PROCEEDINGS OF THE
AMERICAN MATHEMATICAL SOCIETY
Volume 123, Number 12, December 1995

EVERY LOW $_2$ BOOLEAN ALGEBRA HAS A RECURSIVE COPY

JOHN J. THURBER

(Communicated by Andreas Blass)

ABSTRACT. The degree of a structure $\mathscr{A}$ is the Turing degree of its open diagram $D(\mathscr{A})$, coded as a subset of ω. Implicit in the definition is a particular presentation of the structure; the degree is not an isomorphism invariant. We prove that if a Boolean algebra $\mathscr{A}$ has a copy of low $_2$ degree, then there is a recursive Boolean algebra $\mathscr{B}$ which is isomorphic to $\mathscr{A}$. This builds on work of Downey and Jockusch, who proved the analogous result starting with a low $_1$ Boolean algebra.

0. INTRODUCTION

Each structure considered here has a recursive universe. For a countable structure $\mathscr{A}$ in a recursive language, the *degree* of the structure is the Turing degree of its open diagram, $D(\mathscr{A})$, under some effective coding as a subset of ω. Implicit in the definition is a particular presentation of the structure under consideration; the degree is not isomorphically invariant. In this paper we consider a specific case of the general question: For a structure $\mathscr{A}$ of degree **d**, when does $\mathscr{A}$ necessarily have a recursive copy? For an excellent discussion of general background in this area, and related questions, see [D-J]. We will here discuss some of the specific background for our question.

We will use the abbreviations BA for Boolean algebra and r.e. for recusively enumerable. A degree **d** is *low*$_n$ if $\mathbf{d}^{(n)} = \mathbf{0}^{(n)}$, where $\mathbf{d}^{(n)}$ is the degree of the n-th jump of a set D of degree **d**. A degree is *low* if it is low $_1$. A degree **d** is *high*$_n$ if $\mathbf{d}^{(n)} = \mathbf{0}^{(n+1)}$. Note that we do not restrict ourselves to degrees $\leq \mathbf{0}'$, as is common in the definition of low $_n$ and high $_n$. For a linear ordering $\mathscr{L}$, Intalg($\mathscr{L}$) denotes the BA generated by left-closed, right-open intervals in $\mathscr{L}$. For a BA $\mathscr{A}$, At($\mathscr{A}$) denotes the set of atoms of $\mathscr{A}$. Recursion-theoretic notation is standard, as in [R], and the general reference on BAs is [M-B].

Feiner, in [F], constructed an r.e. BA with no recursive copy. His quite complicated construction had the stronger property that the r.e. BA constructed was not isomorphic to any low $_n$ BA for any n. Was this a necessary condition to avoid a recursive copy? Or, in other words, if $\mathscr{A}$ is a BA of low $_n$ degree, does $\mathscr{A}$ necessarily have a recursive copy? This question was answered affirmatively for $n = 1$ by Downey and Jockusch in [D-J]. This paper provides an affirmative answer for $n = 2$. The question remains open for $n > 2$.

Received by the editors August 20, 1993 and, in revised form, June 6, 1994.
1991 *Mathematics Subject Classification*. Primary 03C57; Secondary 03D80.

With respect to high$_n$ degrees, we note that it follows from the extensions of Feiner's work in [T2] that for any high$_n$ degree $\mathbf{d}$, there is a BA of degree $\mathbf{d}$ with no recursive copy.

1. Main result

Theorem 1. *Let $\mathscr{A}$ be a BA such that $D(\mathscr{A})$ is of low$_2$ degree. Then $\mathscr{A}$ has a recursive copy.*

This theorem extends the following result.

Theorem 2 (Downey–Jockusch, [D-J]). *Let $\mathscr{A}$ be a BA such that $D(\mathscr{A})$ is of low degree. Then $\mathscr{A}$ has a recursive copy.*

The constructions of Theorems 1 and 2 are done via linear orderings, based on the following:

Fact. Given a BA $\mathscr{A}$ of degree $\mathbf{d}$, there is a linear ordering $\mathscr{L}$ of degree $\mathbf{d}$ such that $\mathscr{A}$ is isomorphic to $\mathrm{Intalg}(\mathscr{L})$. This fact is based on the effectivity of the argument that every countable BA is isomorphic to an interval algebra (see [M-B]). It's also easy to see that for an ordering $\mathscr{L}$ of degree $\mathbf{d}$, $\mathrm{Intalg}(\mathscr{L})$ is of degree $\mathbf{d}$.

For an ordering $\mathscr{L}$, $a, b \in \mathscr{L}$ and $a < b$, (a, b) is an *adjacency* of $\mathscr{L}$ if there is no $c \in \mathscr{L}$ such that $a < c < b$. We write $adj(a, b)$ if (a, b) is an adjacency of $\mathscr{L}$. Note that for $\mathscr{A}$ isomorphic to $\mathrm{Intalg}(\mathscr{L})$, atoms of $\mathscr{A}$ correspond to adjacencies of $\mathscr{L}$.

In passing to the framework of linear orderings, Downey and Jockusch actually proved the following result (see the proof of Theorem 1, [D-J]).

Theorem 3 (Downey–Jockusch). *Let $\mathscr{L}$ be a linear ordering with $D(\mathscr{L}) \in \Delta_2^0$ and $adj_{\mathscr{L}}$ a Δ_2^0 predicate. There is a recursive linear ordering $\hat{\mathscr{L}}$ such that* $\mathrm{Intalg}(\mathscr{L})$ *is isomorphic to* $\mathrm{Intalg}(\hat{\mathscr{L}})$.

For any low structure, its existential diagram is recursive in $\varnothing'$. Thus a low linear ordering $\mathscr{L}$ satisfies the hypothesis of Theorem 3, and it is clear that Theorem 3 implies Theorem 2.

To prove Theorem 1, we will also pass to the framework of linear orderings and work with a low$_2$ linear ordering $\mathscr{L}_1$ such that $\mathrm{Intalg}(\mathscr{L}_1)$ is isomorphic to our low$_2$ BA $\mathscr{A}$. Making use of a more extensive table of information available at the Δ_3^0 level, we will be able to construct a Δ_2^0 ordering $\mathscr{L}_2$ with a Δ_2^0 adjacency relation such that $\mathrm{Intalg}(\mathscr{L}_2)$ is isomorphic to $\mathrm{Intalg}(\mathscr{L}_1)$. We then apply Theorem 3 to $\mathscr{L}_2$ to recover the desired recursive BA.

With respect to constructing an ordering which satisfies the hypotheses of Theorem 3, note that if $\mathscr{L}$ is a Δ_2^0 ordering, then to insure that $adj_{\mathscr{L}}$ is also Δ_2^0, it will be enough to enumerate the $adj_{\mathscr{L}}$ relation at the Δ_2^0 level. This is easy to see since $\neg adj$ is clearly Σ_2^0 if the ordering is Δ_2^0.

The constructions in Theorems 1 and 2 make use of the following result of Remmel ([Re], Theorem 1.2).

Theorem 4 (Remmel). *Let $\mathscr{B}, \mathscr{C}$ be countable BAs, such that $\mathscr{B}$ is a subalgebra of $\mathscr{C}$, $\mathscr{B}$ has infinitely many atoms, each atom of $\mathscr{B}$ is the join of finitely many atoms of $\mathscr{C}$, and for each $c \in At(\mathscr{C})$ there is a $b \in At(\mathscr{B})$*

with $c \leq b$. Suppose further that $\mathscr{C}$ is generated by $\mathscr{B} \cup At(\mathscr{C})$. Then $\mathscr{B}$ is isomorphic to $\mathscr{C}$.

For any BA $\mathscr{A}$, if $\mathscr{B}$ is the BA generated by $\mathscr{A}$ and the addition of n atoms, $n < \omega$, then $\mathscr{B}$ is naturally isomorphic to the product algebra $\mathscr{A} \times 2^n$ (2^n denotes the finite algebra generated by n atoms). It follows from a theorem of Vaught ([M-B], Proposition 6.6) that if $\mathscr{A}$ is countable with infinitely many atoms, then $\mathscr{A}$ is isomorphic to $\mathscr{B}$. Thus we may strengthen Theorem 4 by replacing the condition "for each $c \in At(\mathscr{C})$..." with the weaker condition that all but finitely many atoms of $\mathscr{C}$ lie below atoms of $\mathscr{B}$. We will assume the strengthened version.

Theorem 4 allows us a certain freedom in putting new elements into a given linear ordering without changing the isomorphism type of the corresponding Boolean algebra. For our purposes, it is safe to assume that our Boolean algebras have infinitely many atoms; any countable algebra with only finitely many atoms is either finite or can be expressed as the disjoint union of a finite element and an atomless element, and hence is isomorphic to Intalg($n + \eta$) for some n, with η denoting the order type of the rationals. Clearly, any finite algebra has a recursive copy, and the existence of a recursive copy in the latter case follows from the existence of a recursive copy of η.

In view of Theorem 3 and its subsequent comments, it is now clear that to prove Theorem 1, it is sufficient to prove Theorem 5 below.

Theorem 5. *Let $\mathscr{L}$ be a low$_2$ ordering. There exists a Δ^0_2 ordering $\hat{\mathscr{L}}$ such that $\hat{adj}$ is a Σ^0_2 predicate, and* Intalg($\mathscr{L}$) *is isomorphic to* Intalg($\hat{\mathscr{L}}$).

Proof. If $\mathscr{L}$ is low$_2$, then the Σ_2 diagram of $\mathscr{L}$ is Δ^0_3. Thus the following relations on $\mathscr{L}$ are Δ^0_3:

(1) $x < y$.
(2) $adj(x, y) := [x < y \;\&\; \forall z(x \leq z \leq y \Rightarrow z = x \text{ or } z = y)]$.
(3) $P^-(x) := \forall y < x(\exists z(y < z < x))$, or, "*$x$ is a left-hand limit point*", $P^+(x) := \forall y > x(\exists z(y > z > x))$, or, "*$x$ is a right-hand limit point*".
(4) $dn(x, y) := \forall u, v\,(x \leq u < v \leq y \Rightarrow \neg adj(u, v))$, or "*$[x, y]$ is dense*".

We will make use of these to construct $\hat{\mathscr{L}}$ and $h: \mathscr{L} \to \hat{\mathscr{L}}$ satisfying the hypotheses of the following lemma. The lemma is an easy consequence of the strengthened version of Theorem 4 referred to above. The construction will be a finite injury construction at the Δ^0_2 level.

Lemma 6. *Let $\mathscr{L}$ and $\hat{\mathscr{L}}$ be orderings such that $\mathscr{L}$ has infinitely many adjacencies, and $\mathscr{L}$ has a first element and a last element. Suppose there is a function $h: \mathscr{L} \to \hat{\mathscr{L}}$ with the following properties:*

(i) *h is one-to-one and order preserving.*
(ii) *If (a, b) is an adjacency in $\mathscr{L}$, then $(h(a), h(b))$ is finite in $\hat{\mathscr{L}}$.*
(iii) *If a is the first element of $\mathscr{L}$, then $\{x \in \hat{\mathscr{L}} : x \leq h(a)\}$ is finite. If b is the last element of $\mathscr{L}$, then $\{x \in \hat{\mathscr{L}} : x \geq h(b)\}$ is finite.*
(iv) *If $c \in \hat{\mathscr{L}} - rng(h)$, then there exist $a, b \in \mathscr{L}$ such that exactly one of the following holds:*
(1) *$h(a) < c < h(b)$ and $adj(a, b)$ holds in $\mathscr{L}$,*

(2) $c \leq h(a)$ *and* a *is the first element of* $\mathscr{L}$,
(3) $h(b) \leq c$ *and* b *is the last element of* $\mathscr{L}$.

Then Intalg($\mathscr{L}$) *is isomorphic to* Intalg($\hat{\mathscr{L}}$).

For our given ordering, as $D(\mathscr{L})'' \in \Delta^0_3$, we can find a Δ^0_3 function g such that $g(k)$ yields the open diagram of the structure $\mathscr{L}$ augmented with symbols for the predicates (1)–(4) and restricted to the first $k+1$ elements of $\mathscr{L}$. Thus $g(k)$ is a (Gödel number for a) finite set of sentences. By the Limit Lemma there is a Δ^0_2 approximating function L, such that $\lim_s L(k, s) = g(k)$. We can assume that for fixed s and $k < s$, $L(k, s) \subseteq L(k+1, s)$. We let $L_s = L(s, s)$.

Without loss of generality we can make two more assumptions about L. First, we assume L_s is consistent in the sense that there is a structure $\mathscr{A}_s = (A, <, adj, dn, P^+, P^-)$ such that $<$ is an ordering, the additional predicates are as defined earlier, and $\mathscr{A}_s \models L_s$. Second, we assume that $L(m, s)$ gives the complete atomic type of $a_0, a_1, \ldots, a_m$ in the expanded language.

We will use subscripts to denote the various approximations. Corresponding relations on $\hat{\mathscr{L}}$ will be denoted by $\hat{adj}$, $\hat{P}^+$, $\hat{P}^-$, and $\hat{dn}$.

Let $\mathscr{L} = \{a_0, a_1, \ldots\}$, and let $\mathscr{B} = \{b_0, b_1, \ldots\}$ be a new set of constants from which we will build $\hat{\mathscr{L}}$. Assume that a_0, a_1 are the first and last elements of $\mathscr{L}$ respectively. Let $\mathscr{L}_s$ denote the set $\{a_0, a_1, \ldots, a_s\}$. We will construct, at stage s, recursively in L_t for some $t \geq s$, a finite ordering $\hat{\mathscr{L}}_s$, and $h_s : \mathscr{L}_t \to \hat{\mathscr{L}}_s$. We will also enumerate part of the relation $\hat{adj}$.

At stage 0, let $h(a_0) = b_0$, $h(a_1) = b_1$. We will only change the map on these elements under the following conditions—we will consider a_0; the argument for a_1 is similar. Note that we can assume $\neg adj_s(a_0, a_1)$ holds for all s and $\neg dn_s(a_0, a_1)$ holds for all s. Suppose for some s, L_s indicates that $adj(a_0, a_i)$ holds for some i. We may enumerate $\hat{adj}(h(a_0), h(a_i))$, and then for $t > s$, have L_t indicate $P^+(a_0)$, that is, that a_0 has no successor. There will be a string $h(a_0) = x_1 < x_2 < \cdots < x_k$, which is maximal with respect to the property that $\hat{adj}(x_i, x_{i+1})$ holds for $i < k$. We would now set $h(a_0) = x_k$, cancelling $h(a_j)$ for all j such that $h(a_j) \leq x_k$. If, for some $q > t$, we find some m such that $adj_q(a_0, a_m)$ holds, we can simply add a successor. Such changes will happen only finitely often, so that (iii) of the lemma will hold.

To continue the construction we will have two types of requirements which we will identify respectively with elements $a_m \in \mathscr{L}$ and $b_k \in \mathscr{B}$, prioritized in decreasing order according to the list $a_0, b_0, a_1, b_1, \ldots$.

Let us first establish the following conventions. For $u, v \in \hat{\mathscr{L}}$, we write $u \prec v$ if $u < v$ and for all $c \in \hat{\mathscr{L}}$, $c \leq u$ or $v \leq c$. We say u, v are *attached* if there are $x_1 < x_2 < \cdots < x_k \in \hat{\mathscr{L}}$ such that $u = x_1$, $v = x_k$ and $\hat{adj}(x_i, x_{i+1})$ holds for $i < k$; b_k is *correctly attached* if there is some $a_i \in L$ such that b_k is attached to $h(a_i)$ and a_i never again receives attention. Elements of $\hat{\mathscr{L}}$ which satisfy condition (iv) of the lemma are said to be *bound.*

The requirements are

- a_m: $h(a_m)$ is defined.
- b_k: $b_k \in \hat{\mathscr{L}} - rng(h) \Rightarrow \exists a \in \mathscr{L}$ such that b_k is attached to $h(a)$.

We will also insist that at each stage our definition of h preserves the following conditions, with respect to the approximation L_t we are following at that stage:

- If $a_i <_t a_j$ holds, then $h(a_i) < h(a_j)$.
- If $adj_t(a_i, a_j)$ holds, then $(h(a_i), h(a_j))$ receives no new elements.
- If $h(a_i) < h(a_j)$ and these elements are attached, then $[a_i, a_j]$ must appear to be a finite chain of adjacencies of length at most the length of $[h(a_i), h(a_j)]$.
- If $P_t^+(a_i)$ holds, then there must not exist any enumerated adjacency of the form $(h(a_i), b_k)$.
- If $P_t^-(a_i)$ holds, then there must not exist any enumerated adjacency of the form $(b_k, h(a_i))$.
- If $dn_t(a_i, a_j)$ holds, there may not be any enumerated adjacencies in the interval $[h(a_i), h(a_j)]$.

Comment. The first two conditions to be preserved clearly relate to assuring that h and $\hat{\mathscr{L}}$ satisfy (i) and (ii) of the lemma. The remaining four are concerned with satisfying (iv) of the lemma. Since we will also be enumerating $\hat{adj}$, preservation of limit points and density under h will be necessary to achieve (iv) as we will have enumerated adjacencies in $\hat{\mathscr{L}} - rng(h)$ to account for. With regard to the third condition, we note that the predicate "$\|[a_i, a_j]\| = k$" is Δ_3^0 because the Σ_2 diagram of $\mathscr{L}$ is Δ_3^0, and thus a Δ_2^0 approximation exists. We could, therefore, formally introduce a recursive sequence of such predicates, one for each k, if desired. Satisfaction of requirement b_k does not leave b_k bound, however if we can show that b_k will be correctly attached to $h(a_i)$ for some i, then b_k will be bound when the true predecessor or successor of a_i appears.

Define $t_s = t$ if we are following L_t at stage s.

Requirement a_m *requires attention* at stage s if $h(a_m)$ is not defined.

Requirement b_k *requires attention* at stage s if b_k is not equal to or attached to any element currently in the range of h.

When we define $h_s(a_i) = b_k$, this is done with the priority of the requirement which needed attention.

Stage 0. Let $t_0 = 0$. As described earlier, we let $h(a_0) = b_0$, $h(a_1) = b_1$.

Stage $s, s > 0$. Let $t = t_s = t_{s-1} + 1$. Cancel all mappings $h(a_m)$ such that the definition is inconsistent with L_t according to our list of conditions to be preserved. If two defined values jointly create an inconsistency, cancel the one of lower priority. These inconsistencies arise from changes in the Δ_2^0 approximations to $<, dn, P^+$, and P^-. Now choose c with highest priority which requires attention. We follow the appropriate strategy as described below.

Strategy for $c = a_m$. As we described the strategy for $m = 0$ and $m = 1$ in our earlier discussion of stage 0, we assume $m > 1$. Cancel all mappings defined with lower priority than a_m. Thus m is the least number such that $h(a_m)$ is undefined. By virtue of our choice of a_m, there are i, j such that $a_i <_t a_m <_t a_j$, $h(a_i)$ and $h(a_j)$ are defined with higher priority than a_m, and for any c with $a_i < c < a_j$, $h(c)$ is not defined. Also, there are no values $h(a_n)$ defined with higher priority such that $h(a_i) < h(a_n) < h(a_j)$, as our cancellation makes h appear to be order-preserving.

By hypothesis, $\neg adj_t(a_i, a_j)$ holds. First, suppose that $h(a_i)$ is attached to $h(a_j)$. By the choice of m and the third condition above, there exists some x, $h(a_i) < x < h(a_j)$, such that we can set $h(a_m) = x$ and preserve the third condition. Choose such an x and define $h(a_m) = x$.

If $h(a_i)$ is not attached to $h(a_j)$, we proceed as follows. If there are no unbound elements in $(h(a_i), h(a_j))$, let b_k be the first unused constant, set $h(a_m) = b_k$, and $h(a_i) < b_k < h(a_j)$. If $adj_t(a_i, a_m)$ holds, set $\hat{adj}(h(a_i), b_k)$; if $adj_t(a_m, a_j)$ holds, set $\hat{adj}(b_k, h(a_j))$.

If there are unbound elements in $(h(a_i), h(a_j))$, we will try to bind them as follows, considering three cases.

Case I. Either $adj_t(a_i, a_m)$ or $adj_t(a_m, a_j)$ holds; assume the former, and if $\neg adj_t(a_i, a_m)$ holds, we can apply a similar argument to (a_m, a_j). Let $x_1 < x_2 < \cdots < x_k$ be those unbound elements in $(h(a_i), h(a_j))$ with x_k the greatest element which is not attached to $h(a_j)$. Set $h(a_m) = x_k$, and $\hat{adj}(h(a_i), x_1)$, $\hat{adj}(x_i, x_{i+1})$ for $i < k$. If $adj_t(a_m, a_j)$ holds, set $\hat{adj}(u, v)$ for all u, v such that $x_k \leq u \prec v \leq h(a_j)$. Note that these elements will now be bound if this is the last time a_m receives attention.

Suppose that neither $adj_t(a_i, a_m)$ nor $adj_t(a_m, a_j)$ holds. We have one of the next two cases depending on whether or not we think a_m is a limit point.

Case II. Either $\neg P_t^-(a_m)$ or $\neg P_t^+(a_m)$ holds; we'll consider the former and act in a symmetric way if $P_t^-(a_m)$ holds. We now believe that a_m has an immediate predecessor; this offers us a place to attach elements. If, for each $c \in (h(a_i), h(a_j))$, c is attached to either $h(a_i)$ or $h(a_j)$, we choose a new constant b_k, and set $h(a_m) = b_k$ and $u \prec b_k$ for greatest u which is attached to $h(a_i)$. Otherwise let $x_1 < x_2 < \cdots < x_k$ be the unbound elements in $(h(a_i), h(a_j))$ with x_1 the least which is not attached to $h(a_i)$ and x_k the greatest which is not attached to $h(a_j)$. Set $h(a_m) = x_k$ and $\hat{adj}(x_i, x_{i+1})$ for $i < k$. Thus there are no unattached constants in $(h(a_i), h(a_j))$.

Case III. Suppose $P_t^-(a_m)$ and $P_t^+(a_m)$ both hold. We cannot attach unbound constants now, but if there are enumerated adjacencies in $(h(a_i), h(a_j))$ which are not attached to either $h(a_i)$ or $h(a_j)$, they must be put into the appropriate subinterval in order to preserve the condition regarding density. If there are no such enumerated adjacencies in $(h(a_i), h(a_j))$ to worry about, let k be least such that $h(a_i) < b_k < h(a_j)$ and b_k is unattached, or such that b_k is the first unused constant, and set $h(a_m) = b_k$. In the latter case, choose x and y in $[h(a_i), h(a_j)]$ with x the greatest element which is attached to $h(a_i)$ and y the least element which is attached to $h(a_j)$. Set $x < b_k < y$.

Otherwise, suppose there are u, v in $(h(a_i), h(a_j))$ such that $\hat{adj}(u, v)$ holds and u, v are not attached to either $h(a_i)$ or $h(a_j)$. Then $\neg dn_t(a_i, a_j)$ holds, hence either $\neg dn_t(a_i, a_m)$ holds or $\neg dn_t(a_m, a_j)$ holds; as before we will assume the former. Let x be the greatest element in $(h(a_i), h(a_j))$ which is not attached to $h(a_j)$. If there is no $c < x$ such that $\hat{adj}(c, x)$ holds, set $h(a_m) = x$; otherwise let b_k be the first unused constant, and set $h(a_m) = b_k$, $x \prec b_k$.

Strategy for $c = b_k$. We are assuming that b_k is not attached to anything in the range of h. Cancel all values of h defined with lower priority than b_k. There are i, j such that $h(a_i) < b_k < h(a_j)$, $h(a_i)$ and $h(a_j)$ are defined with higher priority than b_k, and $(h(a_i), h(a_j))$ contains nothing else defined with higher priority than b_k. Such i and j exist because of our stage 0 definition of h and the limited conditions under which that mapping could change.

If $\neg P_t^+(a_i)$ holds, we attach b_k to $h(a_i)$ by setting $\hat{adj}(u, v)$ for all $u, v \in \hat{\mathscr{L}}$ such that $h(a_i) \leq u \prec v \leq b_k$. If $P_t^+(a_i)$ holds and $\neg P_t^-(a_j)$ holds, act symmetrically.

Suppose $P_t^+(a_i)$ and $P_t^-(a_j)$ both hold. If there are no enumerated adjacencies in $(h(a_i), h(a_j))$, then we search in L_r, increasing in $r \geq t$, for the first m such that $a_i <_r a_m <_r a_j$. Either the search will halt and we reset $t_s = r$, and $h(a_m) = b_k$, or for some r we will discover new inconsistencies between L_r and our currently defined map. In the latter case we proceed to stage $s+1$, letting $t_{s+1} = r$.

If there are enumerated adjacencies in $(h(a_i), h(a_j))$, then we can assume that $\neg dn_t(a_i, a_j)$ holds. Let x and y be the least and greatest elements respectively of $(h(a_i), h(a_j)) - rng(h)$. Conduct a search in L_r, $r \geq t$, for the first m such that $a_i <_r a_m <_r a_j$ and $\neg P_r^+(a_m)$ or $\neg P_r^-(a_m)$ holds. The existence of the enumerated adjacencies implies that either the search will halt and we will set $h(a_m) = x$ or $h(a_m) = y$ as appropriate, also setting $\hat{adj}(u, v)$ for all u and v such that $x \leq u \prec v \leq y$, or for some r we will discover new inconsistencies and start the next stage as above.

This describes the construction. It remains to argue that all requirements are eventually satisfied. From our discussion of stage 0, it's easy to see that this is true for a_0, b_0, a_1 and b_1. Action for these elements also makes it clear that at a given stage s, we will eventually settle on a highest priority requirement for attention.

By induction, suppose that for $i < k$, a_i and b_i have received attention for the last time. The finite nature of our Δ_2^0 approximations makes it clear that a_k will require attention only finitely often, and will be satisfied after the last such time.

Considering b_k, assume that for $i \leq k$, $j < k$, a_i and b_j have received attention for the last time. Reconsidering the strategy for b_k, action taken for b_k will involve a search for a suitable preimage either for b_k, or for some c to which b_k is attached. From the hypothesis that higher priority requirements will never again require attention, it follows that a suitable preimage a_m exists. If a_m is the first such for which the approximations settle down, we will have b_k correctly attached to $h(a_m)$, as this mapping is established with priority b_k. Clearly, being correctly attached implies that a_m has a predecessor or a successor, and b_k will be bound when that element last receives attention.

Finally, we claim that the enumerated adjacencies of $\hat{\mathscr{L}}$ give all the adjacencies of $\hat{\mathscr{L}}$. Suppose (u, v) is an adjacency of $\hat{\mathscr{L}}$. Clearly, if u and v are both in $rng(h)$, then (u, v) is the image of an adjacency of $\mathscr{L}$ and so will be enumerated. Suppose exactly one of u and v, say $u = h(a_i)$, is in $rng(h)$. If $\neg P^+(a_i)$ holds, then $\hat{adj}(u, v)$ will be enumerated when we last act for a_i and its successor. If $P^+(a_i)$ holds, then when v receives attention it will be attached to $h(a_k)$ for some k with $a_i < a_k$, and (u, v) cannot be an actual adjacency. Suppose (u, v) is a true adjacency and neither u nor v is in $rng(h)$. Let a_i, a_j be such that $a_i < a_j$, u is correctly attached to $h(a_i)$ and v is correctly attached to $h(a_j)$. If (u, v) is a true adjacency, then $(h(a_i), h(a_j))$ is finite, thus (a_i, a_j) must be finite, and $\hat{adj}(u, v)$ will be enumerated when the intermediate adjacencies of (a_i, a_j) are recognized and attended to.

Let $h = \lim_s h_s$, and $\hat{\mathscr{L}} = \bigcup \hat{\mathscr{L}}_s$. These satisfy the hypotheses of Lemma 6.

This completes the proof of Theorem 5 and also of Theorem 1.

Acknowledgments

This paper forms part of the author's Ph.D. dissertation [T1] prepared at the University of Notre Dame under the direction of Julia Knight. Carl Jockusch also made numerous helpful suggestions.

References

[D-J] R. Downey and C. G. Jockusch, *Every low Boolean algebra is isomorphic to a recursive one*, Proc. Amer. Math. Soc. **122** (1994), 871–880.

[F] L. Feiner, *Hierarchies of Boolean algebras*, J. Symbolic Logic **35** (1974), 365–374.

[M-B] J. D. Monk and R. Bonnet (eds.), *Handbook of Boolean algebras*, Vol. 1, Elsevier Science Publishers B.V., Amsterdam, 1989.

[R] H. Rogers, *Theory of recursive functions and effective computability*, MIT Press, Cambridge, MA, 1987.

[Re] J. B. Remmel, *Recursive isomorphism types of recursive Boolean algebras*, J. Symbolic Logic **46** (1981), 572–594.

[T1] J. Thurber, *Degrees of Boolean algebras*, Ph.D. Dissertation, University of Notre Dame, 1994.

[T2] ——, *Recursive and r.e. quotient Boolean algebras*, Arch. Math. Logic **33** (1994), 121–129.

Department of Mathematics, University of Notre Dame, Notre Dame, Indiana 46556

Current address: Department of Mathematics, Loso Hall, Eastern Oregon State College, La Grande, Oregon 97850

E-mail address: `jthurber@eosc.osshe.edu`