THE JOURNAL OF SYMBOLIC LOGIC
Volume 48, Number 2, June 1983

MORE ABOUT UNIFORM UPPER BOUNDS ON IDEALS OF TURING DEGREES[1]

HAROLD T. HODES

Abstract. Let I be a countable jump ideal in $\mathscr{D} = \langle$The Turing degrees, $\leq\rangle$. The central theorem of this paper is:

***a** is a uniform upper bound on I iff **a** computes the join of an I-exact pair whose double jump $\boldsymbol{a}^{(1)}$ computes.*

We may replace "the join of an I-exact pair" in the above theorem by "a weak uniform upper bound on I".

We also answer two minimality questions: the class of uniform upper bounds on I never has a minimal member; if $\bigcup I = L_\alpha[A] \cap {}^\omega\omega$ for α admissible or a limit of admissibles, the same holds for nice uniform upper bounds.

The central technique used in proving these theorems consists in this: by trial and error construct a generic sequence approximating the desired object; simultaneously settle definitely on finite pieces of that object; make sure that the guessing settles down to the object determined by the limit of these finite pieces.

Fix recursive pairing and unpairing functions on ω, such that $x = \langle (x)_0, (x)_1\rangle$. For f: $\omega \to \omega$, let $(f)_x(y) = f(\langle x, y\rangle)$. If $\mathscr{F} \subseteq {}^\omega\omega$, f parametrizes $\mathscr{F}$ iff $\mathscr{F} = \{(f)_x | x \in \omega\}$. We depart from standard practice and view Turing degrees as equivalence classes on ${}^\omega\omega$, not $\mathscr{P}(\omega)$, under $\equiv_T$. This has no importance; the following definitions could be rephrased to apply to Turing degrees as usually defined. All degrees in this paper are Turing degrees.

A degree $\boldsymbol{a}$ is a uniform upper bound (u.u.b.) on a class I of degrees iff some $f \in \boldsymbol{a}$ parametrizes $\bigcup I$; $\boldsymbol{a}$ is a weak u.u.b. iff some $f \in \boldsymbol{a}$ parametrizes $\bigcup I \cap {}^\omega 2$. I is an ideal iff I is downward closed under $\leq$ and closed under join. I is a jump ideal iff I is an ideal closed under jump. Where I is an ideal, the pair $(\boldsymbol{b}, \boldsymbol{c})$ is I-exact iff $I = \{\boldsymbol{d} | \boldsymbol{d} \leq \boldsymbol{b} \;\&\; \boldsymbol{d} \leq \boldsymbol{c}\}$. Recent results of Shore imply that there is a degree-theoretic definition of the relation: $\boldsymbol{a}$ is a u.u.b. on I, where I is a countable jump ideal; it is obtained by encoding the analytic definition of a u.u.b. into degree-theoretic terms. The central result of this paper provides a more natural degree-theoretic definition of this relation.

THEOREM 1. *Where I is a countable jump ideal: $\boldsymbol{a}$ is a u.u.b. on I iff there is an I-exact pair $(\boldsymbol{b}, \boldsymbol{c})$, $\boldsymbol{b} \vee \boldsymbol{c} \leq \boldsymbol{a}$ and $(\boldsymbol{b} \vee \boldsymbol{c})^{(2)} \leq \boldsymbol{a}^{(1)}$.*

The technique used in proving the hard direction ($\Rightarrow$) is then extended to answer further questions about u.u.b.s, some of which were raised in [2].

For $\mathscr{F} \subseteq \{(f)_x | x \in \omega\}$, f is a subparametrization of $\mathscr{F}$. Let $f = f_0 \oplus \cdots \oplus f_{n-1}$ iff for all $x, f(x) = f_i((x)_1)$ if $(x)_0 = i < n, f(x) = 0$ otherwise.

Received May 10, 1981.

[1]I wish to thank David Posner for an illuminating discussion which led to all these theorems.

0022-4812/83/4802-0021/$02.70

GUESSING LEMMA. *Let I be an ideal of degrees, f subparametrizes $\bigcup I$. There are two and three-place partial f-recursive functions G and H such that:*

(1) *if* $(f)_{x_0} \oplus \cdots \oplus (f)_{x_{m-1}} \in \bigcup I$ *then* $\lim_n H(m, \langle x_0, \ldots, x_{m-1}\rangle, n)$ *exists and if it is z,* $(f)_z = (f)_{x_0} \oplus \cdots \oplus (f)_{x_{m-1}}$;

(2) *if* $(f)_x^{(1)} \in \bigcup I$, *then* $\lim_n G(x, n)$ *exists and if it is z,* $(f)_z = (f)_x^{(1)}$.

(*Here* $\langle x_0, \ldots, x_{m-1}\rangle$ *is a recursive coding of finite sequences from ω into ω.*)

PROOF. We construct G.

Let

$$g(x, u) = \begin{cases} \text{the least } t \text{ such that } \{u\}^{(f)_x}(u) \text{ converges in } t \text{ steps} \\ \qquad\qquad\qquad\qquad \text{if there is such a } t; \\ 0 \quad \text{otherwise.} \end{cases}$$

$\lambda u.\, g(x, u) \equiv_T (f)_x^{(1)}$. Thus if $(f)_x^{(1)} \in \bigcup I$, $\lambda u.\, g(x, u) \in \bigcup I$. Let h be a non-decreasing function which eventually dominates each member of $\bigcup I$, $h \leq_T f$: for example, $h(z) = \max_{u \leq z}(f)_u(u)$. We shall say that z is a candidate for x at step n iff for every $u < n$:

$$(f)_z(u) = \begin{cases} 1 & \text{if } \{u\}^{(f)_x}(u) \text{ converges in } h(u) + n \text{ steps,} \\ 0 & \text{if not.} \end{cases}$$

Given x, select u_0 such that for all $u \geq u_0$, $h(u) \geq g(x, u)$. Let $n_0 = \max\{g(x, u) \mid u < u_0\}$. For $n \geq n_0$, if z is a candidate for x at step n, $(f)_z \upharpoonright n = (f)_x^{(1)} \upharpoonright n$, since for all u, $g(x, u) \leq h(u) + n$. Let $G(x, n)$ = the least z which is a candidate for x at step n. Suppose that $(f)_x^{(1)} \in \bigcup I$, z_0 is the least z such that $(f)_z = (f)_x^{(1)}$, and n_1 is the least n such that for each $z < z_0$, $(f)_z(n) \neq (f)_{z_0}(n)$ for some $n < n_1$. Then for $n \geq \max(n_0, n_1)$ and any $z < z_0$, z is not a candidate for x at step n. But z_0 *is* one as of step n. So $G(x, n) = z_0$ for such n. The construction of H is easier and we omit it. Q.E.D.

We note the following. Suppose f parametrizes $\bigcup I \cap {}^{\omega}2$ and $\mathbf{0} \in I$. deg(f) is a u.u.b. on I iff there is a $G \leq_T f$ as above which guesses at the location of jumps. This is easy to prove.

Let $g =^* h$ iff for all but finitely many x, $g(x) = h(x)$.

LEMMA 1. *If I is a set of degrees and f is a function such that for every* $g \in \bigcup I$ *there is an e such that* $g =^* (f)_e$, *and for every e,* $(f)_e \in \bigcup I$, *then* deg(f) *is a u.u.b. on I.*

PROOF. Let Seq be the set of sequence numbers, letting $s = \langle (s)_0, \ldots, (s)_{\mathrm{lh}(s)-1}\rangle$. Let

$$(\hat{f})_{\langle e,s\rangle}(x) = \begin{cases} (f)_0(x) & \text{if } s \notin \text{Seq}, \\ (s)_x & \text{if } s \notin \text{Seq} \;\&\; x < \mathrm{lh}(s), \\ (f)_e(x) & \text{if otherwise.} \end{cases}$$

$\hat{f} \leq_T f$ and $\hat{f}$ parametrizes $\bigcup I$. Since the class of u.u.b's on I is closed upwards, deg(f) is a u.u.b. on I. Q.E.D.

PROOF OF THEOREM 1 (⇐). Suppose $(\boldsymbol{b}, \boldsymbol{c})$ is I-exact, $\boldsymbol{b} \vee \boldsymbol{c} \leq \boldsymbol{a}$ and $(\boldsymbol{b} \vee \boldsymbol{c})^{(2)} \leq \boldsymbol{a}^{(1)}$, $A \in \boldsymbol{a}$, $B \in \boldsymbol{b}$, $C \in \boldsymbol{c}$. Since $(B \oplus C)^{(2)} \leq_T A^{(1)}$, recursively in A we may guess at the truth of Π_2^0 sentences about B and C so that in the limit these guesses are correct. Let f be such that

$$(f)_{\langle\langle e_1, e_2\rangle, n\rangle}(x) = \begin{cases} 0 & \text{if for some } t \geq \max(x, n) \text{, the } t\text{th guess is that for some } y \text{, either } \{e_1\}^B(y) \text{ is undefined or } \{e_1\}^B(y) \neq \{e_2\}^B(y) \text{, and either } t = \max(x, n) \text{ or } \{e_1\}_t^B(x) \text{ is undefined;} \\ \{e_1\}^B(x) & \text{otherwise.} \end{cases}$$

$f \leq_T A$. In the otherwise case, $\{e_1\}^B(x)$ is defined, since in the limit our guesses at whether $\neg(\forall y)(\{e_1\}^B(y)$ is defined & $\{e_1\}^B(y) = \{e_2\}^C(y))$ are right. If $\{e_1\}^B$ is total and $\{e_1\}^B = \{e_2\}^C$, then $(f)_{\langle\langle e_1, e_2\rangle, n\rangle} =^* \{e_1\}^B$; otherwise $(f)_{\langle\langle e_1, e_2\rangle, n\rangle} =^* \lambda x. 0$. By Lemma 1, $\boldsymbol{a}$ is a u.u.b. on I.

($\Rightarrow$). Let Str be the set of finite strings of 0's and 1's, coded into ω. For $\sigma, \tau \in$ Str, $\sigma\hat{\ }\tau$ is the concatenation of σ and τ; $\sigma \preccurlyeq \tau$ iff σ extends τ; $\sigma \prec \tau$ iff $\sigma \preccurlyeq \tau$ and $\sigma \neq \tau$. P is a tree iff P: Str $\to$ Str and for all $\sigma, \tau \in$ Str, if $\tau \preccurlyeq \sigma$ then $P(\tau) \preccurlyeq P(\sigma)$. A tree P is perfect iff for all $\sigma \in$ Str, $P(\sigma\hat{\ }\langle 0\rangle)$ is strictly left of $P(\sigma\hat{\ }\langle 1\rangle)$ in the lexicographic ordering of Str. For $C \in {}^\omega 2$, $C \preccurlyeq \sigma$ if σ codes an initial segment of C. Let $B \in [P]$ iff B is a branch of P iff for some $C \in {}^\omega 2$, $B = \lim\{P(\sigma) \mid C \preccurlyeq \sigma\}$. P is uniformly recursively pointed iff for some e: for all $B \in [P]$, $P = \{e\}^B$. We code $B \in {}^\omega 2$ into a tree P, yielding a tree Code(P, B), as follows:

Code(P, B)($\langle\,\rangle$) = $P(\langle\,\rangle)$,

Code(P, B)(σ) = $P(\langle B(0), (\sigma)_0, \ldots, B(\mathrm{lh}(\sigma) - 1), (\sigma)_{\mathrm{lh}(\sigma)-1}\rangle)$ for $\mathrm{lh}(\sigma) \geq 1$.

Abusing notation, we write Code(P, f) for Code(P, graph(f)).

A condition is a pair $\langle P, Q\rangle$ of uniformly recursively pointed perfect trees belonging to $\bigcup I$ such that $P \equiv_T Q$. P is a subtree of Q iff for all $\sigma \in$ Str, $P(\sigma) \preccurlyeq Q(\sigma)$. Where $\langle P, Q\rangle$ and $\langle R, S\rangle$ are conditions, $\langle P, Q\rangle$ extends $\langle R, S\rangle$ iff P and Q are subtrees of R and S, respectively. Code($\langle P, Q\rangle$, f) = $\langle$Code(P, f), Code(Q, f)$\rangle$. For $f \in \bigcup I$, this is a condition.

Let Str(l) = $\{\sigma \mid \sigma \in \mathrm{Str}\ \&\ \mathrm{lh}(\sigma) \leq l\}$. A function P: Str(l) $\to$ Str is a pretree iff P fulfills the definition of a perfect tree, except with domain restricted to Str(l); l is the height of P = ht(P). If P is a perfect tree, $P \restriction$ Str(l) is a pretree of height l. If for each $l < \omega$, P_l is a pretree of height l and $P_l \subseteq P_{l+1}$, $\bigcup_l \langle P_l\rangle$ is a perfect tree. A precondition of height l is a pair of pretrees of height l. Since pretrees and preconditions are finite objects, we code them into ω. A pretree P is a subpretree of a tree or pretree R iff for each $\sigma \in$ dom(P) there is a $\tau \in$ dom(R), $\tau \preccurlyeq \sigma$ and $P(\sigma) = R(\tau)$. If P is a subpretree of R and $\sigma \in$ dom(P), $\sigma \in$ dom(R) and $P(\sigma) \preccurlyeq R(\sigma)$; if, furthermore, R is a pretree, ht(P) $\leq$ ht(R). $\langle P, Q\rangle$ is a subprecondition of a condition or precondition $\langle R, S\rangle$ iff P and Q are subpretrees of R and S, respectively. Suppose that for each $l < \omega$ $\langle P_l, Q_l\rangle$ is a subprecondition of a condition $\langle R, S\rangle$, l = ht($\langle P_l, Q_l\rangle$), $\langle P_{l+1}, Q_{l+1}\rangle$ is a subprecondition of $\langle P_l, Q_l\rangle$, and $\langle\langle P_l, Q_l\rangle\rangle_{l<\omega}$ is recursive in $R \oplus S$; then $\lim_l\langle P_l, Q_l\rangle = \langle \bigcup_l P_l, \bigcup_l Q_l\rangle$ is a condition extending $\langle R, S\rangle$.

For P a pretree and $B \in {}^\omega 2$, we may code as much of B as possible into P, letting:

Code(P, B)($\langle\,\rangle$) = $P(\langle\,\rangle)$,

Code(P, B)(σ) $\simeq$ $P(\langle B(0), (\sigma)_0, \ldots, B(\mathrm{lh}(\sigma) - 1), (\sigma)_{\mathrm{lh}(\sigma)-1}\rangle)$, for $\mathrm{lh}(\sigma) \geq 1$.

Note that if ht(P) = $2l$ or = $2l + 1$, Code (P, B) has height l. We define

"Code(P, f)" and Code($\langle P, Q\rangle, f$) where $\langle P, Q\rangle$ is a precondition, as one would expect.

For P a tree or pretree and $\sigma \in$ Str, we shall say that σ is on P iff for some $\tau \in \mathrm{dom}(P)$, $P(\tau) \preccurlyeq \sigma$. Full is the tree id $\restriction$ Str. Where P is a tree or pretree, Full(P, σ) is the tree or pretree determined by Full(P, σ)(τ) $= P(\sigma\frown\tau)$. Note that if P is a pretree of height l, Full(P, σ) is totally undefined, and so technically not a pretree, if $l < \mathrm{lh}(\sigma)$.

Fix a listing $\langle \psi_j\rangle_{j<\omega}$ of all primitive recursive relations on ${}^{\omega}2 \times {}^{\omega}2 \times \omega \times \omega$. Introducing "$\underline{B}$" and "$\underline{C}$" as uninterpreted predicate constants, let φ_j be "$(\exists x) \neg (\exists y)\, \psi_j\, (\underline{B}, \underline{C}, x, y)$." We now define forcing, for $\langle P, Q\rangle$ a condition.

$\langle P, Q\rangle \Vdash \neg\varphi_j$ iff for all $\langle B, C\rangle \in [P] \times [Q]$, $\langle B, C\rangle \models \neg\varphi_j$;

$\langle P, Q\rangle \Vdash \varphi_j$ iff for some n for all $\langle B, C\rangle \in [P] \times [Q]$,

$$\langle B, C\rangle \models \neg(\exists y)\psi_j(\underline{B}, \underline{C}, \underline{n}, y).$$

[3] contains a proof of the crucial density theorem: any condition extends to a condition deciding φ_j. Implicit in that proof is the construction of a function force(j, $\langle P, Q\rangle$) with domain $\leq \omega$ such that, letting force(j, $\langle P, Q\rangle$)(l) $= \langle \hat{P}(l), \hat{Q}(l)\rangle$:

(1) force(j, $\langle P, Q\rangle$)(l) is, if defined, a subprecondition of $\langle P, Q\rangle$ of height l;

(2) if $l + 1 \in$ dom(force(j, $\langle P, Q\rangle$)),

$$\mathrm{force}(j, \langle P, Q\rangle)(l) = \langle \hat{P}(l + 1) \restriction \mathrm{Str}(l), \hat{Q}(l + 1) \restriction \mathrm{Str}(l)\rangle;$$

(3) for $l \in$ dom(force(j, $\langle P, Q\rangle$)), σ, τ strings of length l, there is a $y_{\sigma,\tau}$ such that $\psi(\hat{P}(l)(\sigma), \hat{Q}(l)(\tau), l, y_{\sigma,\tau})$. (Following a standard convention, "$\psi_j(\sigma, \tau, x, y)$" means "For all $B \prec \sigma$, $C \prec \tau$, $\psi_j(B, C, x, y)$".) To compute force (j, $\langle P, Q\rangle$)(0), we search for strings σ and τ of the same length and for a $y_{\langle\rangle,\langle\rangle}$ so that $\psi_j(P(\sigma), Q(\tau), 0, y_{\langle\rangle,\langle\rangle})$, and let $\hat{P}(0)(\langle\,\rangle) = P(\sigma)$, $\hat{Q}(0)(\langle\,\rangle) = Q(\tau)$. Call these chosen σ and τ, if they exist, $\langle\,\rangle'$ and $\langle\,\rangle''$, respectively. Now suppose that force(j, $\langle P, Q\rangle$)(l) $= \langle \hat{P}(l), \hat{Q}(l)\rangle$ has been computed; for $\rho \in$ Str(l), we suppose that ρ' and ρ'' have been defined, $\hat{P}(l)(\rho) = P(\rho')$, $\hat{Q}(l)(\rho) = Q(\rho'')$. We now try to compute $\hat{P}(l + 1)$ and $\hat{Q}(l + 1)$ on all of Str($l + 1$). By our computation of $\hat{P}(l)$ and $\hat{Q}(l)$ and (2), it suffices to do this for strings of length $l + 1$. Let $\sigma_1, \ldots, \sigma_{2^{l+1}}, \tau_1, \ldots, \tau_{2^{l+1}}$ be two lists of all strings of length $l + 1$. We search for strings $\sigma'_1, \ldots, \sigma'_{2^{l+1}}, \ldots, \tau''_1, \ldots, \tau''_{2^{l+1}}$ all of the same length, and for witnesses y_{σ_i,τ_k}, $i, k \in \{1, \ldots, 2^{l+1}\}$, such that for $\sigma_i = \sigma\frown\langle m\rangle$ and $\tau_k = \tau\frown\langle n\rangle$, $\sigma'_i \preccurlyeq \sigma'\frown\langle m\rangle$ and $\tau''_k \preccurlyeq \tau''\frown\langle n\rangle$, and $\psi_j(P(\sigma'_i), Q(\tau''_k), l + 1, y_{\sigma_i,\tau_k})$; we let $\hat{P}(l + 1)(\sigma_i) = P(\sigma'_i)$, $\hat{Q}(l + 1)(\tau_k) = Q(\tau''_k)$. For details on this search, see [3]. This search is recursive in $P \oplus Q$. So force(j, $\langle P, Q\rangle$) is partial recursive in $P \oplus Q$, uniformly in j and $\langle P, Q\rangle$, by the procedure outlined. "Force (j, $\langle P, Q\rangle$)(l) is defined in q steps" means that according to the procedure just outlined, that computation converges in q steps. If force(j, $\langle P, Q\rangle$) is total, $\lim_l$force (j, $\langle P, Q\rangle$)(l) $= \langle \bigcup_l \hat{P}(l), \bigcup_l \hat{Q}(l)\rangle$ is a condition forcing $\neg\varphi_j$.

On the other hand, suppose force(j, $\langle P, Q\rangle$) is not total. Call $\langle l, \sigma, \tau\rangle$ a j-witness for $\langle P, Q\rangle$ iff $\sigma, \tau \in$ Str, lh(σ) $=$ lh(τ), and $\langle$Full(P, σ), Full(Q, τ)$\rangle \Vdash \neg(\exists y)\psi_j(\underline{B}, \underline{C}, \underline{l}, y)$. We now find a j-witness for $\langle P, Q\rangle$. Let l be the least $l \notin$ dom(force(j,$\langle P, Q\rangle$)). If $l = 0$, let $\sigma = \tau = \langle\,\rangle$. If $l = x + 1$, let $\langle \sigma_i, \tau_k\rangle$ be the least pair selected from the lists $\sigma_i, \ldots, \sigma_{2^l}; \tau_1, \ldots, \tau_{2^l}$, for which we cannot find appropriate σ_i', τ_k'' and y_{σ_i,τ_k}. Letting $\sigma_i = \sigma^0{}^\frown\langle n\rangle$, $\tau_k = \tau^0{}^\frown\langle m\rangle$, let $\sigma = (\sigma^0)'{}^\frown\langle n\rangle$, $\tau = (\tau^0)'{}^\frown\langle m\rangle$. $\langle l, \sigma, \tau\rangle$ is easily seen to be a j-witness for $\langle P, Q\rangle$. Notice that $\mathrm{lh}(\sigma) = \mathrm{lh}(\tau)$, since in defining $\hat{P}(x)$ and $\hat{Q}(x)$ we required that $\mathrm{lh}((\sigma^0)') = \mathrm{lh}((\tau^0)')$. We have just described a procedure recursive in $(P \oplus Q)^{(1)}$ which halts iff force(j, $\langle P, Q\rangle$) is partial, and, if it halts, delivers a j-witness for $\langle P, Q\rangle$. Call this procedure Wit(j,$\langle P, Q\rangle$).

The construction of force(j, $\langle P, Q\rangle$)(0), and then of force(j, $\langle P, Q\rangle$)($l + 1$) given force(j, $\langle P, Q\rangle$)(l), proceeds by working *down* P and Q, thinking of trees as growing downwards. Thus we may extend our definition of force(j, $\langle P, Q\rangle$) to apply to the case in which $\langle P, Q\rangle$ is a precondition. In this case, dom(force(j,$\langle P, Q\rangle$)) is finite, and in fact, $\leq$ ht($\langle P, Q\rangle$).

Fix $f \in \boldsymbol{a}$, parametrizing $\bigcup I$. We wish to construct B, $C \in {}^\omega 2$, $\langle \deg(B), \deg(C)\rangle$ I-exact, $(B \oplus C)^{(2)} \leq_T f^{(1)}$ and $B \oplus C \leq_T f$.

A natural strategy suggests that we try to construct a sequence of conditions $\{\langle P_j, Q_j\rangle\}_{j<\omega}$, and an auxiliary sequence $\{\langle x_j, \sigma_j, \tau_j\rangle\}_{j<\omega}$ such that:

(1) $P_0 = Q_0 =$ Full;

(2) for all j:

(2a) if $x_j \geq 0$ then

$$\langle x_j, \sigma_j, \tau_j\rangle = \mathrm{Wit}(j, \langle P_{2j}, Q_{2j}\rangle)$$

and

$$\langle P_{2j+1}, Q_{2j+1}\rangle = \langle \mathrm{Full}(P_{2j}, \sigma_j), \mathrm{Full}(Q_{2j}, \tau_j)\rangle;$$

(2b) if $x_j = -1$, $\sigma_j = \tau_j = \langle\ \rangle$ and force(j, $\langle P_{2j}, Q_{2j}\rangle$) is total and

$$\langle P_{2j+1}, Q_{2j+1}\rangle = \lim_l \mathrm{force}(j, \langle P_{2j}, Q_{2j}\rangle)(l);$$

(3) for all j,

$$\langle P_{2j+2}, Q_{2j+2}\rangle = \mathrm{Code}(\langle P_{2j+1}, Q_{2j+1}\rangle, (f)_j).$$

Then we shall let $\{B\} = \bigcap_j[P_j]$, $\{C\} = \bigcap_j[Q_j]$. Choice of $\langle P_{2j+2}, Q_{2j+2}\rangle$ insures that $(f)_j \leq_T B$ and $(f)_j \leq_T C$. The genericity of the sequence of conditions insures that if $g \leq_T B$ and $g \leq_T C$, $g \in \bigcup I$.

We also want our construction to be recursive in f. But choice of $\langle P_{2j+1}, Q_{2j+1}\rangle$ or, equivalently, of $\langle x_j, \sigma_j, \tau_j\rangle$, depends on facts about $(P_{2j} \oplus Q_{2j})^{(2)}$ which cannot be decided uniformly in j and recursively in f. A further difficulty appears when we specify the sense in which we would like $\{\langle P_j, Q_j\rangle\}_{j<\omega}$ to be recursive in f. We want an f-recursive function $j \mapsto \langle n_j, m_j\rangle$ such that $P_j = (f)_{n_j}$, $Q_j = (f)_{m_j}$, and such a function may not exist. Instead we proceed by guessing, recursively in f, at the previously described construction.

For $x \geq 1$, let $d(x) = y$ iff $x = 2y + 1$ or $x = 2y + 2$. At stage i of our construction we will have a number $z_i \geq 1$ and, for each $j \leq z_i$, a guess $\langle P_j^i, Q_j^i\rangle$ at $\langle P_j, Q_j\rangle$, and, for each $j \leq d(z_i)$, guesses x_j^i, σ_j^i and τ_j^i at x_j, σ_j and τ_j. P_j^i and Q_j^i are functions, $\mathrm{dom}(P_j^i) = \mathrm{dom}(Q_j^i) \leq \omega$ such that, letting $\langle P_j^i, Q_j^i\rangle(l) = \langle P_j^i(l), Q_j^i(l)\rangle$, $\langle P_j^i, Q_j^i\rangle$ (l) is, if defined, a precondition of height l such that:

(1′) $\langle P_0^i, Q_0^i\rangle$ (l) $= \langle \mathrm{Full} \restriction \mathrm{Str}(l), \mathrm{Full} \restriction \mathrm{Str}(l)\rangle$;

(2′) for all $j \leq d(z_i)$, if $x_j^i \geq 0$,

$$\langle P^i_{2j+1}, Q^i_{2j+1}\rangle(l) \simeq \langle \text{Full}(P^i_{2j}(k+l), \sigma^i_j), \text{Full}(Q^i_{2j}(k+l), \tau^i_j)\rangle,$$

where $\text{lh}(\sigma^i_j) = \text{lh}(\tau^i_j) = k$;
if $x^i_j = -1$, $\sigma^i_j = \tau^i_j = \langle\ \rangle$ and

$$\langle P^i_{2j+1}, Q^i_{2j+1}\rangle(l) \simeq \text{force}(j, \langle P^i_{2j}, Q^i_{2j}\rangle(l'))(l)$$

for an $l' \in \text{dom}(\langle P^i_{2j}, Q^i_{2j}\rangle)$, but large enough for the right-hand side to be defined, if such there be;

(3′) for all $2j + 2 \le z_i$,

$$\langle P^i_{2j+2}, Q^i_{2j+2}\rangle(l) \simeq \text{Code}(\langle P^i_{2j+1}, Q^i_{2j+1}\rangle(2l), (f)_j).$$

For reasons to appear shortly, we need to modify this outline in one respect. In the sequence described by (1)–(3) we shall add, between consecutive conditions $\langle P_j, Q_j\rangle$ and $\langle P_{j+1}, Q_{j+1}\rangle$, an intermediate condition $\langle P^*_j, Q^*_j\rangle$, determined by strings δ_j and ε_j of equal length, so that:

(4*) for all j,

$$\langle P^*_j, Q^*_j\rangle = \langle \text{Full}(P_j, \delta_j), \text{Full}(Q_j, \varepsilon_j)\rangle,$$

with (2) and (3) revised to (2*) and (3*), (2*) saying that $\langle P_{2j+1}, Q_{2j+1}\rangle$ is formed from $\langle P^*_{2j}, Q^*_{2j}\rangle$ in the way in which (2) says it is formed from $\langle P_{2j}, Q_{2j}\rangle$, and (3*) saying that $\langle P_{2j+2}, Q_{2j+2}\rangle$ is formed from $\langle P^*_{2j+1}, Q^*_{2j+1}\rangle$ in the way in which (3) says it is formed from $\langle P_{2j}, Q_{2j}\rangle$. In our guessing construction, at stage i for all $j < z_i$ we shall have guesses δ^i_j and ε^i_j at δ_j and ε_j and guesses $\langle P^{i*}_j, Q^{i*}_j\rangle$ at $\langle P^*_j, Q^*_j\rangle$ given by:

(4′*) for $j < z_i$,

$$\langle P^{i*}_j, Q^{i*}_j\rangle(l) \simeq \langle \text{Full}(P^i_j(k+l), \delta^i_j), \text{Full}(Q^i_j, (k+l), \varepsilon^i_j)\rangle,$$
$$\text{for } k = \text{lh}(\delta^i_j) = \text{lh}(\varepsilon^i_j).$$

(2′) and (3′) are now revised to (2′*) and (3′*), following the obvious analogy with (2*) and (3*).

If our guess converges appropriately, we shall have $(B \oplus C)^{(2)} \le_T f^{(1)}$. To insure that $B \oplus C \le_T f$, we must supplement the guessing procedure just described with a nonguessing process such that for each n we can f-recursively find a stage i which definitely settles the questions "$n \in B$?" and "$n \in C$?".

To this end we construct sequences $\{\beta_i\}_{i<\omega}$ and $\{\gamma_i\}_{i<\omega}$ of strings $\beta_{i+1} \precsim \beta_i$, $\gamma_{i+1} \precsim \gamma_i$, and we make sure that $B = \lim_i \beta_i$, $C = \lim_i \gamma_i$. β_i and γ_i will be fixed at stage i on the basis of our guesses as of stage i. But thereafter any further guesses, including revisions of guesses on the basis of which β_i and γ_i were fixed, must honor the commitments that $B \prec \beta_i$ and $C \prec \gamma_i$. This is where δ^i_j and ε^i_j come in; when we make a decision at stage i about what $\langle P_{j+1}, Q_{j+1}\rangle$ looks like, we shall choose δ^i_j, ε^i_j to "protect" β_i and γ_i; that is, we shall try to make sure that $P_{j+1}(\langle\ \rangle) \precsim P^*_j(\langle\ \rangle) \precsim \beta_i$ and $Q_{j+1}(\langle\ \rangle) \precsim Q^*_{j+1}(\langle\ \rangle) \precsim \gamma_i$. To carry all this out, at stage i we shall actually have to compute, for each $j = z_i$, $\langle P^i_j, Q^i_j\rangle(k^i_j)$ for a certain k^i_j. To this end, we introduce functions l^i_j, $j \le z_i$, and l^{i*}_j, $j < z_i$. Intuitively, $l^i_j(q)$ is the largest l such that we can compute $\langle P^i_j, Q^i_j\rangle(l)$ in $\le q$ steps; $l^{i*}_j(q)$ is the largest l such that we can compute $\langle P^{i*}_j, Q^{i*}_j\rangle(l)$ in $\le q$ steps. l^i_j or l^{i*}_j may be undefined on an initial segment of ω, since it can take a while even to compute $\langle P^i_j, Q^i_j\rangle$ (0) or $\langle P^{i*}_j, Q^{i*}_j\rangle(0)$. But if defined, $l^i_j(q) \in \mathrm{dom}(\langle P^i_j, Q^i_j\rangle)$, and for $q < q'$, $l^i_j(q')$ is defined and $\geq l^i_j(q)$; similarly for l^{i*}_j. If $l^i_{j+1}(q)$ is defined, $\langle P^i_{j+1}, Q^i_{j+1}\rangle(l^i_{j+1}(q))$ is a subprecondition of $\langle P^{i*}_j, Q^{i*}_j\rangle$ $(l^{i*}_j(q))$ with $l^{i*}_j(q)$ defined; if $l^{i*}_j(q)$ is defined, $\langle P^{i*}_j, Q^{i*}_j\rangle$ $(l^{i*}_j(q))$ is a subprecondition of $\langle P^i_j, Q^i_j\rangle$ $(l^i_j(q))$, with $l^i_j(q)$ defined. Furthermore, for $j < z_i$, if $\lim_q l^i_j(q) = \omega$ then $\lim_q l^{i*}_j(q) = \omega$; for $2j + 1 < z_i$, if $\lim_q l^{i*}_{2j+1}(q) = \omega$ then $\lim_q l^i_{2j+2}(q) = \omega$; for $2j < z_i$, if $\lim_q l^{i*}_{2j}(q) = \omega$ then: if $x^i_j \geq 0$, $\lim_q l^i_{2j+1}(q) = \omega$; if $x^i_j = -1$, $\lim_q l^i_{2j+1}(q) = \omega$ iff force(j, $\langle P, Q\rangle$) is total, for $\langle P, Q\rangle = \lim_l \langle P^{i*}_{2j}, Q^{i*}_{2j}\rangle$ (l).

Our informal description of l^i_j and l^{i*}_j could serve as a definition of these functions, but we offer definitions anyway:

$l^i_0(q) = q$;

$l^{i*}_j(q) \simeq l^i_j(q) - \mathrm{lh}(\delta^i_j)$, if $l^i_j(q)$ is defined and $\geq \mathrm{lh}(\delta^i_j)$;

if $x^i_j = -1$, $l^i_{2j+1}(q) \simeq$ the maximum l such that

$$\mathrm{force}(j, \langle P^{i*}_{2j}, Q^{i*}_{2j}\rangle(l^{i*}_{2j}(q)))(l)$$

is defined in $\leq q$ steps;

$l^i_{2j+2}(q) \simeq l$ if $l^1_{2j+1}(q) = 2l$ or $= 2l + 1$.

We shall have an f-recursive increasing function g which serves as a clock, telling us when to stop computing preconditions and move on the stage $i + 1$. The relevant k^i_j will be $k^i_j = l^i_j(g(i))$.

We shall arrange our construction so that at each stage i:

(1.i) $l^i_{z_i}(g(i))$ is defined, with β_i on $P^i_{z_i}(l^i_{z_i}(g(i)))$ and γ_i on $Q^i_{z_i}$ $(l^i_{z_i}(g(i)))$.

In addition to the sequences so far described, we also need a sequence $\{\langle n_j, m_j\rangle\}_{j<\omega}$ such that:

(5) $$\text{for all } j, \quad \langle P_j, Q_j\rangle = \langle (f)_{n_j}, (f)_{m_j}\rangle.$$

We shall also need guess $\langle n^i_j, m^i_j\rangle$ at $\langle n_j, m_j\rangle$ for $j \leq z$. Let $[n, m/\delta, \varepsilon]$ abbreviate $\langle \mathrm{Full}((f)_n, \delta), \mathrm{Full}((f)_m, \varepsilon)\rangle$. For $2j + 1 \leq z_i$, let $2j + 1$ have property 1 at stage i iff $[n^i_{2j}, m^i_{2j}/\delta^i_{2j}, \varepsilon^i_{2j}]$ is a condition, and: if $x^i_j \geq 0$,

$$\langle x^i_j, \delta^i_j, \tau^i_j\rangle = \mathrm{Wit}(j, [n^i_{2j}, m^i_{2j}/\delta^i_{2j}, \varepsilon^i_{2j}]);$$

if $x^i_j = -1$, $\mathrm{Wit}(j, [n^i_{2j}, m^i_{2j}/\delta^i_{2j}, \varepsilon^i_{2j}])$ is undefined. Note that "$2j + 1$ has property 1 at stage i" is Σ^0_3 in $((f)_{n^i_{2j}} \oplus (f)_{m^i_{2j}})$. It would be nice at stage i to have all $2j + 1 \leq z_i$ with property 1. But to keep the construction recursive in f we can only guess at whether a given $2j + 1$ has property 1. We do this by asking the question of our $g(i)$th guess at $((f)_{n^i_{2j}} \oplus (f)_{m^i_{2j}})^{(3)}$, namely

$$(f)_{G(G(G(H(n^i_{2j}, m^i_{2j}, g(i)), g(i)), g(i)), g(i))}.$$

We content ourselves with insuring that at each stage i:

(2.1) for each $2j + 1 \leq z_i$ our $g(i)$th guess at $((f)_{n^i_{2j}} \oplus (f)_{m^i_{2j}})^{(3)}$ says that $2j + 1$ has property 1 at stage i.

This can be checked recursively in f.

For $j \leq z_i$, let j have property 2 at stage i iff $\lim_l \langle P^i_j, Q^i_j\rangle(l) = \langle (f)_{n^i_{2j}}, (f)_{m^i_{2j}}\rangle$, which is a condition. Again, "j has property 2 at stage i" is Σ^0_3 in $((f)_{n^i_{2j}} \oplus (f)_{m^i_{2j}})$.

It would be nice to have all $j \leq z_i$ with property 2 at stage i, so that our guesses at $\langle n_j, m_j \rangle$ accurately reflect our guesses at $\langle P_j, Q_j \rangle$. But, to keep the construction recursive in f, the best we can do is to insure that at each stage i:

(3.i) for each $j \leq z_i$,

$$\langle P_j^i, Q_j^i \rangle (l_j^i(g(i))) = \langle (f)_{n_j^i} \upharpoonright \mathrm{Str}(l_j^i(g(i))), (f)_{m_j^i} \upharpoonright \mathrm{Str}(l_j^i(g(i))) \rangle.$$

Checking this will be recursive in f.

After such extensive previewing, the presentation of the construction may, at least, be brief.

Stage 0. $z_0 = 0$, $g(0) = 0$, $\beta_0 = \gamma_0 = \langle\ \rangle$; for all l, $P_0^0(l) = Q_0^0(l) = \mathrm{Full} \upharpoonright \mathrm{Str}(l)$; select $\langle n_0^0, m_0^0 \rangle$ so that $(f)_{n_0^0} = (f)_{m_0^0} = \mathrm{Full}$.

Stage $i + 1$. Suppose we already have z_i, $g(i)$, $\{\langle P_j^i, Q_j^i \rangle\}_{j \leq z_i}$, $\{\langle \delta_j^i, \varepsilon_j^i \rangle\}_{j < d(z_i)}$, $\{\langle x_j^i, \sigma_j^i, \tau_j^i \rangle\}_{j \leq d(z_i)}$, β_i and γ_i, with (1.i)–(3.i) all true. For $2j + 1 \leq z_i$, let $2j + 1$ be 1-bad at (i, q) iff our $(g(i) + q + 1)$st guess at $((f)_{n_{2j}^i} \oplus (f)_{m_{2j}^i})^{(3)}$ says that $2j + 1$ lacks property 1 at stage i. For $j \leq z_i$, j is 2-bad at $\langle i, q \rangle$ iff

$$\langle P_j^i, Q_j^i \rangle (l_j^i(g(i) + q + 1)) \neq \langle (f)_{n_j^i} \upharpoonright \mathrm{Str}(l_j^i(g(i) + q + 1)), (f)_{m_j^i} \upharpoonright \mathrm{Str}(l_j^i(g(i) + q + 1)) \rangle.$$

Let (δ, ε) be a q-combination for $2j \leq z_i$ iff $\mathrm{lh}(\delta) = \mathrm{lh}(\varepsilon) \leq l_{2j}^i(g(i) + q + 1) = l$ and

(6) either:

(a) our $(g(i) + q + 1)$st guess at $((f)_{n_{2j}^i} \oplus (f)_{m_{2j}^i})^{(3)}$ says that

$$\mathrm{Wit}(j, [n_{2j}^i, m_{2j}^i/\delta, \varepsilon]) = \langle x, \sigma, \tau \rangle$$

in $\leq g(i) + q + 1$ steps for some $\langle x, \sigma, \tau \rangle$ with $\mathrm{lh}(\delta) + \mathrm{lh}(\sigma) \leq l$; or

(b) our $(g(i) + q + 1)$st guess at $((f)_{n_{2j}^i} \oplus (f)_{m_{2j}^i})^{(3)}$ says that $\mathrm{Wit}(j, [n_{2j}^i, m_{2j}^i/\sigma, \varepsilon])$ is undefined, and force $(j, \langle P, Q \rangle)(0)$ is defined in $\leq g(i) + q + 1$ steps for

$$\langle P, Q \rangle = \langle \mathrm{Full}(P_{2j}^i(l), \delta), \mathrm{Full}(Q_{2j}^i(l), \varepsilon) \rangle.$$

Whether (δ, ε) is a q-combination, in fact whether there is a q-combination for a given $2j$, is decidable recursively in f. We shall say that q changes the primary guess at $2j + 1 \leq z_i$ iff: for $k < 2j + 1$, k is neither 1-bad nor 2-bad at (i, q); $2j + 1$ is 1-bad at (i, q); and

(7) there is a q-combination (δ, ε) such that

$$P_{2j}^i(l_{2j}^i(g(i) + q + 1))(\delta) \precsim \beta_i$$

and

$$Q_{2j}^i(l_{2j}^i(g(i) + q + 1))(\varepsilon) \precsim \gamma_i.$$

We shall say that q changes the secondary guess at $j \leq z_i$ iff: for all $k < j$, k is neither 1-bad nor 2-bad at (i, q); j is 2-bad but not 1-bad at (i, q); and

(8) there are strings β and γ on $P_j^i(l_j^i(g(i) + q + 1))$ and $Q_j^i(l_j^i(g(i) + q + 1))$, respectively, $\beta \precsim \beta_i$ and $\gamma \precsim \gamma_i$. We shall say that q creates a guess at $z_i + 1 = z$ iff for all $j \leq z_i$, j is neither 1-bad nor 2-bad at (i, q), and

(9) there are strings δ and ε such that

(9.1) $$P^i_{z_i}(l^i_{z_i}(g(i) + q + 1))(\delta) \precsim \beta_i$$

and

$$Q^i_{z_i}(l^i_{z_i}(g(i) + q + 1))(\varepsilon) \precsim \gamma_i;$$

(9.2) if $z = 2j + 1$, (δ, ε) is a q-combination for $2j$.

LEMMA 2. *There is a q which either changes or creates a guess.*

PROOF. Let $\hat{\jmath}$ = the least $j \leq z_i$ which lacks either property 1 or 2, if there is one; $\hat{\jmath} = z_i + 1$ otherwise. If $\hat{\jmath}$ lacks property 1, we find a q changing the primary guess at $\hat{\jmath}$; if $\hat{\jmath}$ has property 1 but not property 2, we find a q changing the secondary guess at $\hat{\jmath}$; if $\hat{\jmath} = z_i + 1$, we find a q creating a guess at $\hat{\jmath}$. Consider the first situation. Suppose that for $q \geq q_0$, our $(g(i) + q + 1)$st guess at $((f_{n^i_{2j}} \oplus (f)_{m^i_{2j}})^{(3)}$ is correct for all $2j \leq \hat{\jmath}$. So for $q \geq q_0$, all $k < \hat{\jmath}$ are neither 1-bad nor 2-bad at (i, q), and $\hat{\jmath}$ is 1-bad at (i, q). For $j < \hat{\jmath}$, $\lim_l l^i_j(l) = \omega$. If not, let j be the least counterexample; by remarks preceding the definition of l^i_j, $j = 2j' + 1$, $x^i_{j'} = -1$ and force$(j', \lim_l \langle P^{i*}_{2j}, Q^{i*}_{2j}\rangle(l))$ is partial; so Wit$(j', [n^i_{2j'}, m^i_{2j'}/\delta^i_{2j'}, \varepsilon^i_{2j'}])$ is defined, and j lacks property 1; contradiction with $j < \hat{\jmath}$. Now let $\hat{\jmath} = 2j + 1$. For sufficiently large q we may increase $l = l^i_{2j}(g(i) + q + 1)$ large enough to find (δ, ε), $P^i_{2j}(l)(\delta) \precsim \beta_i$ and $Q^i_{2j}(l)(\varepsilon) \precsim \gamma_i$, $\mathrm{lh}(\delta) = \mathrm{lh}(\varepsilon) = l$. Note that

$$[n^i_{2j}, m^i_{2j}/\delta, \varepsilon] = \lim_l \langle \mathrm{Full}(P^i_{2j}(l), \delta), \mathrm{Full}(Q^i_{2j}(l), \varepsilon)\rangle.$$

If Wit$(j, [n^i_{2j}, m^i_{2j}/\delta, \varepsilon])$ is defined, then for $q \geq q_0$ our $(g(i) + q + 1)$st guess at $((f)_{n^i_{2j}} \oplus (f)_{m^i_{2j}})^{(3)}$ says it is; so for sufficiently large $q \geq q_0$, it truthfully says that Wit$(j, [n^i_{2j}, m^i_{2j}/\delta, \varepsilon]) = \langle x, \sigma, \tau\rangle$ in $\leq g(i) + q + 1$ steps, and $\mathrm{lh}(\sigma) + \mathrm{lh}(\delta) \leq l^i_{2j}(g(i) + q + 1)$. On the other hand, if Wit$(j, [n^i_{2j}, m^i_{2j}/\delta, \varepsilon])$ is undefined, our $(g(i) + q + 1)$st guess at $((f)_{n^i_{2j}} \oplus (f)_{m^i_{2j}})^{(3)}$ says so. For sufficiently large q, force$(j, \langle P, Q\rangle)$ (0) is defined in $\leq g(i) + q + 1$ steps, for

$$\langle P, Q\rangle = \langle \mathrm{Full}(P^i_{2j}(l^i_{2j}(q(i) + q + 1)), \delta), \mathrm{Full}(Q^i_{2j}(l^i_{2j}(q(i) + q + 1)), \varepsilon)\rangle.$$

So a sufficiently large $q \geq q_0$ is as desired. Similar arguments apply in the other two situations. Q.E.D.

Notice that we can f-recursively decide whether q is as described in Lemma 2. We proceed as follows, recursively in f. Search for the least q as described in Lemma 2. Let $g(i + 1) = g(i) + q + 1$. If q changes the primary or secondary guess at j, let $j = z_{i+1} = z$. Otherwise let $z_{i+1} = z = z_i + 1$. Now we preserve some earlier guesses: for $j < z$, let

$$\langle P^{i+1}_j, Q^{i+1}_j \rangle = \langle P^i_j, Q^i_j\rangle, \quad \langle n^{i+1}_j, n^{i+1}_j\rangle = \langle n^i_j, m^i_j\rangle;$$

for $2j + 1 < z$, let

$$x^{i+1}_j = x^i_j, \quad \sigma^{i+1}_j = \sigma^i_j, \quad \tau^{i+1}_j = \tau^i_j;$$

for $j < z - 1$, let

$$\delta^{i+1}_j = \delta^i_j, \quad \varepsilon^{i+1}_j = \varepsilon^i_j.$$

The situation in which q changes the secondary guess at z is easiest to handle.

Here our guesses $\langle n_z^i, m_z^i\rangle$ have been found to be wrong relative for $\langle P_z^i, Q_z^i\rangle$. We let $\delta_{z-1}^{i+1} = \delta_{z-1}^i$, $\varepsilon_{z-1}^{i+1} = \varepsilon_{z-1}^i$, $\langle P_z^{i+1}, Q_z^{i+1}\rangle = \langle P_z^i, Q_z^i\rangle$ and, if $z = 2j + 1$, $x_j^{i+1} = x_j^i$, $\sigma_j^{i+1} = \sigma_j^i$, $\tau_j^{i+1} = \tau_j^i$. Select β and γ as in (8) and let $\beta_{i+1} = \beta$, $\gamma_{i+1} = \gamma$. Note that $l_z^i = l_z^{i+1}$. Now find the least $\langle n, m\rangle$ such that

$$\langle P_z^{i+1}, Q_z^{i+1}\rangle\,(l_z^{i+1}(g(i+1))) = \langle (f)_n \restriction \mathrm{Str}(l_z^{i+1}(g(i+1))), (f)_m \restriction \mathrm{Str}(l_z^{i+1}(g(i+1)))\rangle,$$

and let $\langle n_z^{i+1}, m_z^{i+1}\rangle =$ that $\langle n, m\rangle$.

Next easiest is the case in which q creates a new guess at $z = 2j + 2$. Select strings δ and ε as described in (9.1) to be δ_{z-1}^{i+1} and ε_{z-1}^{i+1}, respectively. Let $\beta_{i+1} = P_{z-1}^i(l_{z-1}^i(g(i+1)))(\delta)$ and $\gamma_{i+1} = Q_{z-1}^i(l_{z-1}^i(g(i+1)))(\varepsilon)$. So $\langle P_{z-1}^{i*}, Q_{z-1}^{i*}\rangle$ and $\langle P_z^i, Q_z^i\rangle$ are defined as described before the construction began. Now select $\langle n_z^{i+1}, m_z^{i+1}\rangle$ as in the previous case.

The cases in which q changes the primary guess at z and in which q creates a new condition at $z = 2j + 1$ are similar. Select δ and ε as described in (7) or in (9), and let $\delta_{z-1}^{i+1} = \delta$, $\varepsilon_{z-1}^{i+1} = \varepsilon$, $\beta_i = P_{z-1}^i(l_{z-1}^i(g(i+1)))(\delta)$, $\gamma_i = Q_{z-1}^i(l_{z-1}^i(g(i+1)))(\varepsilon)$. $\langle P_{z-1}^{i*}, Q_{z-1}^{i*}\rangle$ is now determined. If (δ, ε) is a q-combination by virtue of (6)(a), let $\langle x_j^{i+1}, \sigma_j^{i+1}, \tau_j^{i+1}\rangle =$ the $\langle x, \sigma, \tau\rangle$ described in (6)(a). If (δ, ε) is a q-combination by (6)(b), let $x_j^{i+1} = -1$, $\sigma_j^{i+1} = \tau_j^{i+1} = \langle\ \rangle$. Form $\langle P_z^i, Q_z^i\rangle$ as indicated in the preparatory remarks. We now select $\langle n_z^i, m_z^i\rangle$ as in the previous two cases.

Notice that $\langle \delta_{z-1}^{i+1}, \varepsilon_{z-1}^{i+1}\rangle$ is changed from $\langle \delta_{z-1}^i, \varepsilon_{z-1}^i\rangle$ only if we changed a primary guess; $\langle \delta_{z-1}^{i+1}, \varepsilon_{z-1}^{i+1}\rangle$ is defined while $\langle \delta_{z-1}^i, \varepsilon_{z-1}^i\rangle$ was undefined iff we created a new guess at z. It is easy to verify that $(1.i + 1)$, $(2.i + 1)$ and $(3.i + 1)$ are true. We now show that all our guesses settle down to sequences as described in (1), (2*), (3*), and (4*) and (5).

LEMMA 3. *There are sequences* $\{\langle P_j, Q_j\rangle\}_{j<\omega}$, $\{\langle \delta_j, \varepsilon_j\rangle\}_{j<\omega}$, $\{\langle x_j, \sigma_j, \tau_j\rangle\}_{j<\omega}$, $\langle\langle n_j, m_j\rangle\rangle_{j<\omega}$ *making* (1), (2*), (3*), (4*), *and* (5) *true*; *and for any k there is an* i_k *such that for all* $i \geq i_k$:

(10) *for* $j \leq k$, *j has properties* 1 *and* 2 *at i*; $k < z_i$;

(11) *for* $j \leq k$, $\langle n_j^i, m_j^i\rangle = \langle n_j, m_j\rangle$;

(12) *for* $j \leq k$, $\lim_l \langle P_j^i, Q_j^i\rangle\,(l) = \langle P_j, Q_j\rangle$;

(13) *for* $j < k$, $\langle \delta_j^i, \varepsilon_j^i\rangle = \langle \delta_j, \varepsilon_j\rangle$;

(14) *for* $2j + 1 \leq k$, $\langle x_j^i, \sigma_j^i, \tau_j^i\rangle = \langle x_j, \sigma_j, \tau_j\rangle$.

PROOF. The crucial fact here is that g is increasing. For $k = 0$, $i_k = 0$. Assume for k. Select $\hat{\imath} \geq i_k$ such that for all $q \geq g(\hat{\imath})$ and all $2j \leq k$, our qth guess at $((f)_{n_j} \oplus (f)_{m_j})^{(3)}$ is correct. For all $i \geq \hat{\imath}$, if k is even, $k + 1$ has property 1 at i, is not 1-bad at any (i, q'), and so is not selected for a primary change. We may let $\langle P_{k+1}, Q_{k+1}\rangle = \lim_l \langle P_{k+1}^i, Q_{k+1}^i\rangle\,(l)$, and let $\langle n_{k+1}, m_{k+1}\rangle$ be least $\langle n, m\rangle$ such that $\langle (f)_n, (f)_m\rangle = \langle P_{k+1}, Q_{k+1}\rangle$. For each $\langle n', m'\rangle < \langle n_{k+1}, m_{k+1}\rangle$ there is an $l_{\langle n', m'\rangle} = l$ such that

$$\langle (f)_{n'} \restriction \mathrm{Str}(l), (f)_{m'} \restriction \mathrm{Str}(l)\rangle \neq \langle P_{k+1} \restriction \mathrm{Str}(l), Q_{k+1} \restriction \mathrm{Str}(l)\rangle.$$

Let i_{k+1} be an $i \geq \hat{\imath}$ such that $l_{k+1}^i(g(i)) \geq l_{\langle n', m'\rangle}$ for all such $\langle n', m'\rangle$. For $i \geq i_{k+1}$, we have $\langle n_{k+1}^i, m_{k+1}^i\rangle = \langle n_{k+1}, m_{k+1}\rangle$. $k + 1$ has property 2 at such a stage i, so is not 1-bad at any (i, q'), and so is not selected for a secondary change. So

$k + 1 < z_i$. (13) and (14) are obviously true, letting $\delta_k = \delta_k^{i_k+1}$, $\varepsilon_k = \varepsilon_k^{i_k+1}$, and $x_j = x_j^{i_k+1}$, $\delta_j = \delta_j^{i_k+1}$, $\tau_j = \tau_j^{i_k+1}$ if $k + 1 = 2j + 1$. Q.E.D.

We finally must check that $B = \lim_i \beta_i$, $C = \lim_i \gamma_i$. For any j there is a least i at which either we create a new guess at j or make a primary change at j. For such an i, we have arranged that $P_j^i(l_j^i(g(i))) \precsim \beta_i$, $Q_j^i(l_j^i(g(i))) \precsim \gamma_i$. But for sufficiently large j, these $P_j^i(g(i)))$ and $Q_j^i(l_j^i(g(i)))$ may be made arbitrarily long. This insures the desired limits. Q.E.D.

COROLLARY. *Where I is a countable jump ideal and* $\boldsymbol{a}$ *is an u.u.b. on I then there is an I exact* $(\underline{b}, \underline{c})$ *with* $(\underline{b} \vee \underline{c}) < \underline{a}$.

PROOF. With $\underline{a}, \underline{b}, \underline{c}$ as above, if $\boldsymbol{b} \vee \boldsymbol{c} = \boldsymbol{a}$, $(\boldsymbol{b} \vee \boldsymbol{c})^{(2)} \leq \boldsymbol{a}^{(1)} = (\boldsymbol{b} \vee \boldsymbol{c})^{(1)}$, a contradiction. Thus $(\boldsymbol{b} \vee \boldsymbol{c}) < \boldsymbol{a}$.

The construction of Theorem 1 may be altered, using Sacks' technique for constructing minimal upper bounds, to insure that $\underline{b}$ and $\underline{c}$ are both minimal.

Recall that $\boldsymbol{a}$ is high over $\boldsymbol{b}$ iff $\boldsymbol{b} \leq \boldsymbol{a} \leq \boldsymbol{b}^{(1)} < \boldsymbol{b}^{(2)} \leq \boldsymbol{a}^{(1)}$. Can Theorem 1 be improved to: $\boldsymbol{a}$ is an u.u.b. on I iff $\boldsymbol{a}$ is high over the join of an I-exact pair? Perhaps. But we see no way to modify the previous construction to make $f \leq_T (B \oplus C)^{(1)}$. Furthermore, for all we know now Theorem 1 may be strengthened to: $\boldsymbol{a}$ is an u.u.b. on I iff for some I-exact $\{\boldsymbol{b}, \boldsymbol{c}\}$, $(\boldsymbol{b} \vee \boldsymbol{c})^{(1)} = \boldsymbol{a}$; this is equivalent to: if $\boldsymbol{a}$ is an u.u.b. on I, for some I-exact $\{\boldsymbol{b}, \boldsymbol{c}\}$, $(\boldsymbol{b} \vee \boldsymbol{c})^{(1)} \leq \boldsymbol{a}$.

We now characterize u.u.b.s in terms of weak u.u.b.s.

THEOREM 2. *For a countable jump ideal I,* $\boldsymbol{a}$ *is an u.u.b. on I iff for some* $\boldsymbol{b} \leq \boldsymbol{a}$, $\boldsymbol{b}$ *is a weak u.u.b. on I and* $\boldsymbol{b}^{(2)} \leq \boldsymbol{a}^{(1)}$.

PROOF ($\Leftarrow$). Let $B \in \boldsymbol{b}$ parametrize $\bigcup I \cap {}^{\omega}2$. Fix $A \in \boldsymbol{a}$. $X \subseteq \omega$ is total iff for every x there is a y such that $\langle x, y\rangle \in X$. Since $B^{(2)} \leq_T A^{(1)}$, we may guess recursively in A at whether $(B)_e$ is total and in the limit we are correct. Fix such a guessing procedure. Let $h(x, e, n) =$ the least y such that either $\langle x, y\rangle \in (B)_e$ or the $(n + y)$th guess is that $(B)_e$ is not total. Define f by:

$$(f)_{\langle e,n\rangle}(x) = \begin{cases} 0 & \text{if the } (n + h(x, e, n))\text{th guess} \\ & \text{is that } (B)_e \text{ is not total;} \\ h(x, e, n) & \text{otherwise.} \end{cases}$$

If $(B)_e$ is total, $(B)_e =^* \operatorname{graph}((f)_{\langle e,n\rangle})$; if $(B)_e$ is not total, $(f)_{\langle e,n\rangle} =^* \lambda x.0$. By Lemma 1, $\deg(f)$ is an u.u.b. on I. Since $f \leq_T A$, so is $\boldsymbol{a}$.

($\Rightarrow$) Let $f \in \boldsymbol{a}$ parametrize $\bigcup I$. Let $\langle \psi_j \rangle_{j<\omega}$ be a recursive enumeration of primitive recursive relations on ${}^{\omega}2 \times \omega \times \omega$. Introducing "$\underline{B}$" as an uninterpreted one place predicate constant, let φ_j be "$(\exists x)\neg(\exists y)\psi_j(\underline{B}, x, y)$." Let a condition be a finite sequence of members of $\bigcup I \cap {}^{\omega}2$. Where $\langle f_0, \ldots, f_{k-1}\rangle = K$ is a condition, let

$$K \Vdash \underline{B}(\underline{m}) \text{ iff } (m)_0 < k \text{ and } f_{(m)_0}((m)_1) = 1.$$

Other clauses in the definition of forcing run as usual. Note that

$$K \Vdash \neg \underline{B}(m) \text{ iff } (m)_0 < k \text{ and } f_{(m)_0}((m)_1) = 0.$$

Conditions may be coded as sequence numbers:

$$\langle n_0, \ldots, n_{k-1}\rangle \text{ codes } \langle \operatorname{sg}((f)_{n_0}), \ldots, \operatorname{sg}((f)_{n_{k-1}})\rangle,$$

where for any $x \in \omega$ and $h \in {}^{\omega}\omega$,

$$\mathrm{sg}(h)(x) = \begin{cases} 0 & \text{if } h(x) = 0, \\ 1 & \text{otherwise.} \end{cases}$$

We abuse terminology and call sequence numbers conditions.

For $X \subseteq \omega$, $X^{(<k)} = \{\langle x, y\rangle \in X \mid x < k\}$. For a condition $K = \langle f_0, \ldots, f_{k-1}\rangle$, $\hat{K} = f_0 \oplus \cdots \oplus f_{k-1}$. If B is generic and extends K, we shall have $B^{(<k)} = \hat{K}$. For $\sigma \in \mathrm{Str}$, σ is consistent with K iff for all $x < \mathrm{lh}(\sigma)$, if $(x)_0 < k$, $(\sigma)_x = f_{(x)_0}((x)_1)$; K includes σ iff σ is consistent with K and for all $x < \mathrm{lh}(\sigma)$, $(x)_0 < k$. All these definitions carry over to where K is a sequence number via the encoding previously described. From now on, conditions are sequence numbers.

The use of sg in this encoding leads to another abuse of terminology. For $K = \langle n_0, \ldots, n_{k-1}\rangle$, our qth guess at $X = ((f)_{n_0} \oplus \cdots \oplus (f)_{n_{k-1}})^{(2)}$ is $Y = (f)_{G(G(H(k, K, q), q), q)}$. Since $\hat{K}^{(2)}$ is clearly 1-reducible to X, we shall call Y our qth guess at $\hat{K}^{(2)}$.

LEMMA 4. *"$K \Vdash \varphi_j$" and "$K \Vdash \neg\varphi_j$" are Σ^0_2 and Π^0_2 in $\hat{K}$, respectively.*

PROOF. $K \Vdash \neg(\exists y)\ \psi_j(\underline{n}, y)$ iff for any $\sigma \in \mathrm{Str}$ and any y, if σ is consistent with K, "$\neg\psi_j(\underline{\sigma}, \underline{n}, \underline{y})$" is true. Thus "$K \Vdash \varphi_j$" is Σ^0_2 in $\hat{K}$. For $X \subseteq \omega$ and $\mathrm{lh}(K) = k$, let $\Phi(K, X, m) = \hat{K} \cup \{\langle x + k, y\rangle \mid \langle x, y\rangle \in X^{(<m)}\}$. Notice that K' extends K iff for some $X \in \bigcup I \cap {}^{\omega}2$ and some m, $\hat{K}' = \Phi(K, X, m)$. Using this fact we can show that $K \Vdash \neg\varphi_j$ iff for every $x, m \in \omega$ and $X \in \bigcup I \cap {}^{\omega}2$:

(†) there are $\sigma \in \mathrm{Str}$ and y such that σ is consistent with $\Phi(K, X, m)$ and $\psi_j(\sigma, x, y)$.

(†) has the form "$(\exists\sigma)(\exists y)P(\hat{K}, X, m, \sigma, x, y)$", with P recursive. So $K \Vdash \neg\varphi_j$ iff for all x and m:

(††) for all $X \in \bigcup I \cap {}^{\omega}2$, $(\exists\sigma)(\exists y)P(\hat{K}, X, m, \sigma, x, y)$.

(††) is equivalent to a Σ^0_1 in $\hat{K}$ formula by the Kreisel basis theorem and the fact that $\hat{K}^{(1)} \in \bigcup I$. Notice that here is where the difference between $\bigcup I$ and $\bigcup I \cap {}^{\omega}2$ appears. We now have "$K \Vdash \neg\varphi_j$" in a Π^0_2 in $\hat{K}$ form. Q.E.D.

Our goal is to construct sequences $\{K_j\}_{j<\omega}$, $\{x_j\}_{j<\omega}$ and $\{\beta_i\}_{i<\omega}$ such that:

(1) for all j, K_j is a condition and K_{j+1} extends K_j;

(2) for all j,

$$\text{if } x_j \geq 0,\ K_{2j+1} \Vdash \neg(\exists y)\psi_j(\underline{x}_j, y);$$
$$\text{if } x_j = -1,\ K_{2j+1} \Vdash \neg\varphi_j;$$

(3) for all j, $K_{2j+2} = K_{2j+1} \cap \langle j\rangle$;

(4) for all i and j, $\beta_i \in \mathrm{Str}$, $\beta_{i+1} \preccurlyeq \beta_i$ and β_i is consistent with K_j.

Notice that (2) implies $\lim_j \mathrm{lh}(K_j) = \omega$, which with (4) implies that $\lim_i \beta_i = \bigcup_j \hat{K}_j$.

Of course, such a construction cannot be carried out recursively in f. We resort to guessing at the sequences $\langle K_j\rangle_{j<\omega}$ and $\langle x_j\rangle_{j<\omega}$. At stage i we shall have z_i, for $j \leq 2z_i$ guesses K^i_j at K_j, and for $j < z_i$ guesses x^i_j at x_j. Revising previous terminology, let (K', x) be a j-witness for K iff K' extends K and forces "$\neg(\exists y)\ \psi_j(\underline{B}, \underline{x}, y)$". "$(K', x)$ is a j-witness for K" and "K has a j-witness" are Π^0_1 and Σ^0_2 in $\hat{K}$, respectively. Clearly if K'' extends K' and (K', x) is a j-witness for K, (K'', x) is also a j-witness for K. We shall say that (K, x) is consistent with a string β iff K is consistent with β. Notice that if K has no j-witness consistent with β, any condition extending K and including β forces φ_j. Fix an f-recursive function Incl such that: for β consistent with K, Incl(K, β) extends K and includes β. For example, where $\mathrm{lh}(K) = k$, and β is consistent with K, let

$$\mathrm{Incl}(K, \beta) = \begin{cases} K & \text{if } K \text{ includes } \beta, \\ K^\frown\langle n_k, \ldots, n_l\rangle & \text{otherwise,} \end{cases}$$

where for $k \le i \le l$, n_i is the least n such that for all $x < \mathrm{lh}(\beta)$ with $(x)_0 = i$, $(\beta)_x = \mathrm{sg}(((f)_{n_i})((x)_0)$. For $j < z_i$, we shall say that $2j + 1$ has property 1 at stage i iff:

if $x_j^i \ge 0$ then (K_{2j+1}^i, x_j^i) is a j-witness for K_{2j}^i;

if $x_j^i = -1$, then there is no j-witness for K_{2j}^i consistent with β_i.

We would like to have all $2j + 1$ with property 1 at stage i for $j < z_i$. But to keep our construction recursive in f, we cannot be so straightforward. Instead we insure that for all stages i:

(1.i) for all $j < z_i$ our $g(i)$th guess at $\widehat{(K_{2j}^i)}^{(2)}$ says that $2j + 1$ has property 1. Furthermore, we insure that for all stages i:

(2.i) if $z_i > 0$, β_i is included in $K_{2z_i-1}^i$. (This permits us to have $K_{2z_i}^i = K_{2z_i-1}^i{}^\frown\langle z_i\rangle$ without fear of destroying consistency with β_i.)

We now sketch the construction.

Stage 0. $z_0 = 0$, $K_0^0 = \langle\ \rangle$; $\beta_0 = \langle\ \rangle$, $g(0) = 0$. (1.0) and (2.0) are vacuously true.

Stage $i + 1$. Assume that z_i, $g(i)$, β_i, $\langle K_j^i\rangle_{j\le 2z_i}$ and $\langle x_j^i\rangle_{j<z_i}$ are defined with (1.i) and (2.i) true. For $j < z_i$, $2j + 1$ is bad at (i, q) iff our $(g(i) + q + 1)$st guess at $\widehat{(K_{2j}^i)}^{(2)}$ says that $2j + 1$ lacks property 1. Call β a q-combination for $2j$ at stage i, where $j \le z_i$, iff $\beta \preccurlyeq \beta_i$, $\beta \le g(i) + q + 1$, β is consistent with K_{2j}^i, and: if our $(g(i) + q + 1)$st guess at $\widehat{(K_{2j}^i)}^{(2)}$ says that K_{2j}^i has a j-witness consistent with β, it identifies one in $\le g(i) + q + 1$ steps. This property is decidable in f. We shall say that q changes the guess at $2j + 1$, for $j \le z_i$, iff for all $k < j$, $2k + 1$ is not bad at (i, q), $2j + 1$ is bad at (i, q), and there is a q-combination for $2j$. We shall say that q creates a guess at $2z_i + 1$ iff for all $k \le z_i$, $2k + 1$ is not bad at (i, q) and there is a q-combination for $2z_i$.

LEMMA 5. *There is a q such that for some $j \le z_i$, q either changes or creates a guess at* $2j + 1$.

PROOF. Fix $j^* =$ the least $j < z_i$ for which $2j + 1$ lacks property 1, if there is one; $j^* = z_i$ otherwise. Suppose that for all $q \ge q_0$, our $(g(i) + q + 1)$st guess at $\widehat{(K_{2k}^i)}^{(2)}$ for any $k \le j^*$ is correct. Thus for $q \ge q_0$ if $k < j$, $2k + 1$ is not bad at (i, q); if $j^* < z_i$, $2j^* + 1$ is bad at (i, q). Select a $\beta \preccurlyeq \beta_i$ which is consistent with $K_{2j^*}^i$. Thus for $k \le 2j^*$, β is consistent with K_k^i. If there is a j^*-witness for $K_{2j^*}^i$ consistent with β, let $q \ge q_0$ be large enough so that $\widehat{(K_{2j}^i)}^{(2)}$ identifies one in $\le g(i) + q + 1$ steps. β is a q-combination for $2j^*$. If $j^* < z_i$, q indicates a change at $2j^* + 1$; if $j^* = z_i$, q creates a guess at $2j^* + 1$. Q.E.D.

Notice that whether q is as described in Lemma 5 is decidable in f. So we may search, recursively in f, for the least such q. Let $g(i + 1) = g(i) + q + 1$; where j corresponds to q as required by Lemma 5, let $z_{i+1} = j + 1$. We abbreviate "z_{i+1}" as "z". Select β_{i+1} to be a q-combination for $2z - 2$. We preserve previous guesses as follows: $K_k^{i+1} = K_k^i$ for $k \leq 2z - 2$; $x_k^{i+1} = x_k^i$ for $k < z - 1$. We now define x_{z-1}^{i+1} and K_{2z-1}^{i+1}.

If our $g(i+1)$st guess at $\widehat{(K_{2z-2}^i)}^{(2)}$ says that K_{2z-2}^{i+1} has a $(z-1)$-witness consistent with β_{i+1}, it actually identifies some (K, x) as such a witness in $\leq g(i+1)$ steps. Select the least such $\langle K, x\rangle$ and let $x_{z-1}^{i+1} = x$, $K_{2z-1}^{i+1} = \mathrm{Incl}(K_{2z-2}^{i+1}, \beta_{i+1})$. Otherwise our guess says that K_{2z-2}^i has no $(z-1)$-witness consistent with β_{i+1}. Let $x_{z-1}^{i+1} = -1$ and $K_{2z-1}^{i+1} = \mathrm{Incl}(K_{2z-2}^{i+1}, \beta_{i+1})$. Notice that $(1.i+1)$ and $(2.i+1)$ are true. Let $K_{2z}^{i+1} = K_{2z-1}^{i+1}{}^\frown\langle z\rangle$. This construction settles down.

LEMMA 6. *There are sequences* $\{K_j\}_{j<\omega}$ *and* $\{x_j\}_{j<\omega}$, *with* $\{\beta_i\}_{i<\omega}$ *as just constructed, such that* (1)–(4) *are true*; *furthermore for any* k *there is an* i_k *such that for all* $i \geq i_i$:

(5) $z_i > k$;

(6) *for all* $j \leq 2k$, $K_j^i = K_j$;

(7) *for all* $j < k$, $x_j^i = x_j$.

The proof is very much like that of Lemma 3, except easier, so we omit it.

Letting $B = \bigcup_j \hat{K}_j$, B is a parametrization of $\bigcup I \cap {}^\omega 2$. Since $B = \lim_i \beta_i$, $B \leq_T f$. Since $f^{(1)}$ can tell us when our guesses at $\widehat{(K_{2j}^i)}^{(2)}$ are correct, $B^{(2)} \leq_T f^{(1)}$. Q.E.D.

We do not know whether this theorem may be improved to: $\boldsymbol{a}$ is an u.u.b. on I iff for some weak u.u.b. $\boldsymbol{b}$ on I; $\boldsymbol{a} = \boldsymbol{b}^{(1)}$.

Combining this construction with the exact-pair construction we may obtain $\underline{b}$ and $\underline{c}$ in Theorem 1 which are both weak u.u.b.s on I.

. Clearly the $\boldsymbol{b}$ constructed in Theorem 2 (⇒) is strictly below $\boldsymbol{a}$. This observation is strengthened by the following.

THEOREM 3. *For a countable jump ideal* I, $\{\boldsymbol{a} \mid \boldsymbol{a}$ *is an u.u.b. on* $I\}$ *has no minimal member.*

PROOF. Let $f \in \boldsymbol{a}$ parametrize $\bigcup I$. We construct $h <_T f$, h parametrizing $\bigcup I$. Let $\langle \psi_j \rangle_{j<\omega}$ be as in the previous proof; we introduce an uninterpreted binary predicate letter "$\underline{H}$" intended to denote the graph of a generic function. Let a condition be a sequence $K = \langle f_0, \ldots, f_{k-1}\rangle$ of members of $\bigcup I$. Let

$$K \Vdash \underline{H}(\underline{n}, \underline{m}) \text{ iff } (n)_0 < k \text{ and } f_{(n)_0}((n)_1) = m.$$

The other clauses in the definition of forcing are as usual. Again we note that

$$K \Vdash \neg \underline{H}(\underline{n}, \underline{m}) \text{ iff } (n)_0 < k \text{ and } f_{(n)_0}((n)_1) \neq m.$$

Let $\hat{K}$ be the partial function with domain $\omega^{(<k)}$ such that $\hat{K}(\langle i, x\rangle) = f_i(x)$. Since $\hat{K}$ is partial, $\hat{K}^{(1)}$ is undefined; therefore we shall abuse notation and write "$\hat{K}^{(1)}$" for "$(f_0 \oplus \cdots \oplus f_{k-1})^{(1)}$".

Notice that Lemma 1 provides a fixed f-recursive way of guessing at an f-index for that set, uniformly in a code for K. A finite function shall be one from a member of ω into ω. A finite function h is consistent with K iff for all $x \in \mathrm{dom}(h)$ with $(x)_0 < k$, $\hat{K}(x) = h(x)$; K includes h iff $\mathrm{dom}(h) \subseteq \omega^{(<k)}$ and h is consistent with K. R_j is the requirement $\{j\}^H \neq f$. K meets R_j with x in t steps iff for some y, $K \Vdash$ "$\{\underline{j}\}^{\underline{H}}(\underline{x})$ converges to $\underline{y}$ in $\underline{t}$ steps" and $f(x) \neq y$. Where h is a partial function, we understand a computation in graph(h) to halt as soon as the oracle for graph(h) is asked: "Is $\langle x, y\rangle \in$ graph(h)?" for $x \notin \mathrm{dom}(h)$. With this understanding, observe

that K has an extension meeting R_j with x in t steps iff there is a finite function consistent with K and a $y \neq f(x)$ such that $\{j\}^{\text{graph}(h)}(x)$ converges to y in t steps; we may search for such an h recursively in $\hat{K}$, since finite functions code as sequence numbers.

Let sequence numbers encode conditions by $\langle n_0, \ldots, n_{k-1}\rangle \mapsto \langle (f)_{n_0}, \ldots, (f)_{n_{k-1}}\rangle$. So we freely abuse our terminology and treat sequence numbers as conditions.

Fix an f-recursive function Incl such that for a finite h consistent with K, Incl(K, h) extends K and includes h. (For example, vary the corresponding definition in the previous proof.)

Let (K', x) be a j-witness for K iff K' extends K and meets R_j with $(x)_0$ in $\leq (x)_1$, steps. Call h consistent with (K, x) iff consistent with K. Suppose K has no j-witness consistent with a finite function h, K' extends K and includes h. Then for some x, $K' \Vdash$ "$\{\underline{j}\}^{\underline{H}}(\underline{x})$ is undefined." Suppose not. We may define f by $f(x) = y$ iff

(*) some finite function h' is consistent with K' and $\{j\}^{\text{graph}(h')}(x) = y$.

Here is why. By our assumption, for any x, K' has an extension K'' forcing "$\{j\}^{\underline{H}}(x)$ is defined." Since K'' includes h, (K'', x) is not a j-witness for K. So if $K'' \Vdash$ "$\{\underline{j}\}^{\underline{H}}(\underline{x}) = \underline{y}$", $y = f(x)$. The existence of such a K'' is equivalent with (*). We would like to define sequences $\{\underline{K}_j\}_{j<\omega}$, $\{x_j\}_{j<\omega}$ and $\{h_i\}_{i<\omega}$ such that:

(1) for each j, K_j is a condition;

(2) for each j,

if $x_j \geq 0$, (K_{2j+1}, x_j) is a j-witness for K_{2j}; if $x_j = -1$, K_{2j+1}, $\Vdash$ "$\{\underline{j}\}^{\underline{H}}(x)$ is undefined" for some x;

(3) for each j, $K_{2j+2} = K_{2j+1}{}^\frown\langle j\rangle$;

(4) for each i and j, h_i is a finite function, h_{i+1} properly extends h_i, and h_i is consistent with K_j.

(3) implies that $h = \lim_j \hat{K}_j$ is total;

(4) implies that $h = \lim_i h_i$. By (3), h parametrizes $\bigcup I$. By (2) $f \not\leq_T h$.

To make this construction recursive in f, we resort to guessing. At stage i, we shall have z_i, h_i, $g(i)$, for $j \leq 2z_i$ a guess K_j^i at K_j, and for $j \leq z_i$ a guess x_j^i at x_j. We make sure that at each stage i:

(1.i) for $j < z_i$, if $x_j^i \geq 0$, (K_{2j+1}^i, x_j^i) is a j-witness for K_{2j}^i;

(2.i) for $j < z_i$, if $x_j^i = -1$, our $g(i)$th guess at $\widehat{(K_{2j}^i)}^{(1)}$ says

($*, i, j$) for some $x \leq g(i)$ for all finite h consistent with K_{2j}^i and h_i, $\{j\}^{\text{graph}(h)}(x)$ is undefined.

(3.i) $K_{2z_i-1}^i$ includes h_i.

We now describe the construction.

Stage 0. $z_0 = 0$, $h_0 =$ the null function, $K_0^0 = \langle\ \rangle$, $g(0) = 0$.

Stage $i + 1$. Suppose we have t_i, h_i, $g(i)$, $\langle K_j^i\rangle_{j\leq 2z_i}$, $\langle x_j^i\rangle_{j\leq z_i}$, with (1.$i$)–(3.$i$) true. For $j < z_i$, $2j + 1$ is bad at (i, q) iff $x_j^i = -1$ and our $(g(i) + q + 1)$st guess at $\widehat{(K_{2j}^i)}^{(1)}$ says that ($*, i, j$) is false. For a finite function h, (h, x) is a q-combination for $2j$ at i iff h properly extends h_i, $\langle h, x\rangle \leq g(i) + q + 1$, and $\{j\}^{\text{graph}(h)}((x)_0)$ is defined in $(x)_1$ steps and has value $\neq f((x)_0)$.

We shall say that q changes the guess for $2j + 1$ at stage i iff: for all $k < j$, $2k + 1$ is not bad at (i, q), $2j + 1$ is, and there is a q-combination for $2j$. We shall say that q creates a guess for $2z_i = 1$ iff: for all $k < z_i$, $2k + 1$ is not bad at (i, q), and either there is a q-combination for $2z_i$ or else $q = 0$ and our $(g(i) + 1)$st guess at $\widehat{(K^i_{2j})}^{(1)}$ says that $(*, i, z_i)$ is true.

LEMMA 7. *Some q either changes or creates a guess.*

Proof is very much like that of Lemma 5.

Whether q changes or creates a guess is decidable in f. So recursively in f we search for the least such q. Let $g(i + 1) = g(i) + q + 1$. If q changes or creates a guess at $2j + 1$, let $j + 1 = z_{i+1}$. Letting $z = z_{i+1}$, we preserve earlier guesses:

$$\text{for } j \leq 2z - 2,\ K^{i+1}_j = K_j; \quad \text{for } j < z - 1,\ x^{i+1}_j = x^i_j.$$

If there is a q-combination for $2z - 2$, let (h_{i+1}, x^{i+1}_{z-1}) be the least such. Otherwise let $x^{i+1}_{z-1} = -1$ and $h_{i+1} = h_i \cup \{\langle \mathrm{dom}(h_i), 0\rangle\}$. Let $K^{i+1}_{2z-1} = \mathrm{Incl}(K^{i+1}_{2z-2}, h_{i+1})$. Notice that $(1.i + 1)$–$(3.i + 1)$ are true. Now let $K^{i+1}_{2z} = K^{i+1}_{2z-1}{}^\frown\langle z\rangle$.

LEMMA 8. *With $\langle h_i\rangle_{i<\omega}$ as just constructed, there are sequences $\langle K_j\rangle_{j<\omega}$ and $\langle x_j\rangle_{j<\omega}$ of which* (1)–(4) *are true; furthermore for each k there is an i_k such that for all $i \geq i_k$:*

(5) *for $j \leq 2k$, $K_j = K^i_j$;*

(6) *for $j < k$, $x_j = x^i_j$.*

The proof of this lemma should now be routine. Because this entire construction is recursive in f, and $h = \lim_i h_i$, $h \leq_T f$. So by preliminary remarks, we are done. Q.E.D.

Where I is a countable jump ideal $\boldsymbol{a}$ is a nice u.u.b. on I iff $\boldsymbol{a}$ is the degree of a nice parametrization of $\bigcup I$; a parametrization f of $\bigcup I$ is nice iff for some $G \leq_T f$, $H \leq_T f$, for all x and y: $(f)_{G(x)} = (f)^{(1)}_x$; $(f)_{H(x,y)} = (f)_x \oplus (f)_y$. This notion is introduced in [1]; in [2] it is shown that $\boldsymbol{a}$ is a nice u.u.b. on I iff for some u.u.b. $\boldsymbol{b}$ on I, $\boldsymbol{a} = \boldsymbol{b}^{(1)}$. In [2] the following notions are defined. I is a hierarchy ideal iff for some $A \subseteq \omega$ and some α, $\bigcup I = L_\alpha[A] \cap {}^\omega\omega$. I is a case 1 hierarchy ideal iff for some $B \in L_\alpha[A]$, $\alpha < \omega^B_1$ and $\bigcup I = L_\alpha[A] \cap {}^\omega\omega$; I is a case 2 hierarchy ideal iff for some $B \in L_\alpha[A]$, $\alpha = \omega^B_1$ and $\bigcup I = L_\alpha[A] \cup {}^\omega\omega$; I is a case 3 hierarchy ideal if it is a hierarchy ideal not falling under cases 1 or 2. Any case 1 hierarchy ideal has a least nice u.u.b.; for example, if $\bigcup I = \{f \mid f \text{ is arithmetic}\}$, that nice u.u.b. is $\mathbf{0}^{(\omega)}$. In [2] it is asked whether any case 2 or case 3 hierarchy ideals have a minimal nice u.u.b. The technique of Theorem 3 may be modified to provide a negative answer.

THEOREM 4. *For I a case* 2 *or case* 3 *hierarchy ideal, $\{\boldsymbol{a} \mid \boldsymbol{a}$ is a nice u.u.b. on $I\}$ has no minimal member.*

PROOF. Let $f \in \boldsymbol{a}$ be a nice parametrization of $\bigcup I$. It suffices to construct a parametrization h of $\bigcup I$ with $h^{(1)} <_T f$. Let conditions and forcing be as in the previous proofs except that "$\underline{H}$" is monadic, and:

$$K \Vdash \underline{H}(\underline{x}) \text{ iff for } = \langle n, m\rangle,\ (n)_0 < k \text{ and for } K = \langle f_0, \ldots f_{k-1}\rangle,\ f_{(n)_0}((n)_1) = m.$$

This way "$x \in \underline{H}^{(1)}$" makes sense. Let R_j be the requirement $\{j\}^{H^{(1)}} \neq f$. K meets R_j with x iff for some $y \neq f(x)$, $K \Vdash$ "$\{\underline{j}\}^{\underline{H}^{(1)}}(\underline{x}) = \underline{y}$." Because f is nice, whether

$K \Vdash$ "$\{\underline{j}\}^{\underline{H}^{(1)}}(\underline{x}) = \underline{y}$" is decidable in f. Let (K', x) be a j-witness for K iff K' extends K and meets R_j with x.

LEMMA 9. *Suppose K is consistent with a finite function h. If there is no j-witness for K consistent with h, and K' extends K and includes h, then for some x, $K' \Vdash$ "$\{\underline{j}\}^{\underline{H}^{(1)}}(x)$ is undefined."*

PROOF. If not, we may define f by $f(x) = y$ iff some extension of K' forces "$\{\underline{j}\}^{\underline{H}^{(1)}}(\underline{x}) = \underline{y}$". "$\langle f_0, \ldots, f_{k-1}\rangle \Vdash$ "$\{\underline{j}\}^{\underline{H}^{(1)}}(x) = y$" is Σ^0_2 in $f_0 \oplus \cdots \oplus f_{k-1}$. So f is Σ^1_1 over $\bigcup I$ with $\operatorname{graph}\widehat{(K')}$ as a parameter. Since f is a function, f is even Δ^1_1 over $\bigcup I$ in that parameter.

By familiar facts about hyperarithmeticity, in case 2, $f \leq_{\mathrm{HYP}} \operatorname{graph}\widehat{(K')}$; in case 3, f is recursive in the hyperjump of $\operatorname{graph}\widehat{(K')}$ which belongs to $\bigcup I$. Either way, $f \in \bigcup I$, contradiction. Q.E.D.

The construction of h is much like that used for Theorem 3, with "$\{\underline{j}\}^{\underline{H}^{(1)}}$" replacing "$\{\underline{j}\}^{\underline{H}}$". But $(2.i)$ must be changed to: if $j < z_i$, if $x^i_j = -1$ then there is no j-witness for K^i_{2j} which is consistent with h_i and $\leq g(i)$.

The notion of being bad at (i, q) is correspondingly changed. (We are forcing Σ^0_2 and Π^0_2 sentences; so $K^{i(1)}_{2j}$ cannot tell us how to select K^i_{2j+1}. Since f is nice, "K has a j-witness consistent with h_i" is Σ^0_1 in f; thus guessing at $\widehat{K^{i(1)}_{2j}}$ is replaced by a search recursive in f.) The rest is routine. Q.E.D.

In conclusion, we note that weak u.u.b.s remain shrouded in mystery. For example: are any weak u.u.b.s also minimal u.b.s? The technique of Theorem 3 does not yield a negative answer, for it cannot construct objects recursive in weak u.u.b.s which are not also u.u.b.s. It essentially involves guessing at jumps as described in the guessing lemma; thus by the remark immediately following the proof of the guessing lemma, the previous claim follows. Hopefully the techniques involved in answering questions like the one just posed will suggest a degree-theoretic definition of a weak u.u.b. in some way analogous to that of Theorem 1.

BIBLIOGRAPHY

[1] G. HENSEL and H. PUTNAM, *On the notational independence of various hierarchies of degrees of unsolvability*, this JOURNAL, vol. 30 (1965), pp. 64–86.

[2] H. HODES, *Uniform upper bounds on ideals of Turing degrees*, this JOURNAL, vol. 43 (1978), pp. 601–612.

[3] C. JOCKUSCH and S. SIMPSON, *A degree-theoretic characterization of the ramified analytical hierarchy*, ***Annals of Mathematical Logic***, vol. 10 (1976), pp. 1–32.

DEPARTMENT OF PHILOSOPHY
CORNELL UNIVERSITY
ITHACA, NEW YORK 14853