mathcal{B}(\mathcal{A}(z)) \subseteq \mathcal{B}(x)$ so (5) is still satisfied and the Extension Theorem 4.1 applies.

5 The second nonextension theorem

Theorem 5.1 (Second Nonextension) *Suppose that P and Q are finite partially ordered sets with 0 and 1 and that P is a sub-partial-order of Q . If there is an x in $Q - P$ such that $\mathcal{Z}(x)$ is not empty and $\mathcal{B}(\mathcal{A}(\mathcal{Z}(x) \cup \mathcal{B}(x)))$ is not contained in $\mathcal{B}(x)$ then there is an embedding of P into \mathcal{R} which cannot be extended to an embedding of Q into \mathcal{R} .*

Under the hypotheses of Theorem 5.1, there is an x in $Q - P$ and an element h of P such that h is not below x but h is below every element of P which is an upper bound of $\mathcal{Z}(x) \cup \mathcal{B}(x)$. We will embed P into \mathcal{R} so that for every sequence of candidates \mathbf{Z} which could be assigned to the elements of $\mathcal{Z}(x)$, either \mathbf{Z} does not satisfy the comparability and incomparability requirements associated with $\mathcal{Z}(x)$ or the join of \mathbf{Z} with the image of $\mathcal{B}(x)$ is above the image of h . Consider the example of Figure 4, in which P equals $\{0, a, b, 1\}$ and $Q - P$ equals $\{z, x\}$.

Here, $\mathcal{A}(z)$ is $\{a, 1\}$; $\mathcal{B}(\mathcal{A}(z))$ is $\{a, 0\}$; $\mathcal{B}(x)$ is $\{b, 0\}$; thus, $\mathcal{B}(x)$ is not contained in $\mathcal{B}(\mathcal{A}(z))$. Since $z < x$, $z \in \mathcal{Z}(x)$. Of course, z is the only possible element of $\mathcal{Z}(x)$ since it is the only element of $Q - P$ which is strictly below x . Further, $\mathcal{A}(\mathcal{Z}(x) \cup \mathcal{B}(x))$ is $\mathcal{A}(\{z\} \cup \{0, b\})$ and the only element of P above all the elements of $\{0, z, b\}$ is 1. Thus, $\mathcal{A}(\mathcal{Z}(x) \cup \mathcal{B}(x))$ is $\{1\}$. Then, $\mathcal{B}(\mathcal{A}(\mathcal{Z}(x) \cup \mathcal{B}(x)))$ equals $\mathcal{B}(1)$, which is $\{0, a, b, 1\}$. So, $\mathcal{B}(\mathcal{A}(\mathcal{Z}(x) \cup \mathcal{B}(x))) \not\subseteq \mathcal{B}(x)$ since $\{1, a, b, 0\} \not\subseteq \{b, 0\}$. Thus, this example satisfies the hypotheses of Theorem 5.1.

Definition 5.2 (1) Let q be the number of elements in $\mathcal{Z}(x)$. Let z_1, z_2, \dots, z_q enumerate the elements of $\mathcal{Z}(x)$. (2) for each i less than or equal to q , let q_i be the number of elements of $\mathcal{A}(z_i)$ and let $a_{(i,1)}, \dots, a_{(i,q_i)}$ enumerate $\mathcal{A}(z_i)$. Let \mathbf{A}_i equal $A_{(i,1)}, A_{(i,2)}, \dots, A_{(i,q_i)}$ and be an enumeration of the image of $\mathcal{A}(z_i)$. Of course, \mathbf{A}_i is to be constructed. (3) Let b_1, \dots, b_k enumerate the

1

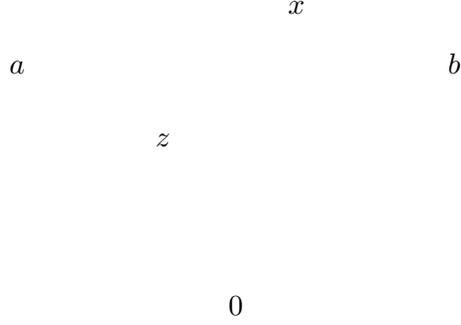


Figure 4: An extension ruled out by Theorem 5.1

nonzero elements of $\mathcal{B}(x)$; since $\mathcal{Z}(x)$ is not empty neither is b_1, \dots, b_k . Let $\mathbf{B} = B_1, \dots, B_k$ be an enumeration of the image of $\mathcal{B}(x)$, to be constructed. (4) Fix h so that h is in $\mathcal{B}(\mathcal{A}(\mathcal{Z}(x) \cup \mathcal{B}(x))) - \mathcal{B}(x)$. Let H denote the image of h , to be constructed.

In addition to satisfying the isomorphism requirements for the embedding of P in \mathcal{R} , we will ensure that for every bijection $z_i \mapsto Z_i$ between $\mathcal{Z}(x)$ and a sequence of computably enumerable sets \mathbf{Z} one of following three conditions holds:

1. There is a z_i in $\mathcal{Z}(x)$ and an $a_{(i,j)}$ in $\mathcal{A}(z_i)$ such that $A_{(i,j)} \not\geq_T Z_i$. Thus, $P \cup \mathcal{Z}(x) \rightarrow P_{\mathcal{R}} \cup \mathbf{Z}$ fails to preserve an instance of comparability required by Q .
2. H is computable in $\mathbf{Z} \oplus \mathbf{B}$. Thus, for any X , if $X \geq_T \mathbf{Z} \oplus \mathbf{B}$ then $X \geq_T H$. Consequently, $P \cup \mathcal{Z}(x) \rightarrow P_{\mathcal{R}} \cup \mathbf{Z}$ cannot be extended to embed $P \cup \mathcal{Z}(x) \cup \{x\}$ ordered as in Q isomorphically into \mathcal{R} .
3. There is a c in P with constructed image C and a z_i in $\mathcal{Z}(x)$ such that $c \notin \mathcal{A}(z_i)$ and $C \geq_T Z_i$. Thus, $P \cup \mathcal{Z}(x) \rightarrow P_{\mathcal{R}} \cup \mathbf{Z}$ fails to preserve an instance of incomparability required by Q .

These conditions ensure that our constructed embedding of P into \mathcal{R} cannot be extended to an embedding of Q into \mathcal{R} . We can translate these conditions

into requirements on the sets to be constructed as follows. For each sequence of c.e. sets \mathbf{Z} of length q and each family of Turing functionals $\Psi = \{\Psi_{(i,j)} \mid i \leq q \text{ and } j \leq q_i\}$, one of the following conditions must hold.

- T1. There are i and j such that $i \leq q$, $j \leq q_i$ and $\Psi_{(i,j)}(A_{(i,j)})$ is not equal to Z_i .
- T2. There is a Turing functional Γ such that $\Gamma(\mathbf{Z} \oplus \mathbf{B})$ equals H .
- T3. There is an i less than or equal to q , a set C in $P_{\mathcal{R}}$ which is not in \mathbf{A}_i and a Turing functional Δ such that $\Delta(C) = Z_i$.

Notation. The requirements that the images of 1 and 0 be complete and computable are denoted by C and R , respectively. For each instance of comparability $c > d$ in P , the requirement that D be computable in C is denoted by $D(C, D)$. For each \mathbf{Z} and Ψ as above, the associated trichotomy requirement is denoted $T(\mathbf{Z}, \Psi)$. We divide the incomparability requirements in P into three types, each to be satisfied differently. If $c \not\geq d$ in P , then for each Turing functional Θ we have the requirement that $\Theta(C)$ is not equal to D . We will denote this requirement by $I1(C, D, \Theta)$, if c is not in $\mathcal{A}(\mathcal{B}(x))$; by $I2(C, D, \Theta)$, if c is in $\mathcal{A}(\mathcal{B}(x) \cup \mathcal{Z}(x))$; and by $I3(C, D, \Theta)$, if c is in $\mathcal{A}(\mathcal{B}(x))$ but not in $\mathcal{A}(\mathcal{Z}(x))$.

5.1 Strategies

We enumerate the elements of $P_{\mathcal{R}}$ in the context of a Π_3 -priority construction. We have nontrivial action on the Π_3 -level by the strategies associated with the \mathbf{Z} -trichotomy requirements: we build functionals Γ and Δ playing the roles described above so that whether Γ is total relative to $\mathbf{Z} \oplus \mathbf{B}$ or Δ is total relative to C is a general Π_3 -statement. In the context of our construction, $\Gamma(\mathbf{Z} \oplus \mathbf{B})$ or $\Delta(C)$ will be total unless some Π_2 -strategy chooses a w and changes the computation of $\Gamma(w, \mathbf{Z} \oplus \mathbf{B})$ or $\Delta(w, C)$ infinitely often.

The incomparability requirements have Π_2 -strategies. A particular Π_2 -strategy will either cause a higher priority Π_3 -strategy to have a Σ_3 -outcome or will ensure that its associated requirement is satisfied. For each requirement, there will only be finitely many requirements of higher priority; so the latter case will apply for some strategy for each requirement.

The tree method. We define T , the tree of strategies, and label the nodes of T by strategies and the edges by outcomes of the strategies associated with the nodes from which they emerge. T will have a greatest element called

the root of T . We uniquely refer to a node α in T by specifying the sequence of pairs of strategies and outcomes which lead from the root to α . We order T by ordering the outcomes of each strategy and then defining $\alpha <_L \beta$ if at the point n at which α and β first disagree the outcome appearing in α is less than that appearing in β .

A Π_2 -strategy σ will work within the environment created by the application of finitely many earlier strategies. We will be given a sequence α of strategies and predicted outcomes for those strategies. So, α will be a sequence $\langle \langle \sigma_i, o_i \rangle \mid i < n \rangle$ in which σ_i is a strategy and o_i is one of its outcomes.

We use $\{\alpha\}$ as a suffix to denote that the preceding expression is interpreted relative to the sequence α . For example, $\sigma\{\alpha\}$ is the strategy assigned to α in the tree of strategies. The strategy $\sigma\{\alpha\}$ might be associated with the requirement $T(\mathbf{Z}, \Psi)\{\alpha\}$. In which case, $\sigma\{\alpha\}$ might enumerate a functional $\Gamma\{\alpha\}$ and ensure that if there are infinitely many expansionary stages for the system of equations $\Psi_{(i,j)}\{\alpha\}(A_{(i,j)}) = Z_i\{\alpha\}$ then there will be infinitely many expansionary stages for the equation $\Gamma\{\alpha\}(\mathbf{Z}\{\alpha\} \oplus \mathbf{B}) = H$. When it does not become too unwieldy, we will economize, such as writing the last expression as $(\Gamma(\mathbf{Z} \oplus \mathbf{B}) = H)\{\alpha\}$. The strategy $\sigma\{\beta\}$ is *above* $\sigma\{\alpha\}$ if β is a proper initial segment of α . In describing strategy σ we will omit appending $\{\alpha\}$ to the objects under construction by σ .

Specifying strategies. Our construction is organized by a global recursion on stages. In describing the recursion step, we speak of the current stage and denote it by s . We specify a strategy σ by describing its step-by-step action in the context of the global recursion. We begin the activity of σ in Step 1. Then, σ follows the instructions in its description until it reaches an instruction which ends its activity for that stage or it has no more instructions to follow. During subsequent stages in which σ acts, σ resumes with the step in which it most recently ended.

Notation for the action during stage s . If W is a given c.e. set then $W[s]$ is the finite set of numbers less than s which can be enumerated into W by computations of length less than s . In the full construction, we will not only have stages s but also substages t and we will use the notation $A[s, t]$ to denote the set A at the beginning of substage t of stage s . Let $v[s, t]$ denote the first t many strategies and approximations to their outcomes which appear during stage s . An element α of T is *active* during stage s if there is a t such that $v[s, t]$ is equal to α and we execute $\sigma\{\alpha\}$. We let $o\{\alpha\}[s, t]$ denote the approximation to the outcome of $\sigma\{\alpha\}$ during substage t of stage s . In this

case, $\sigma\{\alpha\}$ will already have acted during an earlier substage. As above, we write $(\Gamma(\mathbf{Z} \oplus \mathbf{B}) \Vdash x = H \Vdash x)[s]$ to denote that the entire expression is to be evaluated during stage s . Generally, we omit the substage notation and discuss the stage s behavior of σ .

The programming environment. There are be finitely many $T(\mathbf{Z}, \Psi)$ strategies which directly effect σ . We say that the functionals enumerated by these strategies are *active*. They are the functionals which are enumerated by strategies above σ and which no other strategy above σ has caused to be partial. There are two types of $T(\mathbf{Z}, \Psi)$ strategies. A strategy $t1(\mathbf{Z}, \Psi)\{\alpha\}$ enumerates a functional $\Gamma\{\alpha\}$ and a strategy $t2(\mathbf{Z}, \Psi)\{\alpha\}$ enumerates a functional $\Delta\{\alpha\}$ with roles as described in the three conditions T1 to T3. Each strategy $\sigma\{\alpha\}$ is assigned an infinite computable set $R\{\alpha\}$; $\sigma\{\alpha\}$ chooses only numbers from $R\{\alpha\}$. For distinct α and β , $R\{\alpha\}$ and $R\{\beta\}$ are disjoint

During each stage s during which $\sigma\{\alpha\}$ is active, $\sigma\{\alpha\}$ has an interval $(r_0, r)\{\alpha\}[s]$ and the only numbers which $\sigma\{\alpha\}$ may enumerate into any set or functional are those which lie in this interval. This σ imposes restraint on strategies of lower priority enumerating numbers into nonzero elements of $P_{\mathcal{R}}$ by setting their value of r_0 greater than the desired restraint.

Notation associated with the trichotomy strategies. We approach the requirement $T(\mathbf{Z}, \Psi)$ with a strategy $t1(\mathbf{Z}, \Psi)\{\alpha_1\}$, which enumerates $\Gamma\{\alpha_1\}$ and ensures that if $\Gamma(\mathbf{Z} \oplus \mathbf{B})\{\alpha_1\}$ is total then its value is H . If making $\Gamma(\mathbf{Z} \oplus \mathbf{B})\{\alpha_1\}$ total interferes with the action of an incomparability strategy $i3(C, B, \Theta)\{\alpha_2\}$ attempting to establish $(\Theta(n, C) \neq D(n))\{\alpha_2\}$ then $i3(C, B, \Theta)\{\alpha_2\}$ has outcome $R|\alpha_1|$. In this case, $i3(C, B, \Theta)\{\alpha_2\}$ makes $\Gamma(\mathbf{Z} \oplus \mathbf{B})\{\alpha_1\}$ partial. We then use a strategy $t2(\mathbf{Z}, \Psi)\{\alpha_3\}$ to enumerate $\Delta\{\alpha_3\}$ and ensure that if $\Delta(C)\{\alpha_3\}$ is total then it equals $Z_{i^*}\{\alpha_3\}$, where $C\{\alpha_3\}$ equals $C\{\alpha_2\}$, $\mathbf{Z}\{\alpha_3\}$ equals $\mathbf{Z}\{\alpha_1\}$ and $i^*\{\alpha_3\}$ is the least index i such that $c\{\alpha_3\}$ is not above z_i . Finally, if $\Delta(C)\{\alpha_3\}$ is not total then we conclude that there is a pair (i, j) such that $\Psi_{(i,j)}(A_{(i,j)})\{\alpha_1\} \neq Z_i\{\alpha_1\}$.

Definition 5.3 If $\sigma\{\alpha\}$ is $t2(\mathbf{Z}, \Psi)\{\alpha\}$, then define: (1) $k\{\alpha\}$ to be the greatest k such that $\sigma\{\alpha \Vdash k\}$ equals $t1(\mathbf{Z}, \Psi)\{\alpha \Vdash k\}$ and both strategies are working on the requirement $T(\mathbf{Z}, \Psi)\{\alpha \Vdash k\}$; and (2) $l\{\alpha\}$ to be the greatest l such that $\sigma\{\alpha \Vdash l\}$ equals $i3(C, D, \Theta)\{\alpha \Vdash l\}$ and $\sigma\{\alpha\}$ equals $Rk\{\alpha\}$.

Opening and closing stages. The stages during which σ is active are divided into two types: *opening stages* and *closing stages*, suggesting the behavior of the $i3(C, D, \Theta)$ strategies. In turn, σ must provide infinitely many

stages of each type. Each opening stage provided by σ , s must have been an opening stage for σ and, excluding the completeness strategy, σ must not have enumerated any number into any set in $P_{\mathcal{R}}$. Each closing stage provided by σ , must have been an closing stage for σ . We define an ordering $<_L$ of the outcomes of our strategies, to be read as *to the left of*.

Operating hypothesis. The strategy $\sigma\{\alpha\}$ is based on the following assumptions, called the *operating hypothesis*.

Interval of activity assumptions. (a) $r_0\{\alpha\}[s]$ is fixed and $r\{\alpha\}[s]$ goes to monotonically to infinity. (b) The only strategies of higher priority which can enumerate numbers from $(r_0, r)\{\alpha\}[s]$ into any set during or after stage s are the direct coding strategies, the completeness strategy and the strategies enumerating active functionals. (c) The enumeration of elements less than r_0 into the sets named in σ or any strategy above σ is completed before the first stage in which σ acts.

True path assumptions. (a) For every i less than $|\alpha|$ and every time $[s, t]$ in which $\sigma\{\alpha \parallel i\}$ acts, the approximated outcome $o\{\alpha \parallel i\}[s, t + 1]$ of $\sigma\{\alpha \parallel i\}$ during stage s is \geq_L the outcome $o\{\alpha \parallel i\}$ which appears in α . (b) Each strategy $\sigma\{\alpha \parallel i\}$ appearing in α acts infinitely often with approximated outcome $o\{\alpha \parallel i\}$.

Δ -assumptions. (a) There are infinitely many opening and closing stages for α . (b) Suppose that $v[s, t]$ equals α , $[s, t]$ is an opening stage for α , $\Delta\{\alpha \parallel j\}$ is active and $\delta(x, C)\{\alpha \parallel j\}$ is defined and less than $r\{\alpha\}[s]$. Suppose that $[s_1, t_1]$ is the first stage and substage after $[s, t]$ such that $v[s_1, t_1]$ equals $\alpha \parallel k\{\alpha \parallel j\}$ and $[s_1, t_1]$ is $t_1\{\alpha \parallel k\{\alpha \parallel j\}\}$ -expansionary. Then, if $(C \parallel \theta(n, C)\{\alpha \parallel l\{\alpha \parallel j\}\})[s, t] = C\{\alpha \parallel j\}[s_1, t_1] \parallel (\theta(n, C)\{\alpha \parallel l\{\alpha \parallel j\}\})[s, t]$ then $(\mathbf{Z}\{\alpha \parallel j\} \parallel r\{\alpha\})[s, t]$ equals $\mathbf{Z}[s_1, t_1]\{\alpha \parallel j\} \parallel r\{\alpha\}[s, t]$.

The opening stages mark the beginnings of intervals during which each $\mathbf{Z}\{\alpha \parallel j\}$ associated with an active functional $\Delta\{\alpha \parallel j\}$ will only change if permitted to do so by $C\{\alpha \parallel j\}$. During such intervals, we have an implicit ability to restrain numbers from entering $\mathbf{Z}\{\alpha \parallel j\}$ by restraining $C\{\alpha \parallel j\}$.

Presentation of strategies. We now present the families of strategies which will appear in our construction. We divide our discussion of each into these parts: the strategy itself; an analysis and ordering $<_L$ of the strategy's possible outcomes, under the assumption that it is executed under the operating

hypothesis; an approximation method, which looks at the stage s behavior of σ and either terminates stage s or produces an approximation to the outcome of σ and an interval $(r_0, r)[s]$ to be used as the interval of activity for the next strategy in the construction; and an analysis of the strategy's effectiveness toward satisfying the requirements of the construction.

In discussing the direct coding, completeness and computability strategies, we act as if their interval of activity is $[0, \infty)$. These are the first strategies in our construction and we can safely implement them in this way. During stage s , the remaining strategies work within subintervals of $[0, s]$.

$d(C, D)$: Comparability strategies. Suppose that c and d are distinct elements of P with $c \geq d$ and with images C and D . We let $d(C, D)$ denote the strategy to code D into C . As in Section 4, we ensure $C \geq_T D$ by directly coding D into C . Let R be the infinite computable set assigned to $d(C, D)$. We ensure that for each n , $n \in D$ if and only if the n th element of R is in C . Whenever we enumerate a number n into D , we enumerate its image, the n th element of R , into C . We fix a procedure CODE: if a strategy σ enumerates a finite set of numbers into elements of $P_{\mathcal{R}}$ and then calls CODE, CODE will enumerate all of the finitely many numbers into elements of $P_{\mathcal{R}}$ required by the direct coding strategies.

c and r : Completeness and computability strategies. We ensure that 0 and 1 map to a computable set and a complete set, respectively. We ensure the former by mapping 0 to \emptyset . We ensure the latter by directly coding a complete Σ_1^0 set into the image of 1. Although we do not construct the set being coded, the coding strategy works just as in the previous section. We let c and r denote the completeness and computability strategies, respectively.

$t1(\mathbf{Z}, \Psi)$ and $t2(\mathbf{Z}, \Psi)$: \mathbf{Z} -trichotomy strategies. We let $t1(\mathbf{Z}, \Psi)$ and $t2(\mathbf{Z}, \Psi)$ denote Γ and Δ -strategies, respectively.

$t1(\mathbf{Z}, \Psi)$: Enumerating Γ . Let \mathbf{Z} be a sequence of c.e. sets $\langle Z_1, \dots, Z_q \rangle$ and let Ψ be associated family of Turing functionals $\Psi_{(i,j)}$. Suppose that $\sigma\{\alpha\}$ is a $t1(\mathbf{Z}, \Psi)$ strategy to enumerate Γ . Let R be the infinite computable set designated for the sole use of $t1(\mathbf{Z}, \Psi)$.

Notation. Let $\ell[s]$ be the greatest l such that for each x less than or equal to l , i less than or equal to q and j less than or equal to q_i , $\Psi_{(i,j)}(x, A_{(i,j)})[s]$ equals $Z_i(x)[s]$. Stage s is $t1(\mathbf{Z}, \Psi)$ -*expansionary over t* if $\ell[s]$ is larger than

it was during stage t . Stage s is $t1(\mathbf{Z}, \Psi)$ -*expansionary* if it is $t1(\mathbf{Z}, \Psi)$ -expansionary over every earlier stage when $t1(\mathbf{Z}, \Psi)$ was active.

During stage s , $t1(\mathbf{Z}, \Psi)$ acts as follows.

Program 5.4 1. (a) If s is $t1(\mathbf{Z}, \Psi)$ -*expansionary* over last stage (if any) in which $t1(\mathbf{Z}, \Psi)$ was in 3 and s is a closing stage for $t1(\mathbf{Z}, \Psi)$ then go to 2.

(b) Otherwise, end $t1(\mathbf{Z}, \Psi)$'s activity for stage s .

2. (a) If there is a y such that $\Gamma(y, \mathbf{Z} \oplus \mathbf{B})[s]$ is defined and not equal to $H(y)[s]$, then for the least such y , enumerate $\gamma(y, \mathbf{Z} \oplus \mathbf{B})[s]$ into \mathbf{B} and call the procedure *CODE*. For each argument z greater than or equal to y , reset $\gamma(z, \mathbf{Z} \oplus \mathbf{B})$ to be greater than $\gamma(y, \mathbf{Z} \oplus \mathbf{B})[s]$. Go to 2b. Here, $t1(\mathbf{Z}, \Psi)$ requires that Γ be corrected by a change in \mathbf{B} but does not specify which component of \mathbf{B} should be changed. That decision can be made by looking at the other strategies in the construction.

(b) Let y be the least number at which $\Gamma(\mathbf{Z} \oplus \mathbf{B})[s]$ is not defined. Let u be the greatest number such that some strategy has reset the use for $\Gamma(\mathbf{Z} \oplus \mathbf{B})$ at y to be greater than that number, if there is such, and let u be 0, otherwise.

i. If $\gamma(y, \mathbf{Z} \oplus \mathbf{B})$ was defined at an earlier stage t to have a value greater than u then enumerate a computation into Γ , setting $\gamma(y, \mathbf{Z} \oplus \mathbf{B})[s]$ equal to $\gamma(y, \mathbf{Z} \oplus \mathbf{B})[t]$ and $\Gamma(y, \mathbf{Z} \oplus \mathbf{B})[s]$ equal to $H(y)[s]$. Go to 3.

ii. Otherwise, if possible, define g to be the smallest element of R which is in $(r_0, r)[s]$ and is greater than any previous value of γ and greater than u . Enumerate a computation into Γ , setting $\gamma(y, \mathbf{Z} \oplus \mathbf{B})[s]$ equal to g and $\Gamma(y, \mathbf{Z} \oplus \mathbf{B})[s]$ equal to $H(y)[s]$. Go to 3.

3. Let u_0 be the stage when $t1(\mathbf{Z}, \Psi)$ most recently entered Step 3.

(a) If s is $t1(\mathbf{Z}, \Psi)$ -*expansionary* over every stage less than or equal to u_0 and s is an opening stage then go to 1.

(b) Otherwise, end $t1(\mathbf{Z}, \Psi)$'s activity for stage s .

We enumerate Γ so that if there are infinitely many stages which are expansionary for the equations $\Psi_{(i,j)}(A_{(i,j)}) = Z_i$ then there are infinitely many stages during which we extend the equality between $\Gamma(\mathbf{Z} \oplus \mathbf{B})$ and H . We leave it to the reader to work through the analysis of $t1(\mathbf{Z}, \Psi)$.

$t2(\mathbf{Z}, \Psi)$: **Enumerating** $\Delta(C) = Z_{i^*}$. Consider the case when $\sigma\{\alpha\}$ equals $t2(\mathbf{Z}, \Psi)$. This strategy is used when an earlier $i3(C, D, \Theta)\{\alpha \parallel l\{\alpha\}\}$ forced $\Gamma(\mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel k\{\alpha\}\}$ to be partial and attempts to ensure that C computes a member of \mathbf{Z} .

Definition 5.5 We define $C\{\alpha\}$ to be $C\{\alpha \parallel l\{\alpha\}\}$ and $(\mathbf{Z}, \Psi)\{\alpha\}$ to be $(\mathbf{Z}, \Psi)\{\alpha \parallel k\{\alpha\}\}$. The *critical index* for α is the least i such that $c\{\alpha\} \not\leq z_i$. Let $i^*\{\alpha\}$ denote the critical index for α .

We enumerate Δ like Γ . During $t2(\mathbf{Z}, \Psi)$ -expansionary stages, we enumerate computations into Δ to define $\Delta(C)$. If Z_{i^*} changes, then we change C to correct the value of $\Delta(C)$, during closing stages for $t2(\mathbf{Z}, \Psi)$,

The analysis of $t2(\mathbf{Z}, \Psi)$ is parallel to that of $t1(\mathbf{Z}, \Psi)$. We label the outcomes 1 and 2, for finitely and infinitely many changes of state. In addition, note that $C\{\alpha\}$ appears in an $i3$ -strategy. Hence, $c\{\alpha\}$ is above $\mathcal{B}(x)$ and so $t2(\mathbf{Z}, \Psi)$ is compatible with r , which requires that $0 \mapsto \emptyset$.

Incomparability strategies to ensure $\Theta(C) \neq D$. Suppose that c and d are elements of P with $c \not\leq d$ and with images C and D . For each Θ , we must ensure $\Theta(C) \neq D$. We have three possible contributions to C : direct coding strategies $D(C, D)$, $t1(\mathbf{Z}, \Psi)$ -strategies which put numbers into \mathbf{B} which then enter C directly or through direct coding strategies and $t2(\mathbf{Z}, \Psi)$ -strategies which similarly put numbers into C , either directly or through direct coding strategies. By the transitivity of \leq in P , the direct coding strategies alone will not require a change in D to cause a change in C . However, either of the other types of strategies could cause numbers to enter C .

Let I and J be the sets of indices i such that $\Gamma\{\alpha \parallel i\}$ is active at α and j such that $\Delta\{\alpha \parallel j\}$ is active at α , respectively. Then, $\sigma\{\alpha\}$ works during expansionary stages for their associated strategies.

There are three types of incomparability strategies, depending on the position of c with respect to $\mathcal{B}(x)$ and $\mathcal{Z}(x)$. We denote these strategies by $i1(C, D, \Theta)$, $i2(C, D, \Theta)$ and $i3(C, D, \Theta)$.

$i1(C, D, \Theta)$: **Incomparability strategies when c is not in $\mathcal{A}(\mathcal{B}(x))$.** Suppose that $c \notin \mathcal{A}(\mathcal{B}(x))$. Let $b_{k^*} \in (\mathcal{B}(x) \setminus \mathcal{B}(c))$ and let B_{k^*} be its image. Let R be the computable set designated for the sole use of $i1(C, D, \Theta)$. We let $i1(C, D, \Theta)$ act as follows.

Program 5.6 1. If n is not defined, let n be the least element of R which is in $(r_0, r)[s]$ and not in yet in D . Restrain n from entering D . Go to 2.

2. (a) If $\Theta(n, C)[s]$ is not equal to 0 by means of a computation with use $\theta(n, C)[s]$ less than $r[s]$ then end $i1(C, D, \Theta)$'s activity for stage s .
(b) Otherwise, go to 3.
3. (a) Restrain all numbers less than $\theta(C, n)[s]$ from entering C or any set being directly coded into C . For each i in I , whenever $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i\}$ enumerates a number y less than $\theta(C, n)[s]$ into \mathbf{B} , enumerate y into B_{k^*} .
(b) Enumerate n into D . Call the procedure *CODE*.

OPEN-CLOSE. The $i3(C, D, \Theta)$ -strategies naturally produce opening and closing stages. For one of the other strategies σ , we build delays into σ to produce them. We abbreviate our later descriptions of strategies by codifying this delay into a procedure *OPEN-CLOSE*:

- Program 5.7** 1. Let u_0 be the stage when *OPEN-CLOSE* was called. If u_0 equals s , then end all activity for stage s . Otherwise, go to 2.
2. If s is an opening stage, then go to 3.
 3. If s is a closing stage, then let u_1 equal s and go to 4.
 4. If $s = u_1$, then end all activity for stage s . Otherwise, return control to σ .

$i2(C, D, \Theta)$: **Incomparability strategies when c is in $\mathcal{A}(\mathcal{B}(x) \cup \mathcal{Z}(x))$.** Suppose that c is above $\mathcal{B}(x)$ and also above $\mathcal{Z}(x)$. By definition of h , c is also above h . Given that $c \not\geq d$, we have $h \not\geq d$. Consequently, it is possible to preserve H while changing D . Thus, we can prevent the Γ -strategies from causing changes in C by preserving H .

Let J_C equal the set of j in J such that $c\{\alpha \parallel j\} \leq c$. If j is in J_C , a change in $Z_{i^*}\{\alpha \parallel j\}$ is then reflected by a change in C through the functional $\Delta\{\alpha \parallel j\}$. In this section, we use the fact that c is in $\mathcal{A}(\mathcal{Z}(x))$ to conclude that we can indirectly restrain numbers from entering any $Z_{i^*}\{\alpha \parallel j\}$ by restraining C . For each j in J_C , fix $\Psi\{\alpha \parallel j\}$ so that the enumeration of $\Delta(C)\{\alpha \parallel j\}$ is predicated on the existence of infinitely many expansionary stages for $\Psi\{\alpha \parallel j\}(C) = Z_{i^*}\{\alpha \parallel j\}$. Such a $\Psi\{\alpha \parallel j\}$ exists since $c \in \mathcal{A}(\mathcal{Z}(x))$.

Definition 5.8 During stage s of the construction, we say that $C \parallel p$ is *stable* if for all x and all j in J_C ,

$$\delta(x, C)\{\alpha \parallel j\}[s] < p \rightarrow \left[\begin{array}{l} \Delta(x, C)\{\alpha \parallel j\}[s] = Z_{i^*}(x)\{\alpha \parallel j\}[s] \\ \text{and } \psi(x, C)\{\alpha \parallel j\}[s] < p. \end{array} \right]$$

Lemma 5.9 *If $C[s] \upharpoonright p$ is stable then none of the strategies along α enumerating active functionals will initiate the enumeration of any numbers into C below p after stage s .*

We omit the proof of the previous lemma. Our implementation, $i2(C, D, \Theta)$ of the Friedberg-Mučnik strategy in this environment goes as follows during stage s . As before, R is the computable set designated for the sole use of $i2(C, D, \Theta)$ and $(r_0, r)[s]$ is the interval in which $i2(C, D, \Theta)$ can act during stage s .

Program 5.10 1. *If n has not yet been defined and it is possible to do so, choose n to be the least element of R which is in $(r_0, r)[s]$ and not yet in D . Restrain n from entering D . Go to 2. Otherwise, end stage s .*

2. (a) *Call the procedure OPEN-CLOSE. When it returns control go to 2b.*
 - (b) i. *If $\Theta(n, C)[s]$ is not defined to be equal to 0 by means of a computation with use $\theta(n, C)[s]$ less than $r[s]$ or if the computation of $\Theta(n, C)[s]$ is different from what it was during the last stage when $i2(C, D, \Theta)$ reached 2b then go to 2a. We wait for a computation setting $\Theta(n, C)$ equal to 0.*
 - ii. *Otherwise, go to 3.*
3. (a) *If J_C is empty then go to 4.*
 - (b) *Otherwise, call the procedure STABILIZE in its initial state on the argument $\theta(n, C)[s]$ and the set J_C . We attempt to find a stable initial segment of C which includes $\theta(n, C)[s]$.*
 - i. *If STABILIZE returns with a change in C then go to 2. The computation of $\Theta(C, n)$ changed before we could stabilize it.*
 - ii. *If STABILIZE returns with p and $C[s]$ is stable below p , go to 4.*
4. (a) *Restrain all numbers less than or equal to p from entering C or any set being coded into C . We can preserve the computation of $\Theta(C, n)[s]$ by preserving $C[s] \upharpoonright p$.*
 - (b) *Enumerate n into D .*

Now, we present the procedure STABILIZE. It records the stage s_0 when it is called; it accepts a number input k and a set input J_S contained in the set of indices for strategies on α enumerating active Δ -functionals; and it attempts to find a stable initial segment of C larger than k .

Program 5.11 1. (a) If $C[s] \Vdash k$ is not equal to $C[s_0] \Vdash k$ then return to the main program with a C change below k .

(b) Otherwise, go to 2.

2. (a) Let l_J be the size of J_S . If there are fewer than l_J members of R in $(r_0, r)[s]$ which are larger than k and larger than any number previously used as the value of some w then end $i2(C, D, \Theta)$'s activity for this stage. If $r[s]$ goes to infinity, we cannot wait in this state forever.

(b) Otherwise, in decreasing order of j in J_S , for each j for which w_j is not defined, set w_j to a value from $(r_0, r)[s] \cap R$ which is larger than k and larger than any number previously used as the value of some w_j . Go to 3. We will either find a stable point in C or make at least one of $\Delta_j(C)$ undefined at its limiting value for w_j , where j is an element of J .

3. Let p equal the maximum of $\{\psi\{\alpha \Vdash j\}(w_j, C)[s] \mid j \in J_S\}$. We attempt to make $C \Vdash p$ stable with respect to the functionals indexed by J_S .

(a) If C has changed below p since the last stage greater than or equal to s_0 when *STABILIZE* reached 3 then let $j^*[s]$ be the maximal j such that C changed below $\psi\{\alpha \Vdash j\}(w_j, C)[s]$. Go to 3(a)i.

i. At the next closing stage for $i2(C, D, \Theta)$, for each i and j greater than or equal to $j^*[s]$ which are indices for $t1$ and $t2$ strategies respectively, enumerate $\gamma(w_{j^*[s]}, \mathbf{Z} \oplus \mathbf{B})\{\alpha \Vdash i\}[s]$ into \mathbf{B} and enumerate $\delta(w_{j^*[s]}, C)\{\alpha \Vdash j\}[s]$ into $C\{\alpha \Vdash j\}$. Call *CODE*. Require that any future axioms for these functionals at arguments greater than or equal to $w_{j^*[s]}$ have use greater than p .

ii. Cancel the value of each w_j for which j is less than $j^*[s]$.

iii. Call the procedure *OPEN-CLOSE*. When it returns control, cancel the value of j^* and go to 1.

If we are unable to obtain a stable segment with which to preserve C , then we locate the lowest priority functional which causes trouble and we force all the functionals of equal or lower priority to be partial.

(b) Otherwise, go to 4.

4. (a) If there is no j in J , such that $\delta(w_j, C)\{\alpha \parallel j\}[s]$ is less than or equal to p then return to the main program with value p on which $C[s]$ is stable.
- (b) Otherwise, let $j^*[s]$ be the maximal j in J such that $\delta_j(w_j, C_j)[s]$ is less than or equal to p . Go to 4(b)i.
- i. At the next closing stage for $i2(C, D, \Theta)$, for each i and j greater than or equal to $j^*[s]$ which are indices for $t1$ and $t2$ strategies respectively, enumerate $\gamma(w_{j^*[s]}, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}[s]$ into \mathbf{B} and enumerate $\delta(w_{j^*[s]}, C)\{\alpha \parallel j\}[s]$ into $C\{\alpha \parallel j\}$. Call *CODE*. Require that any future axioms for these functionals at arguments greater than or equal to $w_{j^*[s]}$ have use greater than p .
- ii. Call the procedure *OPEN-CLOSE*. When it returns control, cancel the value of j^* and go to 1.

Outcomes. Let j_1, \dots, j_l enumerate J_C in increasing order.

There are two possible states in which $i2(C, D, \Theta)$ could reach a limit. First, $i2(C, D, \Theta)$ could reach Step 2 and never find a computation of $\Theta(C, n)$. We amalgamate this outcome with another below. In outcome 4, either J_C is empty or at some stage s , *STABILIZE* returns a p such that $C[s] \parallel p$ is stable and $i2(C, D, \Theta)$ moves to Step 4. Once it reaches Step 4, there is no instruction by which $i2(C, D, \Theta)$ can leave and $i2(C, D, \Theta)$ reaches a limit there.

The only other step from which $i2(C, D, \Theta)$ does not have an automatic exit is Step 2a in *STABILIZE*. By the assumption that $r[s]$ goes to infinity and R is infinite, *STABILIZE* cannot have Step 2a as its limiting state; eventually $r[s]$ will move past the first l many unused elements of R . Thus, under the operating hypothesis, the above two cases are the only ones in which the state of $i2(C, D, \Theta)$ reaches a limit.

In addition to the finite outcomes, $i2(C, D, \Theta)$ has two types of infinite outcomes: $i2(C, D, \Theta)$ could call *STABILIZE* infinitely often without its ever returning a stable initial segment of C and $i2(C, D, \Theta)$ could call *STABILIZE* at stage s_0 after which *STABILIZE* never returns control to $i2(C, D, \Theta)$.

We let 2 denote the outcome in which there are infinitely many stages during which $i2(C, D, \Theta)$ is in Step 2. This can happen either by $i2(C, D, \Theta)$'s reaching a limit there or by returning there infinitely often. In the second case in the analysis of *STABILIZE*, *STABILIZE* is active for every $i2(C, D, \Theta)$ -stage after s_0 . Note, once w_{j_i} is defined its value is never canceled. Thus, there is a least j such that the value of w_j reaches a limit. Let j^* be the least such j . We denote this outcome by Sj^* and remark that it depends on the value of j^* .

Approximating the outcome. If $i2(C, D, \Theta)$ is in Step 2 at the end of its activity for stage s , then approximate the outcome of $i2(C, D, \Theta)$ as 2. If $i2(C, D, \Theta)$ has called STABILIZE and STABILIZE has executed either Step 3(a)i or Step 4(b)i, then approximate the outcome of $i2(C, D, \Theta)$ as $Sj^*[s]$. If $i2(C, D, \Theta)$ is in Step 4 during stage s , then approximate the outcome of $i2(C, D, \Theta)$ as 4 with the exception that if $i2(C, D, \Theta)$ enters Step 4 for the first time during an opening stage s , then we end stage s with no approximation to the outcome of $i2(C, D, \Theta)$. We order the outcomes by $4 <_L 2 <_L Sj_1 <_L \dots Sj_l$. For each approximated outcome o , s is an opening or closing stage for $\alpha^\wedge\langle i2(C, D, \Theta), o \rangle$ if and only if it is one for α .

We define $r_0\{\alpha^\wedge\langle i2(C, D, \Theta), Sj^* \rangle\}[s]$ to be the maximum of $|\alpha|$ and $r_0\{\alpha\}[s]$ and let $r\{\alpha^\wedge\langle i2(C, D, \Theta), Sj^* \rangle\}[s]$ be the minimum of the sets $\{\gamma(w_{j^*}, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\} \mid i > j^* \text{ and } \sigma\{\alpha \parallel i\} = t1(\mathbf{Z}, \Psi)\{\alpha \parallel i\}\}[s]$ and $\{\delta(w_{j^*}, C)\{\alpha \parallel j\} \mid j \geq j^* \text{ and } \sigma\{\alpha \parallel j\} = t2(\mathbf{Z}, \Psi)\{\alpha \parallel j\}\}[s]$. We let $r_0\{\alpha^\wedge\langle i2(C, D, \Theta), 2 \rangle\}[s]$ be the maximum of $|\alpha|$ and $r_0\{\alpha\}[s]$ and let $r\{\alpha^\wedge\langle i2(C, D, \Theta), 2 \rangle\}[s]$ be $r\{\alpha\}[s]$. We let $r_0\{\alpha^\wedge\langle i2(C, D, \Theta), 4 \rangle\}[s]$ be the maximum of $|\alpha|$, $r_0\{\alpha\}[s]$ and the restraint imposed by $i2(C, D, \Theta)$ in Step 4 and let $r\{\alpha^\wedge\langle i2(C, D, \Theta), 4 \rangle\}[s]$ be $r\{\alpha\}[s]$.

Compatibility. In outcome SJ^* , there is a stage s_0 during which $i2(C, D, \Theta)$ calls STABILIZE and after which control is never returned to $i2(C, D, \Theta)$. After stage s_0 , the approximation to the outcome cannot equal 2 or 4. The approximated outcome to $i2(C, D, \Theta)$ during stage s will be Sj , where j is the greatest index whose value of w was not canceled during stage s . By definition, j^* is the least j such that the value of w_j is only canceled finitely often. Thus, the approximation to the outcome of $i2(C, D, \Theta)$ will equal Sj^* infinitely often and only be to its left finitely often. This verifies the true path assumptions for $\alpha^\wedge\langle i2(C, D, \Theta), Sj^* \rangle$. We defined $r_0\{\alpha^\wedge\langle i2(C, D, \Theta), Sj^* \rangle\}[s]$ to equal the maximum of $|\alpha|$ and $r_0\{\alpha\}[s]$ and so it reaches a limit. We defined $r\{\alpha^\wedge\langle i2(C, D, \Theta), Sj^* \rangle\}[s]$ to equal the minimum of $r\{\alpha\}[s]$ and the elements of the sets $\{\gamma(w_{j^*}, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\} \mid i > j^* \text{ and } \sigma\{\alpha \parallel i\} = t1(\mathbf{Z}, \Psi)\{\alpha \parallel i\}\}[s]$ and $\{\delta(w_{j^*}, C)\{\alpha \parallel j\} \mid j \geq j^* \text{ and } \sigma\{\alpha \parallel j\} = t2(\mathbf{Z}, \Psi)\{\alpha \parallel j\}\}[s]$. Since we reset the use for these functionals infinitely often, this minimum goes to infinity. Our positive action is to enumerate these uses into their relevant sets, so after the last stage in which the value of w_{j^*} is canceled we do not enumerate any number less than this minimum into any set. This verifies the interval of activity assumptions for $\alpha^\wedge\langle i2(C, D, \Theta), Sj^* \rangle$. Finally, we call OPEN-CLOSE each time we execute Step 3(a)i or Step 4(b)i so there are infinitely many opening and closing stages for $\alpha^\wedge\langle i2(C, D, \Theta), Sj^* \rangle$.

Next consider outcome 2. If this outcome is achieved because $i2(C, D, \Theta)$ reaches a limit in Step 2, the operating hypothesis holds for $\alpha^\wedge \langle i2(C, D, \Theta), 2 \rangle$. Assume instead that outcome 2 is achieved because $i2(C, D, \Theta)$ calls STABILIZE infinitely often and control is always returned with a change in C below $\theta(n, C)[s_0]$. Note that once $i2(C, D, \Theta)$ enters Step 4 it cannot exit from that step. So, in outcome 2 the approximation to the outcome of $i2(C, D, \Theta)$ can never equal 4. Each time that $i2(C, D, \Theta)$ executes Step 2, the approximation to the outcome of $i2(C, D, \Theta)$ equals 2. Thus, 2 is the $<_L$ least outcome the approximation takes infinitely often. This verifies the true path assumptions for $\alpha^\wedge \langle i2(C, D, \Theta), 2 \rangle$. Since $(r_0, r) \{ \alpha^\wedge \langle i2(C, D, \Theta), 2 \rangle \} [s]$ is essentially equal to $(r_0, r) \{ \alpha \} [s]$ the interval of activity assumptions are satisfied as well. Finally, each time $i2(C, D, \Theta)$ enters Step 2, it calls OPEN-CLOSE so there are infinitely many opening and closing stages for $\alpha^\wedge \langle i2(C, D, \Theta), 2 \rangle$. Thus, the operating hypothesis holds for $\alpha^\wedge \langle i2(C, D, \Theta), 2 \rangle$.

Outcome 4 preserves the operating hypothesis by the usual argument.

Note, that $i2(C, D, \Theta)$ does not enumerate any number into the image of 0 and does not restrain numbers from entering the image of 1.

Efficacy. If $i2(C, D, \Theta)$ has outcome Sj^* then $\Psi(w_{j^*}, C) \{ \alpha \parallel j^* \}$ is not defined in the limit. Since w_{j^*} reaches a limit, for each j greater than j^* , w_j also reaches a limit and so does the computation of $\Psi(w_j, C) \{ \alpha \parallel j \}$. Otherwise, the value of w_{j^*} would be canceled infinitely often in Step 3(a)ii.

First consider j^* equal to j_1 . In this case, if $\Psi(w_{j^*}, C) \{ \alpha \parallel j^* \}$ converges then p reaches a limit. But then, after redefining the uses of the $\Delta \{ \alpha \parallel j_m \}$'s in Step 4 finitely many times, STABILIZE reaches Step 4a and returns control to the main program for $i2(C, D, \Theta)$. This is contrary to assumption. Otherwise, if $\Psi(w_{j^*}, C) \{ \alpha \parallel j^* \}$ converges in the limit, then once the value of w_{j^*} and the computation of $\Psi(w_{j^*}, C) \{ \alpha \parallel j^* \}$ are fixed there is no mechanism by which the value of w on the predecessor of j^* is canceled. This contradicts the choice of j^* . In either case, there are infinitely many stages during which the approximation to the computation of $\Psi(w_{j^*}, C) \{ \alpha \parallel j^* \}$ changes. Thus, $\Psi(w_{j^*}, C) \{ \alpha \parallel j^* \}$ diverges.

In this case we make each $\Gamma \{ \alpha \parallel i \}$ and $\Delta \{ \alpha \parallel j \}$ with index greater than or equal to j^* undefined at the limiting value of w_{j^*} . Each such strategy thus has a computable effect. It can enumerate a number n only so long as n is less than the minimum of $\gamma(w_{j^*}, \mathbf{Z} \oplus \mathbf{B}) \{ \alpha \parallel i \} [s]$ and $\delta(w_{j^*}, C) \{ \alpha \parallel j \} [s]$ for i and j greater than or equal to j^* , which goes to infinity as s does.

Our cancellation of these strategies is appropriate since they are predicated on the assumption that $\Psi \{ \alpha \parallel j^* \} (C)$ is total. For the strategies earlier in α

than j^* , we redefine their w_j 's infinitely often, whenever we act with $j^*[s]$ equal to j^* . Thus, if i and j are less than j^* then we only affect the value of $\gamma(\mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}$ and $\delta(C)\{\alpha \parallel j\}$ finitely often at each argument. This outcome provides a Σ_3 -outcome for the strategy associated with $T(\mathbf{Z}, \Psi)\{\alpha \parallel j^*\}$, injures all the Π_3 -strategies of lower priority than this one and is compatible with all of the Π_3 -strategies of higher priority.

In outcome 2, either there is a stage after which $\Theta(n, C)[s]$ is never equal to $D(n)[s]$ or each time STABILIZE is called, STABILIZE later returns with a change in C below $\theta(n, C)[s]$. In either case, $\Theta(C, n)$ is not equal to $D(n)$. Further, each time that STABILIZE is called it is put into its initial state and chooses a new value for each of the w_j 's infinitely often. So, there is no w and no i or j such that $i2(C, D, \Theta)$ requires an increase in the use of Γ_i or Δ_j infinitely often at w . Thus, $i2(C, D, \Theta)$ satisfies the diagonalization requirement without interfering with the strategies of higher priority.

In outcome 4, during stage s $i2(C, D, \Theta)$ preserves a stable computation of an inequality between $\Theta(n, C)[s]$ and $D(n)[s]$. By Lemma 5.9, no $t1$ -strategy of higher priority than $i2(C, D, \Theta)$ will initiate a change in $C \parallel \theta(n, C)[s]$. Since c is not above d , the completeness strategy will not change C . Since $i2(C, D, \Theta)$ restrains C below $\theta(n, C)[s]$, no strategy of lower priority than $i2(C, D, \Theta)$ can change $C \parallel \theta(n, C)[s]$. Of course, $i2(C, D, \Theta)$ itself does not change $C \parallel \theta(n, C)[s]$ once it enters Step 4. Thus, $C[s] \parallel \theta(n, C)[s]$ is equal to $C \parallel \theta(n, C)[s]$ and $i2(C, D, \Theta)$ establishes an inequality between $\Theta(C)$ and D .

$i3(C, D, \Theta)$: Incomparability strategies for c not in $\mathcal{A}(\mathcal{B}(x)) - \mathcal{A}(\mathcal{Z}(x))$. Let $c \in \mathcal{A}(\mathcal{B}(x)) - \mathcal{A}(\mathcal{Z}(x))$. Consider the requirement $\Theta(C) \neq D$. Now, D may be equal to H or coded into H . Our enumerating numbers into D sets off a chain reaction: the D change implies an H change and the H change requires a correction to the active $\Gamma\{\alpha \parallel i\}$'s. If at least one $\mathbf{Z}\{\alpha \parallel i\}$ does not change then we must change \mathbf{B} ; the \mathbf{B} change is coded into C ; and the computation of $\Theta(C, n)$ is destroyed. If this occurs infinitely often, then one of the sequences, say $\mathbf{Z}\{\alpha \parallel i\}$, must remain unchanged during infinitely many of the periods between our enumerating numbers into D (*i.e.* providing an opening stage) and our changing C (during a closing stage). In effect, $\mathbf{Z}\{\alpha \parallel i\}$ cannot change unless C changes. This is exactly what we need to preserve the operating hypothesis while introducing a strategy $t2(\mathbf{Z}, \Psi)$ to code $Z_{i^*}\{\alpha \parallel i\}$ into C .

By Definition 5.2, z_1, \dots, z_q is an enumeration of $\mathcal{Z}(x)$, $a_{(i,1)}, \dots, a_{(i,q_i)}$ is an enumeration of $\mathcal{A}(z_i)$, and $A_{(i,j)}$ is the image of $a_{(i,j)}$. We defined $i^*\{\alpha\}$ to be the least i such that z_i is not below $c\{\alpha\}$. By the definition of $\mathcal{Z}(x)$,

for each i less than or equal to q there is an $a_{(i,j)}$ such that $a_{(i,j)}$ is not above $\mathcal{B}(x)$. We let $j^*\{\alpha\}$ be the least j such that $a_{(i^*\{\alpha\},j)}$ is not above $\mathcal{B}(x)$. For the following, let i^* and j^* be $i^*\{\alpha\}$ and $j^*\{\alpha\}$, respectively.

Recall, I is the set of i such that $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i\}$ is enumerating an active Γ -functional and let J be the set of j such that $t2(\mathbf{Z}, \Psi)\{\alpha \parallel j\}$ is enumerating an active Δ -functional relative to $C\{\alpha \parallel j\}$ and $c\{\alpha \parallel j\}$ is less than or equal to c in P . For each index in I there is a possibility that the strategy with that index will prevent our preserving C . The strategies with indices in J can be prevented from causing numbers to enter C by coordinating the actions of $i3(C, D, \Theta)$ with those of strategies of higher priority.

We must preserve an initial segment $C \parallel r$ of C while changing D and respecting the constraints of the strategies enumerating the active functionals causing numbers to enter C . We will use the operating hypothesis to control the $\mathbf{Z}\{\alpha \parallel j\}$ computed by active functionals $\Delta\{\alpha \parallel j\}$ to preserve $Z_{i^*}\{\alpha \parallel j\} \parallel r$ during periods in which we allow numbers to enter sets not below C ; at other times, we can hold $Z_{i^*}\{\alpha \parallel j\} \parallel r$ by holding the computation of $Z_{i^*}\{\alpha \parallel j\} \parallel r$ from $\mathbf{A}_{(i^*,j^*)\{\alpha \parallel j\}}$; and we will attempt to clear $C \parallel r$ of all but a fixed set of values of the $\gamma\{\alpha \parallel i\}$. If we are unable to succeed as above, then we will make some active $\Gamma\{\alpha \parallel i\}$ partial and, in a subsequent strategy, introduce a new functional $\Delta(C\{\alpha\})$ either to compute the i^* th element of $\mathbf{Z}\{\alpha \parallel i\}$ or to demonstrate that some element of $\Psi\{\alpha \parallel i\}$ is not equal to its intend value.

As before, R is the infinite computable set designated for the sole use of $i3(C, D, \Theta)$ and $(r_0, r)[s]$ is the interval in which $i3(C, D, \Theta)$ can act during stage s . Then, $i3(C, D, \Theta)$ acts during stage s as follows.

- Program 5.12**
1. (a) If there is no member of $R \cap (r_0, r)[s]$ then end $i3(C, D, \Theta)$'s activity for this stage. Under the assumption that $r[s]$ goes to infinity, we cannot wait in this state forever.
 - (b) Otherwise, let v be the least element in $(r_0, r)[s] \cap R$. Go to 2. We will attempt to find a computation relative to C which is clear of values of the $\gamma\{\alpha \parallel i\}$ for i in I and arguments greater than or equal to v .
 2. If n is not defined, choose n to be the least element of R which is in $(r_0, r)[s]$ and not yet in D . Restrain n from entering D . Go to 3. We choose the witness n and attempt to make $\Theta(C, n)$ different from $D(n)$.
 3. (a) Call the procedure *OPEN-CLOSE*. When it returns control go to 3b.
 - (b) If $\Theta(n, C)[s]$ is not equal to 0 by means of a computation with use $\theta(n, C)[s]$ less than $r[s]$ or if the computation of $\Theta(n, C)[s]$ is different from what it was during the most recent stage when $i3(C, D, \Theta)$

reached 3b then go to 3a. Wait for a computation setting $\Theta(n, C)$ to 0.

- (c) Otherwise, require any new axiom enumerated into an active functional to have use greater than $\theta(n, C)[s]$. Go to 4. Upon seeing $\Theta(n, C)[s] = 0$, we impose the ongoing constraint that if $\mathbf{Z}\{\alpha \parallel i\}$ changes below $\gamma(x, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}$ then the next computation enumerated for $\Gamma(x, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}$ should have use greater than $\theta(n, C)[s]$.
4. (a) If I is empty then go to 11.
- (b) Otherwise, set u_0 equal to the current stage s . Let i^\diamond be the greatest element of I . Go to 5. This begins the preservation recursion. The recursion variable is i^\diamond and it takes values from I in decreasing order. A cycle in the recursion goes from Step 5 to Step 9b. Initially, we set i^\diamond equal to the greatest element of I and let u_0 record the beginning stage.
5. Suspend all activity below $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$ in T . Go to 6. Not everything initiated by strategies below i^\diamond in α is suspended. If one of these strategies has called SWITCH-TO-A-RESTRAINT, then this procedure goes ahead until it requires an expansionary stage for some strategy which is suspended.
- By suspending the actions below $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$, we ensure that for each j in J which is greater than i^\diamond , the following conditions hold. Since j is greater than i^\diamond , $k\{\alpha \parallel j\}$ is also greater than i^\diamond . (See Definition 5.3.) But then, by suspending the strategies below $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$, we have delayed the next $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\{\alpha \parallel j\}\}$ expansionary stage. By the operating hypothesis, $\mathbf{Z}\{\alpha \parallel j\}$ will not change until the first $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\{\alpha \parallel j\}\}$ expansionary stage after s . Thus, we have a temporary restraint on $\mathbf{Z}\{\alpha \parallel j\}$ lasting as long as the suspension is in place.
6. (a) If s is an opening stage for $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$ then drop all restraint imposed by $i3(C, D, \Theta)$. Go to 7. Wait until the next opening stage for $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$ and then drop restraint on sets not below C .
- (b) Otherwise, end $i3(C, D, \Theta)$'s activity for this stage.
7. Call the procedure SWITCH-TO-A-RESTRAINT with inputs $\theta(C, n)[u_0]$ and U defined by $U = \{j \mid j \in J \text{ and } j < i^\diamond\} \cup \{i^\diamond\}$. At the next $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$ expansionary stage (when control is returned to $i3(C, D, \Theta)$) go to 8. As the strategies in $\alpha \parallel i^\diamond$ act, the prohibition built in to the operating hypothesis against their associated $\mathbf{Z}\{\alpha \parallel j\}$'s

changing ends. We switch to directly holding these $\mathbf{Z}\{\alpha \parallel j\}$'s by holding computations of them from sets that we are building.

8. *At the next opening stage for $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$ go to 9. Wait for another opening stage for $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$.*
9. (a) *If either $C[s] \parallel \theta(n, C)[u_0]$ is not equal to $C[u_0] \parallel \theta(n, C)[u_0]$ or $\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i^\diamond\}[s] < \theta(n, C)[u_0]$ then take the following action.*
 - i. *Cancel the values of n and u_0 , drop all restraint previously imposed by $i3(C, D, \Theta)$, and cancel all delays imposed by $i3(C, D, \Theta)$. Go to 9(a)ii.*
 - ii. *At the next closing stage for $i3(C, D, \Theta)$, go to 9(a)iii.*
 - iii. *For every active functional $\Gamma(\mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}$ or $\Delta(C)\{\alpha \parallel j\}$ with i or j greater than or equal to i^\diamond , enumerate $\gamma(\mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}$ into \mathbf{B} and enumerate $\delta(C)\{\alpha \parallel j\}$ into $C\{\alpha \parallel j\}$. Call *CODE*. Require that any future axioms for these functionals at arguments greater than or equal to v have use greater than the current stage. Go to 3.*

Either we lost the computation relative to C , possibly because of a correction to a $\Gamma\{\alpha \parallel i\}$ with $i < i^\diamond$, or $\mathbf{Z}\{\alpha \parallel i^\diamond\}$ failed to change below $\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i^\diamond\}[u_0]$ during $[u_0, s)$. We kill off the functionals of lower priority than $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$ and prepare to make C compute $Z_{i^*}\{\alpha \parallel i^\diamond\}$.

- (b) *Otherwise, go to 10. Continue the recursion.*
10. (a) *If i^\diamond is not the least element of I , redefine i^\diamond to be greatest i in I which is less than the current value of i^\diamond and go to 5. Begin the next cycle.*
- (b) *Otherwise, go to 11. The recursion is complete.*
11. *Execute the following steps.*
 - (a) *Enumerate n into D and call *CODE*. Drop the delay imposed on strategies of higher priority. Restrain any strategy of lower priority from enumerating any number less than $\theta(C, n)[u_0]$ into C or any set below C . We held all of the strategies of higher priority in an opening state. Thus, we can infer that no $\mathbf{Z}\{\alpha \parallel j\}$ will change below $\theta(C, n)[u_0]$, unless C changes first. We can change D now.*

- (b) Call the procedure SWITCH-TO-A-RESTRAINT with input J . When control is returned to $i3(C, D, \Theta)$ go to 11c. Again, we switch our hold on the $Z_{i^*}\{\alpha \parallel j\}$ to restraints on the $A_{(i^*, j^*)}\{\alpha \parallel j\}$.
- (c) Restrain all numbers less than or equal to $\theta(n, C)[u_0]$ from entering C or any set being coded into C , except for corrections to the $\Gamma\{\alpha \parallel i\}$ for i in I . Maintain the restraints imposed by SWITCH-TO-A-RESTRAINT. Go to 12. With the exception of corrections to the active $\Gamma\{\alpha \parallel i\}$ at arguments less than v , we can preserve the computation of $\Theta(C, n)$ by preserving $C \parallel \theta(C, n)[u_0]$ and the $A_{(i^*, j^*)}\{\alpha \parallel j\}$.

12. If $C[s] \parallel \theta(n, C)[u_0]$ is not equal to $C[u_0] \parallel \theta(n, C)[u_0]$ then go to 2. Watch for a change in C due to a change in $H \parallel v$.

The procedure SWITCH-TO-A-RESTRAINT accepts inputs p , an integer, and U , a finite set of indices for strategies which enumerate Γ -functionals which are active for $i3(C, D, \Theta)$.

Program 5.13 1. Let j be the least element of J . Let k be $k\{\alpha \parallel j\}$. Go to 2.

2. At the next $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\}$ expansionary stage t , restrain any number which is less than or equal to $\psi_{(i^*, j^*)\{\alpha \parallel j\}}\{\alpha \parallel k\}(p, A_{(i^*, j^*)\{\alpha \parallel j\}})[t]$ from entering $A_{(i^*, j^*)}\{\alpha \parallel j\}$. Go to 3. By preserving its computation from $A_{(i^*, j^*)}\{\alpha \parallel j\}$, we effectively preserve $Z_{i^*}\{\alpha \parallel j\} \parallel p$.

3. (a) If j is the greatest element of U then return control to the main strategy.

(b) Otherwise, redefine j be the least element of U which is larger than the current value of j . Go to 2.

Behavior of SWITCH-TO-A-RESTRAINT. Suppose the procedure SWITCH-TO-A-RESTRAINT is called by $i3(C, D, \Theta)$ with inputs p and U at the end of $i3(C, D, \Theta)$'s action during stage s . During stage s , $i3(C, D, \Theta)$ does not restrain numbers from entering sets which are not below C . Since c is not in $\mathcal{A}(\mathcal{Z}(x))$, for some or all of the $m \leq q$, the length of \mathbf{Z} , numbers may enter all of the elements of \mathbf{A}_m during stage s . Consequently, for some or all j 's in U , the computations of $\mathbf{Z}\{\alpha \parallel j\}$ from the relevant elements of \mathbf{A} may change.

SWITCH-TO-A-RESTRAINT operates by recursion, working through the elements of U in increasing order. Its recursion step, say on j with associated k ,

is to wait until the next $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\}$ expansionary stage and then, in Step 2, preserve enough of $A_{(i^*, j^*)}\{\alpha \parallel j\}$ to hold its computation of $Z_{i^*}\{\alpha \parallel j\} \parallel p$. Recall, $A_{(i^*, j^*)}\{\alpha \parallel j\}$ is an element of $\mathbf{A}_{i^*}\{\alpha \parallel j\}$ which is not required to be above \mathbf{B} . The sets \mathbf{A}_i are fixed, but the values of i^* and j^* depend on j .

Suppose that j^+ is the successor of j in J and is associated with k^+ . Then, the first $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\}$ -expansionary stage after stage s occurs before the first $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k^+\}$ -expansionary stage after stage s . So, by recursion on j in U , SWITCH-TO-A-RESTRAINT begins its cycle for j before the first $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\}$ -expansionary stage after s . So, for all j in U , SWITCH-TO-A-RESTRAINT preserves the computation of $\Psi_{(i^*, j^*)\{\alpha \parallel j\}}\{\alpha \parallel k\}(A_{(i^*, j^*)\{\alpha \parallel j\}}) \parallel p$ which exists at the first $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\}$ -expansionary stage after stage s .

By the operating hypothesis for $i3(C, D, \Theta)$, for each j there are infinitely many $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\}$ -expansionary stages. Thus, SWITCH-TO-A-RESTRAINT cannot have a finite outcome in Step 2. Further, it automatically exits from the other steps, 1 and 3. So SWITCH-TO-A-RESTRAINT will complete its recursion, impose various finite restraints and return control to the main program. It will return control to the main program at the first expansionary stage for $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k_m\}$, where j_m is the greatest element of U .

Outcomes of $i3(C, D, \Theta)$. If we find $\Theta(C, n)[s] = 0$, then we attempt to preserve the computation in Steps 4–10. The recursion spanning these steps is the *preservation recursion*. If, during the preservation recursion, none of the strategies from I prevents us from doing so, then in Step 11 we enumerate n into D and eventually impose a combined restraint on C and the sets $A_{(i^*, j^*)}\{\alpha \parallel j\}$ for j in J .

By the operating hypothesis, R is infinite, $r[s]$ goes to infinity, and no strategy other than $i3(C, D, \Theta)$ can enumerate an element of R . So, $i3(C, D, \Theta)$ will eventually find an element of R in $(r_0, r)[s]$, and $i3(C, D, \Theta)$ cannot come to a limit in Step 1. Further, there is no instruction in $i3(C, D, \Theta)$ to return to Step 1. So this step does not produce an outcome of $i3(C, D, \Theta)$. The value for V , once assigned, is never changed.

Similarly, $i3(C, D, \Theta)$ cannot come to a limit in Step 2. However, this step may be repeated during the execution of $i3(C, D, \Theta)$. We will show below that each time we return to Step 2 (from Step 12) it is because of a change in $H \parallel v$, and thus n must reach a limit.

Step 3 is the first step in which $i3(C, D, \Theta)$ could come to a limit. This occurs if there is a stage after which $\Theta(n, C)$ is never defined and equal to 0 for two consecutive opening stages for $i3(C, D, \Theta)$. Then, the only effect of

$i3(C, D, \Theta)$ is to prohibit the enumeration of n into D . Call this outcome 3.

The preservation recursion uses the recursion variable i^\diamond which takes values from I , in decreasing order. Step 4 sets the initial values of the recursion variables and does not contribute an outcome to $i3(C, D, \Theta)$. A full cycle in the preservation recursion consists of executing Steps 5–9b–10 for a particular value of i^\diamond . In Step 5, $i3(C, D, \Theta)$ imposes a delay on the strategies below $\alpha \parallel i^\diamond$ and then goes to Step 6. In Step 6, $i3(C, D, \Theta)$ waits until $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$, a strategy of higher priority, receives an opening stage. By the operating hypothesis, there are infinitely many such stages. Thus, these steps do not generate outcomes of $i3(C, D, \Theta)$. At the next opening stage for $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$, $i3(C, D, \Theta)$ goes to Step 7 and calls the procedure SWITCH-TO-A-RESTRAINT. As we described above, this procedure returns control to $i3(C, D, \Theta)$ at the next $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i^\diamond\}$ expansionary stage with the sole effect of having imposed a finite amount of restraint on finitely many sets not in $\mathcal{A}(\mathbf{B})$. Upon regaining control, $i3(C, D, \Theta)$ moves to Step 8. As noted above, $i3(C, D, \Theta)$ eventually sees an opening stage and goes to Step 9. Thus, Steps 7 and 8 do not generate outcomes of $i3(C, D, \Theta)$.

There are two possibilities when $i3(C, D, \Theta)$ reaches Step 9 at stage t . In case 9b, C was fixed on numbers below $\theta(n, C)[u_0]$ between stage u_0 and t and $\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i^\diamond\}[t]$ is greater than $\theta(n, C)[u_0]$. In this case, $i3(C, D, \Theta)$ completes the cycle by going to Step 10. Otherwise, in case 9a, $i3(C, D, \Theta)$ exits from the recursion. At the next possible stage, for each $\Gamma(\mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}$ and $\Delta(C)\{\alpha \parallel j\}$ with i and j greater than or equal to i^\diamond , $i3(C, D, \Theta)$ enumerates $\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}[t]$ into \mathbf{B} , enumerates $\delta(v, C)\{\alpha \parallel j\}[t]$ into $C\{\alpha \parallel j\}$; $i3(C, D, \Theta)$, and requires that any future axioms for these functionals at arguments $y \geq v$ have use $u > t$; finally, $i3(C, D, \Theta)$ goes to 3.

Next, there could be an i in I such that $i3(C, D, \Theta)$ executes infinitely many cycles of the preservation recursion with $i^\diamond[s]$ equal to i for which all but finitely many result in $i3(C, D, \Theta)$'s exiting from the recursion through Step 9(a)iii. Since, for each element of I , $i3(C, D, \Theta)$ cannot execute a cycle with $i^\diamond[s]$ equal to that value without having completed cycles for all larger elements of I , the value of i as above is unique. Call it i^\diamond . Let Ri^\diamond denote this outcome of $i3(C, D, \Theta)$.

Consider $i3(C, D, \Theta)$ in outcome Ri^\diamond . First, it puts an infinite set of numbers into \mathbf{B} and the sets $C\{\alpha \parallel j_m\}$ above \mathbf{B} . For each $t1(\mathbf{Z}, \Psi)\{\alpha \parallel i\}$ with i greater than or equal to i^\diamond , the values of $\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}$ go into \mathbf{B} . Similarly, for each $t2(\mathbf{Z}, \Psi)\{\alpha \parallel j\}$ with j greater than i^\diamond , the values of $\delta(v, C)\{\alpha \parallel j\}$ go into $C\{\alpha \parallel j\}$. In Step 9(a)iii, $i3(C, D, \Theta)$ requires that the subsequent values of $\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}$ and $\delta(v, C)\{\alpha \parallel j\}$ increase with such action. Conse-

quently, the set of numbers so enumerated by $i3(C, D, \Theta)$ is computable. Since $i3(C, D, \Theta)$ only completes finitely many cycles for i^\diamond in Step 9b, it only enters finitely many cycles for indices from I which are less than i^\diamond . Thus, the use functions for functionals with those indices are only changed finitely often.

Second, $i3(C, D, \Theta)$ restrains the sets $A_{(i^*, j^*)}\{\alpha \parallel j\}$. This restraint is dropped when $i3(C, D, \Theta)$ enters Step 6. Consequently, strategies operating during $i3(C, D, \Theta)$'s cycles for i^\diamond enumerate numbers into these sets subject to timing constraints.

Third, consider $Z\{\alpha \parallel i^\diamond\}$ in outcome Ri^\diamond . If $i3(C, D, \Theta)$ begins a cycle for i^\diamond during stage t_0 , it also completes a cycle for the next larger i in I in Step 9b during stage t_0 . Thus, $C[t_0] \parallel \theta(n, C)[u_0]$ equals $C[u_0] \parallel \theta(n, C)[u_0]$. For all sufficiently large stages, each time that $i3(C, D, \Theta)$ executes a cycle for i^\diamond it exits from the preservation recursion by Step 9a. Thus, if $i3(C, D, \Theta)$ executes a cycle for i^\diamond and exits during stage t , then either $C \parallel \theta(n, C)[u_0]$ changes during the interval $[u_0, t]$ or $\mathbf{Z}\{\alpha \parallel i^\diamond\} \parallel (\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i^\diamond\}[u_0])$ does not change during that interval.

In its final outcome, $i3(C, D, \Theta)$ completes the preservation recursion on all indices and proceeds to Step 11. At that point, it enumerates n into D , calls SWITCH-TO-A-RESTRAINT, and when control is returned imposes a finite restraint on C and various sets $A_{(i^*, j^*)}\{\alpha \parallel j\}$ for j in J . Then $i3(C, D, \Theta)$ goes to Step 12 and can only leave if C changes below $\theta(n, C)[u_0]$. Because $i3(C, D, \Theta)$'s restraint, this change could only come from one of the active Γ 's. For each of these $\Gamma\{\alpha \parallel i\}$'s, if a change in H at x requires a change in \mathbf{B} below $\theta(n, C)[u_0]$ then $x < v$. For $x \geq v$, $\gamma(x, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}$ was seen to be larger than $\theta(n, C)[u_0]$ during Step 9 or $i3(C, D, \Theta)$ would have exited the recursion during the cycle for i . There are only finitely many numbers less than v so $i3(C, D, \Theta)$ can only exit from Step 12 finitely often. So, Steps 11 and 12 contribute only the finite outcome in which $i3(C, D, \Theta)$ reaches a limit in Step 12. We let 12 denote this outcome.

Let i_1, \dots, i_n be an enumeration of I , in increasing order. We order the outcomes by $12 <_L Ri_1 <_L \dots <_L Ri_n <_L 3$.

Approximating the outcome. If $i3(C, D, \Theta)$ is in Step 3 throughout all of stage s then approximate the outcome of $i3(C, D, \Theta)$ to be 3. If $i3(C, D, \Theta)$ is in Step 12 throughout all of stage s then approximate the outcome of $i3(C, D, \Theta)$ to be 12. If o is either of these outcomes, s is an opening stage or closing stage for $\alpha \hat{\langle} i2(C, D, \Theta), o \rangle$ if and only if it is one for α .

If $i3(C, D, \Theta)$ executes Step 6a during stage s , then approximate the outcome to $i3(C, D, \Theta)$ to be $Ri^\diamond[s]$. Such a stage is an opening stage for

$\alpha^\wedge\langle i3(C, D, \Theta), Ri^\diamond[s] \rangle$. Similarly, if $i3(C, D, \Theta)$ executes Step 9(a)iii during stage s , then approximate the outcome to $i3(C, D, \Theta)$ to be $Ri^\diamond[s]$. Such a stage is an closing stage for $\alpha^\wedge\langle i3(C, D, \Theta), Ri^\diamond[s] \rangle$.

Otherwise, make no approximation to the outcome of $i3(C, D, \Theta)$.

Now we define the interval of activity produced by $i3(C, D, \Theta)$. For outcome 3, $r_0\{\alpha^\wedge\langle i3(C, D, \Theta), 3 \rangle\}[s]$ is equal to the maximum of $|\alpha|$ and $r_0\{\alpha\}[s]$; and $r\{\alpha^\wedge\langle i3(C, D, \Theta), 3 \rangle\}[s]$ is equal to $r\{\alpha\}[s]$. For the outcome Ri^\diamond , $r_0\{\alpha^\wedge\langle i3(C, D, \Theta), Ri^\diamond \rangle\}[s]$ equals maximum of $|\alpha|$ and $r_0\{\alpha\}[s]$; $r\{\alpha^\wedge\langle i3(C, D, \Theta), Ri^\diamond \rangle\}[s]$ is the minimum of $r\{\alpha\}[s]$, $\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}[s]$ for i in I and greater than or equal to i^\diamond , and $\delta(v, C)[s]\{\alpha \parallel j\}$ for j in J and greater than or equal to i^\diamond . For outcome 12, $r_0\{\alpha^\wedge\langle i3(C, D, \Theta), 12 \rangle\}[s]$ is equal to the maximum of $|\alpha|$, $r_0\{\alpha\}[s]$ and the total restraint imposed by $i3(C, D, \Theta)$ in Step 11; $r\{\alpha^\wedge\langle i3(C, D, \Theta), 12 \rangle\}[s]$ is equal $r\{\alpha\}[s]$.

Compatibility. First, we consider the effect of $i3(C, D, \Theta)$ on the operating hypothesis. In each of outcome 3 or 12, $i3(C, D, \Theta)$ reaches a finite limit and preserves the operating hypothesis. In outcome Ri^\diamond , $r_0\{\alpha^\wedge\langle i3(C, D, \Theta), Ri^\diamond \rangle\}[s]$ comes to a limit as it is essentially equal to $r_0\{\alpha\}[s]$. Now, $r\{\alpha^\wedge\langle i3(C, D, \Theta), Ri^\diamond \rangle\}[s]$ equals the minimum of $r\{\alpha\}[s]$, $\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i\}[s]$ for i such that $i \in I$ and $i \geq i^\diamond$, and $\delta(v, C)[s]\{\alpha \parallel j\}$ for j such that $j \in J$ and $j \geq i^\diamond$. Whenever $i3(C, D, \Theta)$ exits its cycle with $i^\diamond[s]$ equal to i^\diamond , all of the uses of these functionals are reset to larger values. Since this happens infinitely often, their minimum goes to infinity monotonically. This verifies the interval of activity assumptions.

If $i3(C, D, \Theta)$ has outcome Ri^\diamond then it executes infinitely many cycles in which $i^\diamond[s] = i^\diamond$. During each such cycle, the approximated outcome will be Ri^\diamond . By the definition of Ri^\diamond , there are only finitely many executions of cycles in which $i^\diamond[s] < i^\diamond$. By the remarks made above, there can only be finitely many during which $i3(C, D, \Theta)$ is in Step 12. Consequently, if $i3(C, D, \Theta)$ has outcome Ri^\diamond , then it is the leftmost outcome approximated infinitely often. This verifies the true path assumptions.

During each of its cycles in which $i^\diamond[s]$ equals i^\diamond , $i3(C, D, \Theta)$ produces opening and closing stages for $\alpha^\wedge\langle i3(C, D, \Theta), Ri^\diamond \rangle$. Thus, there are infinitely many of each. Secondly, each time that $i3(C, D, \Theta)$ enters the preservation recursion with $i^\diamond[s]$ equal to i^\diamond it exits that cycle in Step 9a. This happens exactly if either C changes below $\theta(n, C)[u_0]$ between the beginning of the cycle for i^\diamond and its end or if $\mathbf{Z}\{\alpha \parallel i^\diamond\}$ does not change below $\gamma(v, \mathbf{Z} \oplus \mathbf{B})\{\alpha \parallel i^\diamond\}$ during the same interval. This is the condition in the operating hypothesis required to add a $t2$ -strategy for $T(\mathbf{Z}, \mathbf{B})\{\alpha \parallel i^\diamond\}$.

Finally, $i3(C, D, \Theta)$ is compatible with the computability strategy as usual.

Efficacy. In its outcomes 3 and 12, $i3(C, D, \Theta)$ satisfies its incomparability requirement without causing any of the active functionals for α to be partial. In outcome Ri^\diamond , $i3(C, D, \Theta)$ causes each functional enumerated by a strategy at or below $\alpha \parallel i^\diamond$ to be undefined at v . However, in so doing $i3(C, D, \Theta)$ produces the features required by the operating hypothesis in order to execute a $t2$ -strategy for $T(\mathbf{Z}, \mathbf{B})\{\alpha \parallel i^\diamond\}$.

5.2 Construction

We conclude with an overview of the construction and verification.

Assigning priority. We let \mathcal{P} be a computable list of all of the requirements for our theorem and use $>_{\mathcal{P}}$ to denote the ordering of requirements.

$$(16) \quad \begin{aligned} \mathcal{P} = & C, R, D(C_1, D_1), \dots, D(C_n, D_n), \\ & T(\mathbf{Z}_1, \Psi_1), I1(C_1, D_1, \Theta_1), I2(C_1, D_1, \Theta_1), I3(C_1, D_1, \Theta_1), \dots, \\ & T(\mathbf{Z}_n, \Psi_n), I1(C_n, D_n, \Theta_n), I2(C_n, D_n, \Theta_n), I3(C_n, D_n, \Theta_n), \dots \end{aligned}$$

The priority tree. One defines T , the tree of strategies, by recursion based on priority and outcome, and defines $<_L$ on T by $\alpha <_L \beta$ if at the point at which α and β first disagree the outcome indicated in α is $<_L$ the one indicated in β . Simultaneously with defining T , one defines functions *Active* and *Satisfied* on the nodes of T . If α is a node in T then $Active(\alpha)$ will be the set of strategies labeling nodes in T above α which enumerate functionals which are active from the point of view of α . Similarly, $Satisfied(\alpha)$ will be the finite set of requirements which seem to have been satisfied by the action of strategies above α . We omit specifying T .

Links. During stage s we use a finite subrecursion to generate a path $v[s]$ in T of length no greater than s . We begin at the root of T . Given an initial segment $v[s, t]$ of $v[s]$, we execute the strategy $\sigma\{v[s, t]\}$. Depending on the behavior of $\sigma\{v[s, t]\}$, we approximate its outcome by $o\{tp[s, t]\}$. We define $v[s, t+1]$ to be $v[s, t] \hat{\ } \langle \sigma\{v[s, t]\}, o\{tp[s, t]\} \rangle$ and we continue the subrecursion.

We must modify this picture for the $i3(C, D, \Theta)$ strategies. These strategies affect the timing of strategies of higher priority when they impose delays and they act out of turn when they call the procedure SWITCH-TO-A-RESTRAINT. We introduce *links* between nodes on T which indicate that

some action is pending by the lower strategy that only depends on the operating hypothesis of the higher strategy. We have two sorts of links: soft links and hard links. Both are introduced by the action of $i3$ -strategies. A link goes from $i3(C, D, \Theta)\{\alpha\}$ to an earlier strategy $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\}$, where $k < |\alpha|$.

When $i3(C, D, \Theta)\{\alpha\}$ calls SWITCH-TO-A-RESTRAINT in Step 7, we set up soft links to each of the strategies $t1(\mathbf{Z}, \Psi)\{k\}$ at whose expansionary stages $i3(C, D, \Theta)\{\alpha\}$ imposes restraint on $A_{(i^*, j^*)}\{\alpha \parallel j\}$. When we encounter a soft link we impose restraint with priority α , remove the link, and continue.

We install a hard link when, in Step 5, $i3(C, D, \Theta)\{\alpha\}$ imposes a delay on the strategies below $t1(\mathbf{Z}, \Psi)\{\alpha \parallel k\}$. A hard link indicates that these delayed strategies should not act until $i3(C, D, \Theta)\{\alpha\}$ has exited from its recursion, with the exception of their finishing the action implied by soft links. If we encounter a hard link from $i3(C, D, \Theta)\{\alpha\}$ to $t1(\mathbf{Z}, \Psi)\{\alpha_0\}$ during a $t1(\mathbf{Z}, \Psi)\{\alpha_0\}$ expansionary stage, then we execute and remove all the soft links to α_0 , and go directly to $i3(C, D, \Theta)\{\alpha\}$. We move the hard link up when $i3(C, D, \Theta)\{\alpha\}$ imposes a delay for a smaller i^\diamond in Step 5, and remove it when $i3(C, D, \Theta)\{\alpha\}$ drops all of its delays, as in Steps 9a or 11.

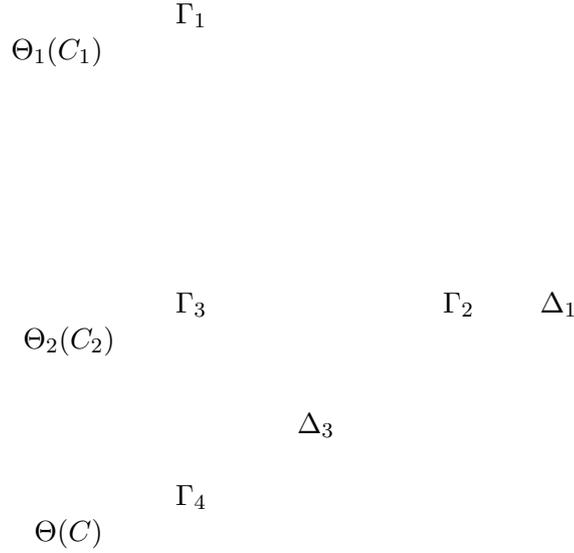


Figure 5: Links between strategies

Figure 5 shows the pattern of links installed during a stage when $i3(C, D, \Theta)\{\alpha \parallel n_2\}$ opens its cycle with i^\diamond equal to n_1 , $i3(C, D, \Theta)\{\alpha \parallel n_6\}$ opens its cycle with i^\diamond equal to n_5 , $i3(C, D, \Theta)\{\alpha \parallel n_{10}\}$ opens its cycle with i^\diamond

equal to n_8 , and $i3(C, D, \Theta)\{\alpha \parallel n_{12}\}$ opens its cycle with i^\diamond equal to n_4 . Soft links are indicated by dashed lines and hard links by solid lines.

All of $i3(C, D, \Theta)\{\alpha \parallel n_6\}$, $i3(C, D, \Theta)\{\alpha \parallel n_{10}\}$ and $i3(C, D, \Theta)\{\alpha \parallel n_{12}\}$ are connected to $t1(\mathbf{Z}, \Psi)\{\alpha \parallel n_1\}$ by soft links. They are relying on $i3(C, D, \Theta)\{\alpha \parallel n_2\}$ to hold $\mathbf{Z}\{\alpha \parallel n_1\}$ until they can restrain $A_{(i^*, j^*)}\{\alpha \parallel n_2\}$. Similarly, $i3(C, D, \Theta)\{\alpha \parallel n_{10}\}$ is connected to $t1(\mathbf{Z}, \Psi)\{\alpha \parallel n_5\}$ and relying on $i3(C, D, \Theta)\{\alpha \parallel n_6\}$. Each of the $i3$ -strategies is connected by a hard link to the $t1$ -strategy indicated by its current value of i^\diamond . Note, none of the hard links cross. One arranges that this is true in general.

Note that $i3(C, D, \Theta)\{\alpha \parallel n_{10}\}$ is not connected to any strategy below $t1(\mathbf{Z}, \Psi)\{\alpha \parallel n_8\}$ and that $i3(C, D, \Theta)\{\alpha \parallel n_{10}\}$ is not connected to any strategy below $t1(\mathbf{Z}, \Psi)\{\alpha \parallel n_4\}$. SWITCH-TO-A-RESTRAINT only works with the strategies of of priority greater than or equal to $\alpha \parallel i^\diamond$.

5.3 Verification

Lemma 5.14 *For all stages $[s, t]$, if there is are hard links between $\alpha_1 \subset \beta_1$ and between $\alpha_2 \subset \beta_2$ at the beginning of substage $[s, t]$ then either the two intervals are disjoint or the closed interval determined by one pair is a subset of the open interval determined by the other pair.*

Two links with the same endpoint would contradict Lemma 5.14, and so for each substage $[s, t]$ and each α in T , there is at most one β below α such that there is a hard link from β to α at the beginning of substage $[s, t]$.

The *true path*, Υ , consists of those α such that there are infinitely many stages s during which there is a substage t with $v[s, t]$ equal to α and there are only finitely many stages during which there is a t such that $v[s, t] <_L \alpha$. Clearly, v is a linearly ordered subset of T .

Lemma 5.15 *For each α in Υ the following conditions obtain.*

1. α is initialized finitely often.
2. After α has been initialized for the final time the environment in which $\sigma\{\alpha\}$ works satisfies the operating hypothesis.
3. After α has been initialized for the final time, the construction executes all of the actions of $\sigma\{\alpha\}$.
4. Let $o\{\alpha\}$ be the outcome of $\sigma\{\alpha\}$ achieved during the construction. The successor of α on Υ is equal to $\alpha \hat{\ } \langle \sigma\{\alpha\}, o \rangle$.

Consequently, Υ is infinite. A requirement S in \mathcal{P} acts at α if α is on Υ and either $\sigma\{\alpha\}$ is a strategy for S or S is an element of $Satisfied\{\alpha^+\} - Satisfied\{\alpha\}$, where α^+ is the immediate successor of α on Υ .

Lemma 5.16 *For each requirement S , S acts finitely often on Υ .*

Consider the requirement $T(\mathbf{Z}_n, \Psi_n)$ and α with $\sigma\{\alpha\} = t1(\mathbf{Z}_n, \Psi_n)$. $T(\mathbf{Z}_n, \Psi_n)$ can act twice without any action by requirements of higher priority: once if there is an α_1 on Υ extending α for which $\sigma\{\alpha_1\}$ equals $t2(\mathbf{Z}_n, \Psi_n)$ and again if there is an α_2 extending α_1 such that $\sigma\{\alpha_2\}$ is $i2(C, D, \Theta\{\alpha_2\})$ and $o\{\alpha_2\}$ is $S|\alpha_1|$. But then S is in $Satisfied\{\alpha_2^+\}$. It can only be removed from $Satisfied$ if some requirement of higher priority acts, contrary to assumption. Thus, $T(\mathbf{Z}_n, \Psi_n)$ acts finitely often on Υ after α .

Lemma 5.17 *For each requirement R in \mathcal{P} , R is satisfied.*

By Lemma 5.16, there is a greatest α on Υ at which R acts. Let α_R be this element of Υ . By the earlier analysis of efficacy, if R is an element of $Satisfied\{\alpha_R\}$, then R is satisfied based on the action of the strategies in Υ appearing on α and the hypothesis that no requirement of higher priority than R acts below α . If $\sigma\{\alpha\}$ is a strategy for R , then either R is an element of $Satisfied\{\alpha^+\}$ and R is satisfied as before or $\sigma\{\alpha\}$ is an element of $Active\{\alpha^+\}$. In the second case, $\sigma\{\alpha\}$ must be in $Active\{\beta\}$ for each β on Υ . Otherwise, R would act again contrary to the choice of α . But then, $\sigma\{\alpha\}$ must produce a total functional. Hence, R is satisfied.

Theorem 5.1 follows directly from Lemma 5.17.

References

- [AS80] Klaus Ambos-Spies. *On the Structure of the Recursively Enumerable Degrees*. PhD thesis, Ludwig-Maximilians-Universität München, Munich, 1980.
- [Dav65] M. Davis. *The undecidable. Basic papers on undecidable propositions, unsolvable problems and computable functions*. Raven Press, Hewlett, N.Y., 1965.
- [Fri57] R. M. Friedberg. Two recursively enumerable sets of incomparable degrees of unsolvability (solution to Post's problem, 1944). *Proc. Nat. Acad. Sci. U.S.A.*, 43:236–238, 1957.

- [FS85] P. A. Fejer and Richard A. Shore. Embeddings and extensions of embeddings in the r. e. tt- and wtt-degrees. In H.-D. Ebbinghaus, G. H. Muller, and Gerald E. Sacks, editors, *Recursion Theory Week*, pages 121–140, Heidelberg, 1985. Springer–Verlag.
- [Göd31] Kurt Gödel. Über formal unentscheidbare sätze der principia mathematica und verwandter systeme I. *Monatsh. Math. Phys.*, 38:349–360, 1931.
- [Göd34] Kurt Gödel. On undecidable propositions of formal mathematical systems. (mimeographed lecture notes, taken by Stephen C. Kleene and J. Barkley Rosser); reprinted with revisions in [Dav65], 1934.
- [HS82] Leo A. Harrington and S. Shelah. The undecidability of the recursively enumerable degrees. *Bull. Amer. Math. Soc.*, 6:79–80, 1982.
- [HS92] Leo A. Harrington and Robert I. Soare. Games in recursion theory and continuity properties of capping degrees. In H. Judah, W. Just, and W. H. Woodin, editors, *Set Theory and the Continuum*, pages 39–62, Heidelberg, 1992. Springer–Verlag.
- [Lac66] Alistair H. Lachlan. Lower bounds for pairs of recursively enumerable degrees. *Proc. London Math. Soc. (3)*, 16:537–569, 1966.
- [Lac75] Alistair H. Lachlan. A recursively enumerable degree which will not split over all lesser ones. *Ann. Math. Logic*, 9:307–365, 1975.
- [Muc56] A. A. Muchnik. On the unsolvability of the problem of reducibility in the theory of algorithms. *Dokl. Akad. Nauk SSSR*, N. S. 108:194–197, 1956.
- [Sac63] Gerald E. Sacks. On the degrees less than $\mathbf{0}'$. *Ann. of Math.*, 77:211–231, 1963.
- [Sac64] Gerald E. Sacks. The recursively enumerable degrees are dense. *Ann. of Math.*, 80:300–312, 1964.
- [Sho65] Joseph R. Shoenfield. Applications of model theory to degrees of unsolvability. In J. W. Addison, L. Henkin, and A. Tarski, editors, *The Theory of Models, Proceedings of the 1963 International Symposium at Berkeley*, Studies in Logic and the Foundations of Mathematics, pages 359–363, Amsterdam, 1965. North–Holland Publishing Co.

- [Soa87] Robert I. Soare. *Recursively Enumerable Sets and Degrees*. Perspectives in Mathematical Logic, Omega Series. Springer-Verlag, Heidelberg, 1987.
- [SS91] Richard A. Shore and Theodore A. Slaman. Working below a low_2 recursively enumerable degree. *Ann. Pure Appl. Logic*, 52:1–25, 1991.
- [Tur39] Alan M. Turing. Systems of logic based on ordinals. *Proc. London Math. Soc. (3)*, 45:161–228, 1939.
- [Yat66] C. E. M. Yates. A minimal pair of recursively enumerable degrees. *J. Symbolic Logic*, 31:159–168, 1966.

UNIVERSITY OF CALIFORNIA, BERKELEY, CA 94720-8204;
email: SLAMAN@MATH.BERKELEY.EDU
UNIVERSITY OF CHICAGO, CHICAGO, ILLINOIS 60637-1546;
email: SOARE@MATH.UCHICAGO.EDU