Jumps of nontrivial splittings of r.e. sets

MICHAEL A. INGRASSIA, STEFFEN LEMPP

Abstract. In an infinite injury construction, we construct a nonrecursive recursively enumerable (r.e.) set \( A \) such that whenever \( A \) is split into nonrecursive r.e. sets \( A_0 \) and \( A_1 \) then \( A'_0, A'_1 < T A' \).

1. The theorem. A pair of recursively enumerable (r.e.) sets \( A_0 \) and \( A_1 \) is said to split an r.e. set \( A \) if \( A = A_0 \cup A_1 \) (i.e., \( A = A_0 \cup A_1 \) and \( \emptyset = A_0 \cap A_1 \)). Friedberg [5] was the first to prove that any nonrecursive r.e. set can be split into two nonrecursive r.e. sets. Sacks [15] improved this result by showing that the two halves can be made of low incomparable degrees. Other well-known splitting results were obtained by Owings [13], R. W. Robinson [14], Morley and Soare [12], and Lachlan [8].

Lerman and Remmel introduced the universal splitting property (USP) of an r.e. set \( A \), namely, that any r.e. degree \( d \leq \text{deg}(A) \) is realized as the degree of a splitting half of \( A \). They showed [10, 11] that both the degrees containing USP sets and the degrees not containing any USP set are downward dense in the partial order \( R \) of the r.e. degrees. Downey [3] exhibited a non-USP set in every nonrecursive r.e. degree. The so-called strong universal splitting property (in which the degrees of both splitting halves can be prescribed) was introduced and studied by Ambos-Spies and Fejer [2].

In a different direction, call an r.e. set \( A \) mitotic if \( A \) can be split into r.e. sets of the same degree. Lachlan [7] proved the existence of nonmitotic r.e. sets, and Ingrassia [6] improved this result by showing that their degrees are dense in \( R \). Ambos-Spies [1], and independently Downey and L. Welch [4], constructed antimitotic sets (r.e. sets such that the degrees of any splitting into nonrecursive r.e. sets form a minimal pair).

Ambos-Spies [1] also initiated the study of jumps of splittings of r.e. sets by building an r.e. set \( A \) such that for any splitting into r.e. sets \( A_0 \) and \( A_1 \), not both \( A_0 \) and \( A_1 \) have the same jump as \( A \) (a property he called strong nonmitoticity). We strengthen this result and answer a question of Remmel (see Downey and L. Welch [4]) as follows:

**Theorem.** There is a nonrecursive r.e. set \( A \) such that whenever \( A \) is split into two nonrecursive r.e. sets \( A_0 \) and \( A_1 \) then \( A'_0, A'_1 < T A' \).

The proof uses a new technique for handling jumps of r.e. sets, developed by Lempp and Slaman [9] in their solution to the deep degree problem.

Our notation follows Soare [16].

2. The requirements and the strategies. We will build an r.e. set \( A \) satisfying the following requirements (for all \( e, i, j \)):

\[
\mathcal{R}_e : A \neq \{e\}, \\
\mathcal{S}_{i,j} : A = W_i \sqcup W_j \implies A' \not\lesssim_T W'_i \text{ or } W'_j \lesssim_T \emptyset.
\]

The second author was partially supported by NSF grant DMS-8701891.
For each $S_{i,j}$ we construct a partial recursive functional $\Gamma_{i,j}$. Using the Limit Lemma, we will then ensure $S_{i,j}$ by satisfying for all $k$ the requirements

$$S_{i,j,k} : A = W_i \sqcup W_j \implies \lim_s \Gamma_{i,j}^A((-), s) \neq \lim_v \Phi_k^W((-), v) \text{ or } W_j \leq T \emptyset$$

(There is also a hidden requirement that if $A = W_i \sqcup W_j$ and $W_j > T \emptyset$ then $\Gamma_{i,j}^A$ is total and $\lim_s \Gamma_{i,j}^A(x, s)$ exists for all $x$.)

Each $R_e$-strategy acts at most once, so an $S_{i,j,k}$-strategy need not be concerned about the (finite) injury by higher-priority $R_e$-strategies. A typical $S_{i,j,k}$-strategy $\alpha$ will first try to show $W_j$ recursive via some recursive functional $\Delta$, which requires (potentially) infinite $A$-restraint (to prevent $W_j$ from changing). It will deal with the (necessary) infinite injury by lower-priority $R_e$-strategies as follows: Whenever a lower-priority $R_e$-strategy wants to put some number $z$ into $A$, then $\alpha$ will first start setting $\Gamma_{i,j}^A(x, s) = 1$ with use $\gamma_{i,j}(x, s) = z$ for larger and larger $s$ (where $z$ is the argument at which $\alpha$ is trying to achieve $\lim_s \Gamma_{i,j}^A(x, s) \neq \lim_v \Phi_k^W(x, v)$) and search for a (new) $v$ such that $\Phi_k^W(x, v) = 1$. If and when it finds that $v$, then $z$ is allowed into $A$, enabling us to reset $\Gamma_{i,j}^A(x, s) = 0$. If only finitely many of these $z$ enter $W_j$ then the injury to $\Delta$ is finite, and therefore $W_j$ is recursive. On the other hand, if infinitely many of these $z$ enter $W_j$ then these $z$ will not enter $W_i$, and if we can protect infinitely many of the corresponding computations $\Phi_k^W(x, v) = 1$ from later injury to $W_i$ then the limits of $\Gamma_{i,j}^A(x, -)$ and $\Phi_k^W(x, -)$ will be different (if the latter exists at all).

Protection of these $\Phi_k^W$-computations of one $S_{i,j,k}$-strategy $\alpha$ from injury by infinitely many $R_e$-strategies can be ensured by “rearranging the priorities” of the $R_e$-strategies, using a noneffective function $b$ and letting an $R_{b(n)}$-strategy $\beta$ have higher priority than any $b' \in C(n)$, the set of $R_e$-strategies with $b(n - 1) < e < b(n)$. Now when $b(n - 1)$ has been determined permanently, then $b(n)$ will be the index of the (next) $R_e$-strategy whose $z$ enters $W_j$, at stage $t(n)$, say, and therefore no $R_e$-strategy can injure the $\Phi_k^W$-computation of $\alpha = a(n)$ since they stopped acting for $e \leq b(n - 1)$ by hypothesis, or have to respect the restraint for $e > b(n - 1)$ by the rearrangement of priorities. (In the construction below, we will actually rearrange the $R_e$-strategies in the tree priority ordering rather than the linear ordering outlined above. To ensure that every $S_{i,j,k}$-strategy has infinitely many chances to rearrange the priorities of the $R_e$-strategies, we will define a function $P$, rearranging the priorities of the $S_{i,j,k}$-strategies for this purpose.)

Notice finally that above we have suppressed the two finite outcomes of an $S_{i,j,k}$-strategy, namely, $A \neq W_i \sqcup W_j$, and that the search for a new $v$ such that $\Phi_k^W(x, v) = 1$ is unsuccessful in which case $\lim_s \Gamma_{i,j}^A(x, s) = 1$ but not $\lim_s \Phi_k^W(x, v) = 1$.

3. The construction. The construction is organized on a tree $T = 2^{<\omega}$ of strategies. Strategy $\gamma \in T$ works on requirement $S_{i,j,k}$ if $|\gamma| = 2(i, j, k)$ is even, and on requirement $R_e$ if $|\gamma| = 2e + 1$ is odd.

Diagrams 1 and 2 show the flow charts for $S_{i,j,k}$- and $R_e$-strategies. A strategy, upon initialization, starts in state $\text{init}$, picking a witness $x$ or $z$ bigger than any number mentioned in the construction so far, and, whenever eligible to act, proceeds along the arrows to the next state (denoted by a circle). Along the way, it executes the instructions (in rectangular boxes) and makes decisions (in diamonds or hexagons). Through outside action, it may
pick new \( x \), set \( v_0 = 0 \), set \( \Delta_\alpha = \lambda n[1] \),
set \( s_0 = \) the current substage

\[ \ell > \ell_{s_0} \]

set \( \Delta_\alpha(|\text{dom} \, \Delta_\alpha|) = W_j(|\text{dom} \, \Delta_\alpha|) \),
set \( s_0 = \) the current substage.

request to start setting \( \Gamma_{i,j}^A(x,-) = 1 \)
with use \( \gamma_{i,j}(x,-) = z \)

\( \exists v > v_0 \)

(\( \Phi_k^{W_i}(x,v) \downarrow = 1 \))

set \( v_0 = \) least such \( v \)

request to stop setting \( \Gamma_{i,j}^A(x,-) = 1 \)

\( z \in W_i \cup W_j \)

Diagram 1: \( S_{i,j,b} \)-strategy \( \alpha \)
Diagram 2: $R_e$-strategy $\beta$

1. Initialize: Pick a new $z$
2. Wait for $n$: Check if $\{e\}(z) \models 0 \land \forall w \in A(\neg \{e\}(w) \models 0)$
3. Put $\alpha_n$ into $req_z$
4. Wait for $\alpha_n$: Check if $\alpha_n$ is ready
5. Put $\alpha_{n-1}$ into $req_z$
6. Wait for $\alpha_{n-1}$: Check if $\alpha_{n-1}$ is ready
7. Put $\alpha_1$ into $req_z$
8. Wait for $\alpha_1$: Check if $\alpha_1$ is ready
9. Win: Put $z$ into $A$, put all $\alpha_m$ into reset
be put into special states (in half-circles) from which it proceeds immediately to the next state. All parameters are taken at the current substage unless sub-indexed by a previous substage. (We will assume from now on that a substage also codes the corresponding stage.)

For diagram 1, the parameters $x, s_0, v_0$, and $\Delta_\alpha$ are defined in the diagram and roughly denote the witness at which $\alpha$ tries to achieve $\lim, r; (', s) \neq \lim v_i IJ :: V' (', v)$, the last $(A = W_i \sqcup W_j)$-expansory stage, the last "opponent's stage" at which $W_i(u) + W_j(u)$, and the partial recursive function trying to witness the recursiveness of $W_i$, respectively.

The parameter $r$ is the length of agreement $r = \max \{ y \mid \forall u < y (A(u) = W_i(u)) \}$. The partial recursive functional $\Gamma_i,j$ is global to the construction and shared by all $S_{i,j,k}$-strategies for this pair $(i, j)$. An $S_{i,j,k}$-strategy can only issue requests for $\Gamma_i,j$, which will be observed at the end of a stage as described below.

For Diagram 2, the parameter $z$ is defined in the diagram and denotes the witness at which $f_3$ is trying to achieve $A, \{ e \}$. The strategies $a_1, \ldots, a_n$ mentioned in the diagram are exactly the $S_{i,j,k}$-strategies $a_m$ with $a_m(0) \subseteq \beta$ in increasing order of length.

We are now ready to describe the full construction.

At stage 0 of the construction, all strategies are initialized in order of increasing length, the functions $a, b, C,$ and $t$ are completely undefined, and we set $P(\gamma) = |\gamma|$ for all $\gamma \in T$.

A stage $s + 1$ consists of three steps:

First, pick the highest-priority $R_c$-strategy $\beta$ that is in some state $\text{wait}_{c,m-1}$ and that can proceed to state $\text{wait}_{c,m-1}$ or win. If $\beta$ exists let it act. If $\beta$ also reaches win and $\beta \in C(n_0)$ for some $n_0$, then initialize all $\gamma > \beta$, make the functions $a, b, C,$ and $t$ undefined for arguments $n > n_0$, and set $P(\gamma) = P_t(n_0) (\gamma)$ for all $\gamma \in T$.

Secondly, we proceed in substages $t \leq s$. At a substage $t < s$, a strategy $\gamma$ of length $t$ is eligible to act according to its flow chart.

If $\gamma$ is an $S_{i,j,k}$-strategy and has changed states from $\text{wait}_d$ to $\text{wait}_W$ at this substage while its $z \in W_j$ then let $n_0$ be the greatest $n$ such that $a(n)$ is defined and $P_t(n_0) (a(n)) \leq P(\gamma)$. (Allow $n_0 = -1$ here.) Then (re)define

\[ a(n_0 + 1) = \gamma, \]
\[ b(n_0 + 1) = \text{the } R_c\text{-strategy that put } z \text{ into } A, \]
\[ C(n_0 + 1) = \beta \in T - \bigcup_{n \leq n_0} C(n) \mid |\beta| \leq s \text{ odd}, \]
\[ t(n_0 + 1) = \text{the current substage}. \]

Make the functions $a, b, C,$ and $t$ undefined for arguments $n > n_0 + 1$.

Increment $P(\gamma)$ by $+1$ and set $P(\alpha) = P_t(n_0) (\alpha)$ for all $S_{i,j,k}$-strategies $\alpha \neq \gamma$. Initialize all $\beta \in C(n_0 + 1)$.

The strategy eligible to act at the next substage is $\gamma(0)$ if $\gamma$ is an $S_{i,j,k}$-strategy and has extended the definition of $\Delta_\gamma$ at the current substage, otherwise $\gamma(1)$.

At the end of the second step of stage $s + 1$, we initialize all $\gamma' > \gamma$ where $\gamma$ is the strategy that acted at substage $s$.

In the third and final step of stage $s + 1$, we (re)define $\Gamma_i,j(x, u)$ for all $i, j, x$ and all $u \leq s$ if it is now undefined. If some $S_{i,j,k}$-strategy $\alpha$ currently works with witness
x and requests to start setting $\Gamma_{i,j}^A(x,\cdot) = 1$, i.e. is currently in state $\text{wait}\Phi$ or ready, then set $\Gamma_{i,j}^A(x, u) = 1$ with requested use $\gamma_{i,j}(x, u) = z$; otherwise set $\Gamma_{i,j}^A(x, u) = 0$ with $\gamma_{i,j}(x, u) = 0$.

4. The verification. We first need to show that the rearrangement of priorities works properly:

**Lemma 1 (Rearrangement of Priorities Lemma).**

(i) The limit functions $a = \lim_s a_s$, $b = \lim_s b_s$, $c = \lim_s c_s$, and $t = \lim_s t_s$ are well-defined and total.

(ii) If for an $S_{i,j,k}$-strategy $\alpha$ there are infinitely many $n$ and $s$ such that $a_s(n) = \alpha$ then there are infinitely many $n$ such that $a(n) = \alpha$. (Thus $P(\alpha) = \lim_n P_s(a(n))$ exists for all $\alpha$.)

**Proof:** i) Since $W_i = 0$, $W_j = A$ for some $i, j$, and since $A$ is infinite, we will set $a_s(n) = \alpha$ infinitely often (for some $\alpha$). Observe that all strategies working on a fixed requirement $R_e$ combined put at most finitely many numbers into $A$. Since $P_s(a_s(n))$ is nondecreasing in $n$ (at all $s$), and since $P_{s-1}(a_{s-1}(n)) \uparrow > P_s(a_s(n))$ when we define $a_s(n)$, part (i) follows by induction on $p = P_s(a_s(n))$ and on $n$.

(ii) We will first show that for any $p$ we define $a(n)$ with $P_s(a(n)) \leq p$ only finitely often. We proceed by induction on $p$ and assume the statement for $P_s(a(n)) < p$. (Allow $p = 0$ here.) Suppose the statement is false for $P_s(a(n)) \leq p$. Since $P_s(\alpha) \geq |\alpha|$, it suffices to show that there are not $n_1 < n_2$ such that $a(n_1) = a(n_2) = \alpha$ and $P_{s-1}(a(n_1)) = P_s(a(n_2)) = p$. For the sake of a contradiction, assume there is such an $\alpha$. It is impossible that some $\beta \in C_2(n)$ decreased $P(\alpha)$ between $t(n_1)$ and $t(n_2)$ since this would have caused a redefinition of $a(n_1)$. So some $S_{i,j,k}$-strategy $\alpha'$ must have decreased $P(\alpha)$ to $p - 1$ between $t(n_1)$ and $t(n_2)$, say, at some (least) substage $s'$. Then $P_{s'}(\alpha) = P_{s'}(a(n_0))$ for some $n_0$ with $n_1 < n_0 < n_2$, and $t_{s'}(n_0) \geq t(n_1)$, so $P_{s'}(a(n_0)) \geq P_{s'}(a(n_1)) = p$, a contradiction.

For part (ii), we now just observe that, by the above, each $\alpha$ will eventually either satisfy $P_s(\alpha) \geq p$ for all $p$, or else eventually not want to set $a(n) = \alpha$ for any $n$.

We now define the true path $f$ of the construction as the leftmost path on $T$ on which any strategy is eligible to act infinitely often.

**Lemma 2 (Initialization Lemma).** Any $\gamma \subset f$ is initialized at most finitely often.

**Proof:** By induction on $|\gamma|$, let $s'$ be the least substage $> |\gamma|$ after which $\gamma^-$ is no longer initialized. (Set $s' = 0$ for $\gamma = \emptyset$.) If $\gamma$ is an $R_e$-strategy, we define $n_s(\gamma)$ to be the unique $n$ such that $\gamma \in C_s(n)$ at $s (> |\gamma|)$, observe that $n_s(\gamma)$ is nonincreasing in $s$, and set $n(\gamma) = \lim_s n_s(\gamma)$. Then we assume furthermore that $C(n(\gamma))$ has been defined permanently before $s'$.

Now, by our assumptions on $s'$, the construction can initialize $\gamma$ at a substage $s$ after $s'$ only if $\gamma = \gamma^- \langle 1 \rangle$ and $\gamma^- \langle 0 \rangle$ is eligible to act at $s$, or if $\gamma$ puts its $z$ into $A$. By the definition of the true path or by the construction, respectively, this will happen at most finitely often.
We are now in a position to prove the two main lemmas that establish the theorem:

**Lemma 3 (Convergence Lemma).** For all $i$ and $j$:

(i) $\Gamma^A_{i,j}$ is total, and

(ii) $\lim_{s} \Gamma^A_{i,j}(x,s)$ exists for all $x$.

**Proof:** Since $\gamma_{i,j}(x,s)$ increases at most once for fixed $x$ and $s$, (i) follows by the third step of each stage of the construction.

Again by the third step, part (ii) is trivial if eventually no $S_{i,j,k}$-strategy works on $x$. Otherwise, some fixed $S_{i,j,k}$-strategy will eventually always work on $x$. But then $\Gamma^A_{i,j}(x,s)$ is set or reset to 0 eventually for all $s$ unless $\alpha$ is eventually always in state wait or ready in which case $\Gamma^A_{i,j}(x,s) = 1$ for almost all $s$.

**Lemma 4 (Outcome Lemma).** Each $\gamma \subset f$ satisfies its requirement.

**Proof:** By Lemma 2, let $s'$ be the least stage such that $-y$ is not initialized after stage $s'$. First assume that $-y$ is an $R_{i,j,k}$-strategy. Since $-y \subset f$, $-y$ must eventually be in state wait or in state win. In either case, $R_{e}$ is satisfied.

On the other hand, assume that $\gamma$ is an $S_{i,j,k}$-strategy and that $A = W_i \sqcup W_j$. Suppose first that $\gamma^A(1) \subset f$. Then, since $\gamma \subset f$ and $A = W_i \sqcup W_j$, $\alpha$ must eventually always be in state wait or in state win. But then $\lim_{s} \Gamma^A_{i,j}(x,s) = 1$ and not $\lim_{s} \Phi^w_k(x,s) = 1$.

Finally, assume $\gamma^A(0) \subset f$. Then $\Delta_{\gamma}$ must be a total recursive function. Suppose $W_j \neq \Delta_{\gamma}$. Then for infinitely many $n$ and $s$, $a_s(n) = \gamma$, so by Lemma 1 (i) there are infinitely many $n$ such that $a(n) = \alpha$. But then, for all these $n$, by the construction, no $\beta \in \bigcup_{m<n} C(m)$ will put a number into $A$ after $t(n)$; every $\beta \in C_i(n)(n)$ is initialized at $t(n)$, so its number $z > t(n)$ if it enters after $t(n)$; and any other $R_{e}$-strategy $\beta$ has $|\beta| > t(n)$. Therefore we have an increasing sequence $\{v_n\}_{n \in w}$ such that $\Phi^A_k(x,v_n) \downarrow = \Phi^A_{i,\alpha(t(n))}(x,v_n) \downarrow = 1$ while $\Gamma^A(x,s) = 0$ for all $s$ as in the proof of Lemma 2 (ii). This establishes $W_j$ recursive, or $\lim_{s} \Gamma^A(x,s) = 0$ and not $\lim_{s} \Phi^w_k(x,s) = 0$, in the case $\gamma^A(0) \subset f$.

The last two lemmas complete the proof of the theorem.

**References**


15. G. E. Sacks, *On the degrees less than $\mathbf{0}'$,* Ann. of Math. (2) 77 (1963), 211–231.


1980 *Mathematics subject classifications: 03D25*

Department of Mathematics and Computer Science, State University of New York, College at New Paltz, New Paltz, NY 12561, USA

Department of Mathematics, University of Wisconsin, Madison, WI 53706, USA