COMPUTABILITY AND THE SYMMETRIC DIFFERENCE OPERATOR

URI ANDREWS, PETER M. GERDES, STEFFEN LEMPP,
JOSEPH S. MILLER, AND NOAH D. SCHWEBER

ABSTRACT. Combinatorial operations on sets are almost never well-defined on Turing degrees, a fact so obvious that counterexamples are worth exhibiting. The case we focus on is the symmetric difference operator; there are pairs of (nonzero) degrees for which the symmetric-difference operation is well-defined. Some examples can be extracted from the literature, for example, from the existence of nonzero degrees with strong minimal covers. We focus on the case of incomparable r.e. degrees for which the symmetric-difference operation is well-defined.

1. Introduction

We generally take it for granted that combinatorial set operations and Turing complexity come from different universes and have no relationship to each other. For example, for any two Turing degrees $\mathbf{a}$ and $\mathbf{b}$, there are sets $A \in \mathbf{a}$ and $B \in \mathbf{b}$ so that $A \cap B$ is empty. This is, of course, expected. But as we will see, the symmetric-difference operator is a natural set operation that is *sometimes* well-defined on Turing degrees. That is, there are degrees $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ such that if $A \in \mathbf{a}$ and $B \in \mathbf{b}$, then $A \,\Delta\, B \in \mathbf{c}$. If this is the case, we write $\mathbf{a} \,\Delta\, \mathbf{b} = \mathbf{c}$. Note that for any $A \in \mathbf{a}$ and $B \in \mathbf{b}$, we have $(A \oplus \emptyset) \,\Delta\, (\emptyset \oplus B) \in \mathbf{a} \vee \mathbf{b}$. Thus, the only possible value for $\mathbf{a} \,\Delta\, \mathbf{b}$, if it exists, is $\mathbf{a} \vee \mathbf{b}$.

To be clear, it is easy to construct examples of degrees that do not have a well-defined symmetric-difference. For example, let X, Y, and Z be independent in the sense that the join of any two fails to compute the third. Let $A = X \oplus Y \oplus \emptyset$ and $B = X \oplus \emptyset \oplus Z$. Then A and B are Turing incomparable and $A \,\Delta\, B = \emptyset \oplus Y \oplus Z \not\equiv_{\mathbf{T}} A \oplus B$ (in particular, $A \,\Delta\, B \not\geq_{\mathbf{T}} X$). Therefore, $\deg_{\mathbf{T}}(A) \,\Delta\, \deg_{\mathbf{T}}(B)$ does not exist.

Indeed, it is fair to say that $\mathbf{a} \,\Delta\, \mathbf{b}$ does not exist for *most* pairs of degrees $\mathbf{a}$ and $\mathbf{b}$. Specifically, if $A \in \mathbf{a}$ and $B \in \mathbf{b}$ are 1-generic or 1-random relative to each other, then $A \,\Delta\, B$ is Turing incomparable with both A and B, hence certainly not equivalent to $A \oplus B$.

So how do we know that there are (nonzero) degrees for which the symmetric-difference is well-defined? For a quick example, consider $\mathbf{b}$ to be a strong minimal

---

Date: March 31, 2021.
2020 *Mathematics Subject Classification.* Primary 03D25.
Andrews was partially supported by NSF Grant DMS-1600228. Lempp's research was partially supported by AMS-Simons Foundation Collaboration Grant 626304. Miller was partially supported by grant #358043 from the Simons Foundation. Schweber's research was partially supported by NSF postdoctoral grant DMS-1606455.

cover of $\mathbf{a}$. For any $A \in \mathbf{a}$ and $B \in \mathbf{b}$, we have $A \,\Delta\, B \leq_{\mathbf{T}} A \oplus B \equiv_{\mathbf{T}} B$ and $A \oplus (A \,\Delta\, B) \geq_{\mathbf{T}} B$. But the only degree $\leq_{\mathbf{T}} \mathbf{b}$ that joins $\mathbf{a}$ to $\mathbf{b}$ is $\mathbf{b}$ itself. Therefore $A \,\Delta\, B \in \mathbf{b}$, proving that $\mathbf{a} \,\Delta\, \mathbf{b}$ exists.

The same argument works for any pair of degrees $\mathbf{a} < \mathbf{b}$ such that no degree below $\mathbf{b}$ joins $\mathbf{a}$ up to $\mathbf{b}$. Slaman and Steel [SS89] and Cooper [Co89] independently showed that there is a pair of recursively enumerable degrees $\mathbf{a} < \mathbf{b}$ so that there is no Turing degree $\mathbf{c} < \mathbf{b}$ such that $\mathbf{a} \vee \mathbf{c} = \mathbf{b}$. Thus, there is a pair of comparable recursively enumerable degrees with $\mathbf{a} \,\Delta\, \mathbf{b}$ defined.

The examples above give us comparable pairs of degrees for which the symmetric-difference is well-defined. To get an incomparable pair with this property, consider the fact that the diamond lattice embeds as an initial segment of the Turing degrees [Sa63]. Let $\mathbf{a}$ and $\mathbf{b}$ be the incomparable degrees from such an embedding. Then for any $A \in \mathbf{a}$ and $B \in \mathbf{b}$, we know that $A \,\Delta\, B \leq_{\mathbf{T}} A \oplus B$ and it cannot be below either A or B; therefore $A \,\Delta\, B \equiv_{\mathbf{T}} A \oplus B$ and $\mathbf{a} \,\Delta\, \mathbf{b}$ is defined.

What about incomparable r.e. degrees? In this paper, we show that there are, in fact, incomparable recursively enumerable degrees $\mathbf{a}$ and $\mathbf{b}$ so that $\mathbf{a} \,\Delta\, \mathbf{b}$ is defined. Furthermore, $\mathbf{a}$ and $\mathbf{b}$ can be chosen to be low and to form a minimal pair.

We then examine a degree-theoretic condition that is sufficient to imply that $\mathbf{a} \Delta \mathbf{b}$ is defined. This condition C states that the only degree $\mathbf{c} \leq \mathbf{a} \vee \mathbf{b}$ such that $\mathbf{a} \vee \mathbf{c} \geq \mathbf{b}$ and $\mathbf{b} \vee \mathbf{c} \geq \mathbf{a}$ is $\mathbf{a} \vee \mathbf{b}$. Note that if $\mathbf{a}$ and $\mathbf{b}$ satisfy condition C, then $\mathbf{a} \,\Delta\, \mathbf{b}$ is defined: for any $A \in \mathbf{a}$ and $B \in \mathbf{b}$, the degree of $A \,\Delta\, B$ joins each of $\mathbf{a}$ and $\mathbf{b}$ above the other. We strengthen our construction to show that there is a pair of incomparable recursively enumerable degrees $\mathbf{a}$ and $\mathbf{b}$ that satisfy condition C. Furthermore, this pair can be made low and a minimal pair.

Note that the examples we gave above, such as strong minimal covers, all satisfy condition C. This leads us to ask if condition C is necessary to ensure the existence of $\mathbf{a} \,\Delta\, \mathbf{b}$. We show that it is not, even for the r.e. degrees: we construct a pair of incomparable recursively enumerable degrees $\mathbf{a}$ and $\mathbf{b}$ such that $\mathbf{a} \,\Delta\, \mathbf{b}$ is defined yet they fail to satisfy condition C.

2. Notational Conventions

We adopt relatively standard notational conventions similar to those found in Odifreddi [Od99]. For clarity, we list some of our conventions below.

- We denote the use of $\Phi_{i,s}(X;y)$ by $\mathfrak{u}\,[\Phi_{i,s}(X;y)]$, where this is at least 1 plus the maximum number whose membership in X is queried in $\Phi_{i,s}(X;y)$ and taken to satisfy $y < \mathfrak{u}\,[\Phi_{i,s}(X;y)] = \mathfrak{u}\,[\Phi_i(X;y)] < s$ if $\Phi_{i,s}(X;y)\downarrow$, and ∞ otherwise. Adopting the standard convention that if $\Phi_{i,s}(X;y)\downarrow$ then $y < s$ makes it clear there is no tension between left and right sides of the above inequality.

- Given partial functions ξ and η, we write $\xi \prec \eta$ (η *extends* ξ) to indicate that $\mathrm{dom}(\eta) \supseteq \mathrm{dom}(\xi)$ and η agrees with ξ on $\mathrm{dom}(\xi)$. We write $\xi \not\mid \eta$ to indicate there is a common extension of ξ, and η and $\xi \mid \eta$ when there is not.

- If ξ and η are partial functions on proper initial segments of ω, we write $\xi^\frown\eta$ to denote the function formed by concatenating ξ with η. We denote the partial function ξ with $\xi(0) = x_0$, $\xi(1) = x_1$, …, $\xi(n) = x_n$ and no other values by $\langle x_0, x_1, \ldots, x_n\rangle$, and the empty function by $\langle\rangle$.
- We identify sets with their characteristic function and, by abuse of notation, identify $X\restriction_n$ with the function in 2^n that agrees with the characteristic function of $X\restriction_n$.
- When building X via a stagewise approximation, any changes made at stage s are reflected in X_{s+1} but not in X_s.

3. Incomparable R.E. sets with the Symmetric Difference Property

The examples we gave in the introduction left open the case of incomparable r.e. degrees. We now construct such a pair $\mathbf{a}, \mathbf{b}$ for which $\mathbf{a} \,\Delta\, \mathbf{b}$ is well-defined. ask whether or not this same phenomenon can occur in the diametrically opposite setting of the r.e. degrees. To this end, we prove the following

Theorem 1. *There are (Turing) incomparable r.e. sets A and B such that for any $\widehat{A}$ and $\widehat{B}$ with $\widehat{A} \equiv_{\mathbf{T}} A$ and $\widehat{B} \equiv_{\mathbf{T}} B$, we have $\widehat{A} \,\Delta\, \widehat{B} \equiv_{\mathbf{T}} A \oplus B$.*

We first adopt the following

Notation 3.1.

$$\widehat{X}_{i,s}(z) = \begin{cases} \uparrow & \text{if } (\exists y < z)\widehat{X}_{i,s}(y)\uparrow \\ \Phi_{i,s}(X_s; z) & \text{otherwise} \end{cases}$$

$$\widehat{X}_i(z) = \lim_{s\to\infty} \widehat{X}_{i,s}(z)$$

Using this notation, we note that it is sufficient to build r.e. sets A and B satisfying the following requirements:

$\mathscr{P}_e^A$: $\quad \Phi_e(A) \neq B$,

$\mathscr{P}_e^B$: $\quad \Phi_e(B) \neq A$, and

$\mathscr{R}_{i,j}$: $\quad \Phi_i(\widehat{A}_i) = A \wedge \Phi_j(\widehat{B}_j) = B \implies \Gamma_{i,j}(\widehat{A}_i \,\Delta\, \widehat{B}_j) = A \oplus B.$

Note that we may safely assume that the functionals from the hatted sets to the unhatted sets are the same as those from the unhatted sets to the hatted sets by assuming that $0 \notin A, B$ and noting that it is sufficient to prove the theorem if the requirements are satisfied for all $\widehat{A}_i, \widehat{B}_j$ with $0 \in \widehat{A}_i, \widehat{B}_j$.

3.1. **Intuition.** We satisfy $\mathscr{R}_{i,j}$ by enumerating axioms of the form $\Gamma_{i,j}(\sigma; x) = y$, where $\sigma \in 2^{<\omega}$, to define the functional $\Gamma_{i,j}$. Such an enumeration represents the commitment that if $Z \succ \sigma$ then $\Gamma_{i,j}(Z; x) = y$. Since $A \oplus B$ is r.e., we may presume that anytime x enters $A_s \oplus B_s$, we enumerate the axiom $\Gamma_{i,j}(\emptyset; x) = 1$, so we may focus our attention on axioms of the form $\Gamma_{i,j}(Z; x) = 0$. (Technically, this may result in contradictory definitions since there may be previous axioms with output 0; so once $x \in A_s \oplus B_s$, we really only define $\Gamma_{i,j}(Z; x) = 0$ when Z disagrees with the uses of previously enumerated contradictory axioms.)

Note that if we only had to worry about $\widehat{A}_{i,s}$ (e.g., if $\widehat{B}_j$ were computable), it would be enough to simply copy the computation given by $\Phi_{i,s}(\widehat{A}_{i,s};x)$, i.e., to enumerate $\Gamma_{i,j}(\widehat{A}_{i,s} \Delta \widehat{B}_j;x) = B(x)$, whenever we see $\Phi_{i,s}(\widehat{A}_{i,s};x)\downarrow = 0$. If x later enters A_s, then either $\widehat{A}_{i,s}$ must change below the use of the computation $\Phi_{i,s}(\widehat{A}_{i,s};x)\downarrow = 0$, or else $\mathscr{R}_{i,j}$ is trivially satisfied.

However, since $\widehat{B}_j$ is not computable, we face the risk that x later enters A_s, forcing some $\widehat{x}$ to enter (or exit) $\widehat{A}_{i,s'}$, but the change in $\widehat{A}_{i,s'} \Delta \widehat{B}_{j,s'}$ is later canceled out when some y enters B_t, causing $\widehat{x}$ to enter (or exit) $\widehat{B}_{j,t'}$ and leaving $\widehat{A}_{i,s-1} \Delta \widehat{B}_{j,s-1}$ equal to $\widehat{A}_{i,t'} \Delta \widehat{B}_{j,t'}$ on the initial segment used to define $\Gamma_{i,j}(\widehat{A}_i \Delta \widehat{B}_j;x)$, despite the entry of x into A.

Naively, we might hope to simply react to the entry of $\widehat{x}$ into $\widehat{A}_i$ by restraining any element y from entering B_s that is not above the use of $\Phi_{j,s}(B_s;\widehat{x})$, so $\widehat{x}$ cannot also enter $\widehat{B}_j$. However, the construction must be allowed to proceed even if $\Phi_j(B;\widehat{x})\uparrow$, so we cannot prevent $\widehat{x}$ from entering one or both of $\widehat{A}_i$ and $\widehat{B}_j$ while it appears that $\Phi_j(\widehat{B}_j) \neq B$ or $\Phi_i(\widehat{A}_i) \neq A$. We can, however, refrain from enumerating any axioms involving $\widehat{x}$ until it once again appears that $\Phi_j(\widehat{B}_j) = B$ and $\Phi_i(\widehat{A}_i) = A$, so we know whether or not $\widehat{x}$ enters $\widehat{B}_j$ before enumerating the axiom using the entry of $\widehat{x}$ into $\widehat{A}_i$ to predict if x enters A.

A further difficulty arises from such an approach. We might repeatedly enumerate elements x_k into A in response to very-low priority requirements, each time restraining B to prevent $\widehat{x}_k$ from entering $\widehat{B}_j$ and thus prevent a high-priority requirement from every enumerating an element into B. To this end, we choose possible witnesses x_k for lower-priority requirements so that if $x_{k'}$ is enumerated into A or B after x_k with $k' < k$ then the enumeration of $x_{k'}$ must produce changes in $\widehat{B}_j$ or $\widehat{A}_i$ smaller than the least change produced by x_k. Note that we need only ensure that higher-priority requirements remain free to act after the lower-priority requirements as we may allow the higher-priority requirements to injure the lower-priority requirements.

The net effect of this strategy is to space out candidates for enumeration into A or B so far, from the point of view of $\mathscr{R}_{i,j}$, that the least element entering A or B (after any stage) changes $\widehat{A}_i$ or $\widehat{B}_j$ below any change allowed by $\widehat{B}_j$ or $\widehat{A}_i$, respectively.

3.2. **Overview.** The construction proceeds as a standard tree argument with $\mathcal{R}_{i,j}$ (if on the true path) responsible for meeting $\mathscr{R}_{i,j}$, and $\mathcal{P}_e^X$ responsible for meeting $\mathscr{P}_e^X$. We assign the module $\mathcal{R}_{i,j}$ to α if $|\alpha| = 3\langle\langle i,j\rangle\rangle$, the module $\mathcal{P}_e^A$ to α if $|\alpha| = 3e+1$ and $\mathcal{P}_e^B$ if $|\alpha| = 3e+2$.

During the construction, each node β will receive an infinite *increasing* stream of potential witnesses (balls) from its parent, allowing nodes to control what their descendants may enumerate. We adopt the following conventions regarding the motion of balls on our tree during the construction and remind the reader of and/or augment the standard conventions for a priority tree construction.

Notation 3.2.

- $\mathbb{T}$ is the tree of strategies (regarded as a subtree of $(\omega^{<\omega}, \prec)$), and θ, ξ, η, ν are nodes (elements of $\omega^{<\omega}$) in the tree. We denote the immediate predecessor (parent) of θ on the tree by θ^-, the root node by $\langle\rangle$, and the height of a node ξ by $|\xi| = |\{\nu \mid \nu \prec \xi \wedge \nu \neq \xi\}|$. The outcomes of ξ are those integers x such that $\xi^\frown\langle x\rangle \in T$.
- We write $\xi <_L \nu$ for the relation ξ is left of ν, and $\xi \prec_L \nu$ for the relation ξ is left of or extended by ν). By an abuse of notation, we also write $x <_L y$ to indicate that outcome x of ξ is left of outcome y of ξ.
- $\mathbf{f}_s \in \mathbb{T}$ is the stagewise approximation to the true path $\mathbf{f} \in [\mathbb{T}]$ with $\mathbf{f}(n) = \liminf_{s\to\infty} \mathbf{f}_s\restriction_(n+1)$ relative to the ordering $<_L$.
- If $s = 0$ or $\mathbf{f}_s <_L \theta$, we say θ is *(re)initialized* at s.
- s_α (the s-th α stage) is the number of times α has been visited prior to stage s since it was last initialized. to be the number of times that α has been visited prior to s. More formally, $s_\alpha = |\{t \mid \mathbf{f}_t \succeq \alpha \wedge s_0 < t < s\}|$, where s_0 is the greatest stage less than s at which α is initialized.
- The n-th ball received by β (since it was last initialized) from β^- is denoted x^β_n, and we let $x^\beta_{-1} = -1$. Note that we ensure x^β_n is monotonically increasing in n.
- If β is initialized, all balls located at β are removed from the tree.
- At the start of stage $s > 0$, we place ball s at the root, at which point it is passed down to descendant nodes until it occupies some node or is permanently discarded from the tree. We assume every time the node β is visited, it receives a ball from β^- larger than any previous ball it has received, i.e., nodes are not allowed to reorder the balls before passing them along.
- We diverge slightly from common usage and ensure that $\mathbf{f}_t$ is always the least node $\beta \preceq \mathbf{f}_t$ such that either $t_\beta = 0$ (i,.e., β has not been visited since it was last initialized) or the module at β chooses not to visit any outcome. To ensure that $\mathbf{f} \in [\mathbb{T}]$, we ensure that $(\exists^\infty s)\,(\mathbf{f}_s \succeq \beta)$ implies $(\exists^\infty s)\,(\exists \beta^+ \succ \beta)\,(\mathbf{f}_s \succeq \beta^+)$.

As discussed above, our construction will need to ensure that if $x^\beta_{n'}$ is enumerated into A or B, then higher-priority requirements still remain free to enumerate some x^β_n with $n < n'$. However, we cannot necessarily ensure the reverse, i.e., if x^β_n is enumerated into A or B, we remain free to also use some already considered $x^\beta_{n'}$. To ensure that after the entry of x^β_n into A or B we can select new witnesses for lower-priority requirements, we maintain the following ordering condition.

Condition 3.1. If x is placed into either A or B at stage s, then any $y > x$ placed on the tree before stage s is removed from the tree by the end of stage s.

As balls arrive at nodes in a monotonically increasing fashion, this condition is equivalent to demanding that the m-th ball x^β_m received by β cannot be placed into A or B after the n-th ball x_n^β if $m > n$ and the m-th ball was already defined when the n-th ball was placed.

We will also maintain a version of the standard condition for tree arguments ensuring that only descendant nodes can potentially inflict injury.

Condition 3.2. Suppose that $\alpha \prec \beta$ or $\beta <_L \alpha$, and that α enumerates x into either A or B at stage s. Then either $x > s_\beta$, or β is initialized before being visited again.

Provided that a node β only examines s_β steps of any computation, the module at β can assume that the only nodes which may have disrupted previously observed computations are descendants of β.

3.3. $\mathcal{P}_e^X$**.** We now describe the behavior of the module $\mathcal{P}_e^X$ when assigned to node α on the assumption that α is active at stage s. Here we let X denote either A or B, and we let Y denote the other, i.e., if $X = A$ then $Y = B$ and vice versa.

The module $\mathcal{P}_e^X$ has two outcomes $\ulcorner w \in X \urcorner <_L \ulcorner w \notin X \urcorner$. We describe the behavior of this module by considering the following cases.

Case $s_\alpha = 0$: This is the first time this node is being executed since it was last initialized. Set $w = x_0^\alpha$ claiming the first ball passed from α^- as a potential witness and select outcome $\ulcorner w \notin X \urcorner$ (which will not be visited at this stage). Take no other action.

Case $s_\alpha > 0$: We consider the following subcases.

- Case $w \in X_s$: Set the outcome to $\ulcorner w \in X \urcorner$ and pass any balls received at this stage to $\alpha^\frown \ulcorner w \in X \urcorner$.
- Case $w \notin X_s$: We consider the following subcases.
  - Case $\Phi_{e,s_\alpha}(Y_s; w) \neq X_s(w)$: Set the outcome to $\ulcorner w \notin X \urcorner$ and pass any balls received at this stage to $\alpha^\frown \ulcorner w \notin X \urcorner$.
  - Case $\Phi_{e,s_\alpha}(Y_s; w)\downarrow = X_s(w)$: Note that this only occurs if $X_s(w) = 0$ since we hold w out of X as long as α remains on the tree. In this case, we enumerate w into X, change the outcome to $\ulcorner w \in X \urcorner$, and pass any balls received at this stage to $\alpha^\frown \ulcorner w \in X \urcorner$.

3.4. $\mathcal{R}_{i,j}$**.** We now describe the behavior of the $\mathcal{R}_{i,j}$ when assigned to node α, where $|\alpha| = 3\langle\langle i, j\rangle\rangle$. This module has two outcomes, $\ulcorner \Gamma \urcorner <_L \ulcorner \neq \urcorner$. The left outcome corresponds to a guess that $\Phi_i(\widehat{A}^\alpha) = A \wedge \Phi_j(\widehat{B}^\alpha) = B$, requiring us to build a functional $\Gamma_\alpha(\widehat{A}^\alpha \,\Delta\, \widehat{B}^\alpha) =^* A \oplus B$, where

$$\widehat{A}_s^\alpha(z) = \begin{cases} \uparrow & \text{if } (\exists y < z)\widehat{A}_s^\alpha(y)\uparrow, \\ \Phi_{i,s_\alpha}(A_s; z) & \text{otherwise.} \end{cases}$$

and likewise for B with i replaced by j. The right outcome corresponds to a guess that $\mathscr{R}_{i,j}$ is trivially satisfied as $\Phi_i(\widehat{A}^\alpha) \neq A \vee \Phi_j(\widehat{B}^\alpha) \neq B$. Note that whenever α is initialized, we abandon our construction of Γ_α and start from scratch. We will define $\Gamma_{i,j}$ to be Γ_α for whatever α is on the true path and implements $\mathcal{R}_{i,j}$.

To describe the behavior of $\mathcal{R}_{i,j}$, we adopt the following notation.

$$e_X = \begin{cases} i & \text{if } X = A \\ j & \text{if } X = B \end{cases} \tag{3.1}$$

Using the convention that the use of a nonconvergent computation is taken to be ∞, as is the use of a functional evaluated at ∞, we now define:

$$\begin{aligned}
\mathfrak{y}^\alpha_s &= x^\beta_{s_\beta - 1} \text{ where } \beta = \alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle \\
\widehat{\mathfrak{l}}^X_\alpha(s) &= \max\left\{x \mid [\forall y < x]\left(\Phi_{e_X, s_\alpha}(\widehat{X}^\alpha_s; x) = X_s(x)\right)\right\} \\
\widehat{\mathfrak{l}}_\alpha(s) &= \min\{\widehat{\mathfrak{l}}^A_\alpha(s), \widehat{\mathfrak{l}}^B_\alpha(s)\} \\
u^X_\alpha(x,s) &= \max_{z<x} \mathfrak{u}\left[\Phi_{e_X,s_\alpha}(\widehat{X}^\alpha_s; z)\right] \\
u_\alpha(x,s) &= \max\{u^A_\alpha(x,s), u^B_\alpha(x,s)\} \\
\widehat{u}^X_\alpha(x,s) &= \max_{z<x} \mathfrak{u}\left[\Phi_{e_X,s_\alpha}(X_s; z)\right] \\
\widehat{u}_\alpha(x,s) &= \max\{\widehat{u}^A_\alpha(x,s), \widehat{u}^B_\alpha(x,s)\} \\
\widehat{\mathfrak{r}}^X_\alpha(s) &= u^X_\alpha(\mathfrak{y}^\alpha_s + 1, s) \\
\widehat{\mathfrak{r}}_\alpha(s) &= \max\{\widehat{\mathfrak{r}}^A_\alpha(s), \widehat{\mathfrak{r}}^B_\alpha(s)\} \\
\mathfrak{r}_\alpha(s) &= \widehat{u}_\alpha(\widehat{\mathfrak{r}}_\alpha(s), s)
\end{aligned} \tag{3.2}$$

Note that $\mathfrak{y}^\alpha_s$ is the last ball passed along to the outcome Γ (or equals -1 if no such ball has yet been passed) and that $\widehat{\mathfrak{l}}_\alpha(s)$ measures the (minimum of) the lengths of agreement of $\Phi_i(\widehat{A}^\alpha_s)$ with A and of $\Phi_j(\widehat{B}^\alpha_s)$ with B. The functions $u_\alpha(x,s)$ and $\widehat{u}_\alpha(x,s)$ capture the uses of the computations of the unhatted sets from the hatted sets, and of the hatted sets from the unhatted sets, respectively. The value $\widehat{\mathfrak{r}}_\alpha(s)$ gives a bound (when we have agreement past $\mathfrak{y}^\alpha_s$) on the hatted use of $(A(\mathfrak{y}^\alpha_s)$ and $B(\mathfrak{y}^\alpha_s)$. Thus, whenever $\widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_s$, the computations of A_s and B_s up through $\mathfrak{y}^\alpha_s$ from their hatted counterparts are preserved as long as neither A_s or B_s are changed below $\mathfrak{r}_\alpha(s)$. The strategy for $\mathcal{R}_{i,j}$ is to ensure that the next ball that could enter A or B is above $\mathfrak{r}_\alpha(s)$. Thus, should $\mathfrak{y}^\alpha_s$ enter A, then $\widehat{A}^\alpha_s$ must change on $\widehat{\mathfrak{r}}_\alpha(s)$, while $\widehat{B}^\alpha_s$ is preserved unmodified on $\widehat{\mathfrak{r}}_\alpha(s)$ (or vice versa with B).

Suppose that α is active at stage s and thus receives ball $x^\alpha_{s_\alpha}$. We consider the following cases (but don't presume all cases are necessarily inhabited).

Case $s_\alpha = 0$: Discard x^α_0 from the tree and end execution. This is the first time α has been visited since last being initialized, so we need not specify an outcome.

Case $s_\alpha = 1$: Set the outcome to $\ulcorner\Gamma\urcorner$ and pass x^α_1 along to $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$.

Case $s_\alpha > 1$: Execution is broken up into the following subcases:

Case $x^\alpha_{s_\alpha} \leq \mathfrak{r}_\alpha(s) \vee \widehat{\mathfrak{l}}_\alpha(s) \leq \mathfrak{y}^\alpha_s$: Select outcome $\ulcorner\neq\urcorner$ and pass $x^\alpha_{s_\alpha}$ to this outcome.

Case $x^\alpha_{s_\alpha} > \mathfrak{r}_\alpha(s) \wedge \widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_s$: In this case, it appears that $\Phi_i(\widehat{A}^\alpha) = A$ and $\Phi_j(\widehat{B}^\alpha) = B$, forcing us to build Γ_α, so we must perform all of the following actions:

(1) Set the outcome to $\ulcorner\Gamma\urcorner$ and pass $x^\alpha_{s_\alpha}$ along to $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$.

(2) For any $x \in A_s \oplus B_s$, enumerate the axiom $\Gamma_\alpha(\langle\rangle; x) = 1$.

(3) For any ball x that has been removed from the tree without being enumerated into A (or B), enumerate the axiom $\Gamma_\alpha(\langle\rangle; 2x) = 0$ (or $\Gamma_\alpha(\langle\rangle; 2x+1) = 0$, respectively). (Recall here that $2x \in A \oplus B$ iff $x \in A$, and $2x+1 \in A \oplus B$ iff $x \in B$.)

(4) If the ball $\mathfrak{y}^\alpha_s$ remains on the tree at the end of this stage for $k = 0, 1$ enumerate the axiom $\Gamma_\alpha(\widehat{A}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} \Delta \widehat{B}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)}; 2x+k) = 0$ Note that $\widehat{\mathfrak{r}}_\alpha(s) < \mathfrak{r}_\alpha(s) < \infty$ (remember that we assume $y < \mathfrak{u}\,[\Phi_i(X; y)]$.

3.5. **Verification.** We begin by verifying that the conditions mentioned above are satisfied.

Lemma 3.1. *Condition 3.1 is satisfied.*

Proof. Given that nodes pass on balls in ascending order and only to their active outcome, it follows that if x and y are located at α and β, respectively, and $\alpha <_L \beta$, then $x < y$. The remainder of the claim follows from the fact that nodes of the form $\mathcal{P}^X_e$ only hold on to the least ball they see and move to their (previously unvisited) leftmost outcome when enumerating that element. □

Lemma 3.2. *Condition 3.2 is satisfied.*

Proof. Suppose that α enumerates an element x into A or B at stage s.

If $\alpha \prec \beta$, then (absent initialization) the outcome of α permanently shifts to the left of any outcomes visited previously (since last initialization). Any $\beta \succ \alpha$ is either initialized at s or satisfies $s_\beta = 0 < x$ (and 0 is never used by any node).

If $\beta <_L \alpha$, then let $t < s$ be the last stage before s at which β was visited (or 0 if there is no such stage). As α is initialized at t, if follows that $x \geq t+1 > t \geq t_\beta = s_\beta$, where the last equality holds as β is not visited in $(t, s]$. □

We are now in a position to argue that should $\mathscr{R}_{i,j}$ not be trivially satisfied, then we take the left outcome of module $\mathcal{R}_{i,j}$.

Lemma 3.3. *Suppose $\alpha \prec \mathbf{f}$ implements $\mathcal{R}_{i,j}$, $\Phi_i(\widehat{A}^\alpha) = A$, and $\Phi_j(\widehat{B}^\alpha) = B$. Then $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle \prec \mathbf{f}$.*

Proof. Suppose not, and let s_0 be the last stage at which $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$ is visited. Choose s large enough so that $\widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_{s_0+1}$, $l = \widehat{u}_\alpha(u_\alpha(\mathfrak{y}^\alpha_{s_0+1}, s), s)$ is finite, $A\restriction_l = A_s\restriction_l$, and $B_s\restriction_l = B\restriction_l$. Note that our assumption that $\Phi_i(\widehat{A}^\alpha) = A$ and $\Phi_j(\widehat{B}^\alpha) = B$ ensures such l and s exist.

Now let $s' > s$ be a stage at which α is visited and receives a ball $x^\alpha_{s'_\alpha} > l$. Since $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$ is not visited subsequent to s_0, it follows that $\mathfrak{y}^\alpha_{s_0+1} = \mathfrak{y}^\alpha_{s'}$. Hence

$\widehat{\mathfrak{l}}_\alpha(s') > \mathfrak{y}^\alpha_{s'}$ and $x^\alpha_{s'_\alpha} > l = \widehat{u}_\alpha(u_\alpha(\mathfrak{y}^\alpha_{s'}, s), s) = \mathfrak{r}_\alpha(s')$. By construction, we have $\mathbf{f}_{s'} \succeq \alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$, a contradiction. □

Lemma 3.4. *Suppose that α implements $\mathcal{R}_{i,j}$ and $\alpha^+ = \alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$ is visited at stage s. If $w \leq \mathfrak{r}_\alpha(s)$ enters A or B at stage $t \geq s$, then $w = x^{\alpha^+}_m$ for some $m < s_\alpha$, provided α has not been initialized during $[s,t]$.*

Note that we will usually invoke this to show that if no $x^{\alpha^+}_m$ with $m < s_\alpha$ has been enumerated into A or B, then the construction respects the constraint $\mathfrak{r}_\alpha(s)$.

Proof. By hypothesis, α is not initialized on $[s,t]$. Thus, no node $\theta <_L \alpha$ can enumerate any $w \leq \mathfrak{r}_\alpha(s)$. As $\mathfrak{r}_\alpha(s) \leq s_\alpha$, by Condition 3.2, only nodes $\beta \succeq \alpha$ can enumerate such w. At stage s, no balls occupy any node to the right of α^+, and if w is placed on the tree after stage s, then (by the conditions required to visit outcome $\ulcorner\Gamma\urcorner$) $w > x^\alpha_{s_\alpha} > \mathfrak{r}_\alpha(s)$. Thus, the only candidates for such w must already occupy a node $\theta \succeq \alpha^+$. Thus, $w = x^{\alpha^+}_m$ for $m < s_\alpha$, proving the claim. □

Lemma 3.5. *Suppose that α implements $\mathcal{R}_{i,j}$. Then for $X = A, B$ and $s < t \leq \infty$:*

(1) If $X_s\restriction_{\mathfrak{r}_\alpha(s)} = X_t\restriction_{\mathfrak{r}_\alpha(s)}$, then $\widehat{X}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} = \widehat{X}^\alpha_t\restriction_{\widehat{\mathfrak{r}}_\alpha(s)}$. Furthermore, $\widehat{\mathfrak{r}}_\alpha(s) \leq \widehat{\mathfrak{r}}_\alpha(t)$.

(2) If $\widehat{X}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} = \widehat{X}^\alpha_t\restriction_{\widehat{\mathfrak{r}}_\alpha(s)}$ and $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$ is visited at stage s, then we have $X_s\restriction_{\mathfrak{y}^\alpha_s+1} = X_t\restriction_{\mathfrak{y}^\alpha_s+1}$.

Proof. (1) Note that the claim is trivially satisfied if $\mathfrak{r}_\alpha(s) = \infty$, so we may assume otherwise. The claim now follows immediately from unpacking the definitions. The "Furthermore" clause follows from the fact that $\mathfrak{y}^\alpha_s \leq \mathfrak{y}^\alpha_t$ and that the claim just established ensures that $u_\alpha(\mathfrak{y}^\alpha_s + 1, s) = u_\alpha(\mathfrak{y}^\alpha_s + 1, t) \leq u_\alpha(\mathfrak{y}^\alpha_t + 1, t)$.

(2) This follows by similarly unpacking the definition of $\widehat{\mathfrak{r}}_\alpha(s)$, provided $\widehat{\mathfrak{r}}_\alpha(s) < \infty$. This is guaranteed by the fact that $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$ is visited at stage s, so $\widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_s$. □

Lemma 3.6. *Suppose that α implements $\mathcal{R}_{i,j}$ and $\alpha^+ = \alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle \prec \mathbf{f}$. Then $\Gamma_\alpha(\widehat{A}^\alpha \,\Delta\, \widehat{B}^\alpha) =^* A \oplus B$.*

Proof. We first show that for almost every z, we enumerate an axiom that commits us to $\Gamma_\alpha(\widehat{A}^\alpha \,\Delta\, \widehat{B}^\alpha; z) = (A \oplus B)(z)$.

If x is eventually removed from $\mathbb{T}$ (including by being enumerated into either A or B), then α^+ is visited after this occurs, so we commit to $\Gamma_\alpha(\widehat{A}^\alpha \,\Delta\, \widehat{B}^\alpha; 2x) = (A \oplus B)(2x)$ and $\Gamma_\alpha(\widehat{A}^\alpha \,\Delta\, \widehat{B}^\alpha; 2x+1) = (A \oplus B)(2x+1)$.

If x is not eventually removed from $\mathbb{T}$, then x eventually settles at some node γ. Only finitely many balls settle at nodes where $\gamma <_L \alpha^+$ or $\gamma \prec \alpha^+$, and if $\alpha^+ <_L \gamma$ then x is eventually removed from the tree. Thus, to establish the above claim, we need only show that if x permanently settles at some $\gamma \succeq \alpha^+$, then we enumerate axioms committing us to $\Gamma_\alpha(\widehat{A}^\alpha \,\Delta\, \widehat{B}^\alpha; 2x) = \Gamma_\alpha(\widehat{A}^\alpha \,\Delta\, \widehat{B}^\alpha; 2x+1) = 0$.

If x settles at $\gamma \succeq \alpha^+$, then x is passed to α^+ at some stage s. As $\alpha^+ \prec \mathbf{f}$, at some later stage s', we visit α^+ and enumerate the axioms $\Gamma_\alpha(\widehat{A}^\alpha_{s'}\restriction_l \Delta \widehat{B}^\alpha_{s'}\restriction_l ; 2x) = 0$ and $\Gamma_\alpha(\widehat{A}^\alpha_{s'}\restriction_l \Delta \widehat{B}^\alpha_{s'}\restriction_l ; 2x+1) = 0$, where $l = u_\alpha(x, s')$.

Note that as x remains on $\mathbb{T}$ permanently, α cannot be initialized after s and, by Condition 3.1 applied to $x = x^{\alpha^+}_{s_\alpha}$, no $x^{\alpha^+}_m$ with $m < s_\alpha$ can leave the tree. Thus, as $s'_\alpha - 1 = s_\alpha$, Lemma 3.4 applied to stage s' lets us conclude for $X = A, B$ that $X_{s'}\restriction_{\mathfrak{r}_\alpha(s')} = X\restriction_{\mathfrak{r}_\alpha(s')}$. Therefore, by Lemma 3.5 applied to both A, B, we have $\widehat{A}^\alpha_{s'}\restriction_l \Delta \widehat{B}^\alpha_{s'}\restriction_l = \widehat{A}^\alpha\restriction_l \Delta \widehat{B}^\alpha\restriction_l$, establishing that for almost every z, we enumerate an appropriate axiom.

To finish the proof, we must also verify that Γ_α is well-defined, i.e., all axioms compatible with $\widehat{A}^\alpha \Delta \widehat{B}^\alpha$ specifying a value for $\Gamma_\alpha(\widehat{A}^\alpha \Delta \widehat{B}^\alpha; z)$ give the same value. Since a ball's disposition is final once it is removed from the tree, it is enough to show that axioms enumerated about balls *while* they are on the tree are rendered inapplicable if those balls later enter A or B.

Assume that $x = \mathfrak{y}^\alpha_s$ and that at stage s, x remains on the tree and we visit α^+ and enumerate axioms making the commitments $\Gamma_\alpha(\widehat{A}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} \Delta \widehat{B}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} ; 2x) = 0$ and $\Gamma_\alpha(\widehat{A}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} \Delta \widehat{B}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} ; 2x+1) = 0$, where $\widehat{\mathfrak{r}}_\alpha(s) = u_\alpha(\mathfrak{y}^\alpha_s + 1, s)$. As this is the only scenario in which we enumerate axioms about x before seeing its final disposition, it is enough to show that if x later enters A or B, then $\widehat{A}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} \Delta \widehat{B}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} \neq \widehat{A}^\alpha\restriction_{\widehat{\mathfrak{r}}_\alpha(s)} \Delta \widehat{B}^\alpha\restriction_{\widehat{\mathfrak{r}}_\alpha(s)}$. Note that we may assume that α is not initialized after stage s since that resets Γ_α and discards the axioms.

Let y be the least element enumerated into either A or B at some stage $t^y > s$. Without loss of generality, we may assume that y is enumerated into A. As $y \leq x < \mathfrak{r}_\alpha(s)$, by Lemma 3.4, we know that $y = x^{\alpha^+}_{t_\alpha - 1}$ for some stage $t \leq s$ at which α^+ was visited. By Lemma 3.1, no $x^{\alpha^+}_n$ with $n \leq t_\alpha - 1$ is enumerated into A or B in the interval $[t, t^y)$. By the minimality of y and fact that it is not enumerated into B, we may extend this to $[t, \infty)$ in the case of B. By Lemma 3.4, we can infer that $B_t\restriction_{\mathfrak{r}_\alpha(t)} = B_s\restriction_{\mathfrak{r}_\alpha(t)} = B$ and $A_t\restriction_{\mathfrak{r}_\alpha(t)} = A_s\restriction_{\mathfrak{r}_\alpha(t)}$. Moreover, Lemma 3.5 gives us that $\widehat{B}^\alpha_t\restriction_{\widehat{\mathfrak{r}}_\alpha(t)} = \widehat{B}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(t)} = \widehat{B}^\alpha$, $\widehat{A}^\alpha_t\restriction_{\widehat{\mathfrak{r}}_\alpha(t)} = \widehat{A}^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(t)}$, and $\widehat{\mathfrak{r}}_\alpha(t) \leq \widehat{\mathfrak{r}}_\alpha(s)$.

Since we visit α^+ at stage t, we must have $\widehat{\mathfrak{l}}_\alpha(t) > y$. Thus $\Phi_{i,t_\alpha}(\widehat{A}^\alpha_t; y)\downarrow = 0$. Let $t' > t^y$ be the least stage at which we visit α^+. Hence $\widehat{\mathfrak{l}}_\alpha(t') > y$, ensuring $\Phi_{i,t'_\alpha}(\widehat{A}^\alpha_{t'}; y)\downarrow = 1$. Since $\mathfrak{u}\left[\Phi_{i,t_\alpha}(\widehat{A}^\alpha_t; y)\right] < \widehat{\mathfrak{r}}_\alpha(t)$ and $\mathfrak{u}\left[\Phi_{i,t'_\alpha}(\widehat{A}^\alpha_{t'}; y)\right] < \widehat{\mathfrak{r}}_\alpha(t')$, if $l = \min\{\widehat{\mathfrak{r}}_\alpha(t), \widehat{\mathfrak{r}}_\alpha(t')\}$, then $\widehat{A}^\alpha_{t'}\restriction_l \neq \widehat{A}^\alpha_t\restriction_l = \widehat{A}^\alpha_s\restriction_l$.

Now, by Lemma 3.4, if $w \leq \mathfrak{r}_\alpha(t')$ and w enters A after stage t', then $w = x^{\alpha^+}_m$ for some $m < t'_\alpha$. However, if $m < t_\alpha - 1$, then w would be less than y, contradicting the minimality of y, and, by Lemma 3.1, we know that, if $t_\alpha - 1 \leq m < t'_\alpha$, then $x^{\alpha^+}_m$ has already been removed from the tree. Therefore, $A_{t'}\restriction_{\mathfrak{r}_\alpha(t')} = A\restriction_{\mathfrak{r}_\alpha(t')}$ and by Lemma 3.5, we can infer $\widehat{A}^\alpha_{t'}\restriction_{\widehat{\mathfrak{r}}_\alpha(t')} = \widehat{A}^\alpha\restriction_{\widehat{\mathfrak{r}}_\alpha(t')}$.

Putting these results together, we have $\widehat{A}^\alpha_s\restriction_l \neq \widehat{A}^\alpha_{t'}\restriction_l = \widehat{A}^\alpha\restriction_l$ but $\widehat{B}^\alpha_s\restriction_l = \widehat{B}^\alpha\restriction_l$. This gives us the desired change in only one side of the symmetric difference, letting us conclude that $\widehat{A}^\alpha_s\restriction_l \Delta \widehat{B}^\alpha_s \neq \widehat{A}^\alpha\restriction_l \Delta \widehat{B}^\alpha$. Finally, as $l \leq \widehat{\mathfrak{r}}_\alpha(t) \leq \widehat{\mathfrak{r}}_\alpha(s)$, this suffices to show that the axioms enumerated at stage s for Γ_α are not applicable to $\widehat{A}^\alpha \,\Delta\, \widehat{B}^\alpha$, completing the proof. □

We are now in a position to prove the theorem claimed above:

Theorem 1. *There are (Turing) incomparable r.e. sets A and B such that for any $\widehat{A}$ and $\widehat{B}$ with $\widehat{A} \equiv_{\mathbf{T}} A$ and $\widehat{B} \equiv_{\mathbf{T}} B$, we have $\widehat{A} \,\Delta\, \widehat{B} \equiv_{\mathbf{T}} A \oplus B$.*

Proof. We first note that if $\alpha \prec \mathbf{f}$ and α implements $\mathcal{P}^X_e$, then we satisfy $\mathscr{P}^X_e$. Suppose s_0 is the last stage at which α is initialized and (without loss of generality) $X = A$. The claim is clearly true if $\alpha^\frown\langle\ulcorner w \notin A\urcorner\rangle$ is along the true path, since then, for all stages $s > s_0$ at which α is active, $\Phi_{e,s_\alpha}(Y_s; w) \neq X_s(w)$, and thus $\Phi_e(Y; w) \neq X(w)$. So suppose that at some stage $s > s_0$, α is active and we see $\Phi_{e,s_\alpha}(Y_s; w)\downarrow = X_s(w) = 0$. At this stage, we enumerate w into X and remove all balls currently at nodes $\gamma \succ \alpha$ from the tree. As in a standard priority tree argument, if any node $\gamma \prec \alpha$ enumerated some element $y < s_\alpha$ into Y at some stage $t \geq s$, then α would be initialized at t, contrary to assumption. As all balls b placed on the tree at stage s or later satisfy $b \geq s > s_\alpha$ and no node to the left of α is visited after stage s, it follows that $Y_s\restriction_{s_\alpha} = Y\restriction_{s_\alpha}$. Thus, $\Phi_e(Y; w) \neq X(w)$.

To verify that $\mathscr{R}_{i,j}$ is satisfied, we find $\alpha \prec \mathbf{f}$ implementing $\mathcal{R}_{i,j}$. By Lemma 3.3, if $\alpha^\frown\langle\ulcorner\neq\urcorner\rangle \prec \mathbf{f}$, then either $\Phi_i(\widehat{B}_j) \neq B$ or $\Phi_i(\widehat{A}_i) \neq A$, trivially satisfying $\mathscr{R}_{i,j}$. If instead, $\alpha^+ \prec \mathbf{f}$, then Lemma 3.6 guarantees that we build a functional Γ_α such that $\Gamma_\alpha(\widehat{A}_i \,\Delta\, \widehat{B}_j) =^* A \oplus B$. Patching Γ_α in finitely many places gives us a functional $\Gamma_{i,j}$ witnessing the satisfaction of $\mathscr{R}_{i,j}$. □

Corollary 1.1. *Theorem 1 can be strengthened to make the degrees of A and B a minimal pair.*

Proof. The finitary actions of the modules $\mathcal{P}^A_e$ and $\mathcal{P}^B_e$ pose no threat to the usual minimal pair strategy of continually extending the mutual length of the computation of some set C from A and B while alternatingly offering A or B the chance to interfere with the computations from C but never both at the same time. □

Corollary 1.2. *Theorem 1 and even Corollary 1.1 can be strengthened to make the degrees of A and B low.*

Proof. Despite the uneasy fit between priority trees and lowness requirements, we can achieve this simply by adding modules $\mathcal{N}^X_e$ for $X = A, B$, implementing the usual strategy of freezing the existing computation when they observe $\Phi_{e,s}(X_s; e)\downarrow$. This is accomplished via the simple expedient of selecting the right outcome as long as $\Phi_{e,s}(X_s; e)\uparrow$ and moving to the left outcome when they observe $\Phi_{e,s}(X_s; e)\downarrow$.

This is complicated by the priority tree which, in the general case, might allow the module $\mathcal{N}^X_e$ located at α to observe $\Phi_{e,s}(X_s; e)\downarrow$ infinitely often only for the approximation to the true path to pass to the left of α and injure this computation.

However, we can replace the incompatibility requirements given above with the even simpler requirements.

$\mathscr{P}_e^A$: $\quad \overline{A} \neq W_e$

$\mathscr{P}_e^B$: $\quad \overline{B} \neq W_e$

Using these requirements, we may allow the node β implementing $\mathcal{P}_e^X$ to retain a memory of whether or not they have acted, even through initialization. With this modification, once β has enumerated w into X, it never again, even after initialization, holds a ball and always selects outcome $\ulcorner w \in X \urcorner$.

Since some priority tree constructions of minimal pairs ([So87]) allow ω-branching nodes, we must take some care in placing our modules on the tree so that only boundedly many nodes β' implementing $\mathcal{P}_i^X$ for some i occur below some β implementing $\mathcal{N}_e^X$. Together, this ensures that we only see $\Phi_{e,s}(X_s; e)$ converge finitely many times before permanently preserving this computation. This is enough to make A and B low.

Specifically, we can assign nodes as follows to satisfy the criterion given above. Given β, let

$$\nu(\beta) = \Sigma_{n<|\beta|} \max\{0, \beta(n) - 1\},$$

where

$$[\forall \alpha] \left(\alpha \widehat{\ } \langle n \rangle <_L \alpha \widehat{\ } \langle m \rangle \iff n < m \right).$$

If $\nu(\beta) > 2e \vee |\beta| > 2e$ then assign $\mathcal{N}_i^X$ to β for the least $i \leq e$ with $\mathcal{N}_i^X$ not assigned to $\beta' \prec \beta$ (trying $X = A$ first and then $X = B$) and assign the remaining modules in the normal order at those nodes not occupied by a module of the form $\mathcal{N}_e^X$. This ensures that for each e, there is a finite set of β' such that β' implements a module of the form $\mathcal{P}_i^X$ and β' isn't above some β implementing $\mathcal{N}_i^X$. □

4. A Stronger Condition

The property that drew our attention to the symmetric difference is that if $C = A_0 \,\Delta\, B_0$ with $A_0 \equiv_{\mathbf{T}} A$ and $B_0 \equiv_{\mathbf{T}} B$ then $A \oplus C \geq_{\mathbf{T}} B$ and $B \oplus C \geq_{\mathbf{T}} A$. Thus, a natural question is whether we can extend the result above to all such C.

Theorem 2. *There are (Turing) incomparable r.e. sets A and B such that for any $C \leq_{\mathbf{T}} A \oplus B$ with $A \oplus C \geq_{\mathbf{T}} B$ and $B \oplus C \geq_{\mathbf{T}} A$, we have $C \equiv_{\mathbf{T}} A \oplus B$.*

Our approach to this theorem is essentially the same as above except that we replace the requirement $\mathscr{R}_{i,j}$ by the following requirement (for all $C \subset \omega$) :

$$\mathscr{S}_{i,j,k}: \quad \Phi_i(A \oplus C) = B \wedge \Phi_j(B \oplus C) = A \wedge \Phi_k(A \oplus B) = C \implies \Gamma_{i,j,k}(C) = A \oplus B$$

We adopt the same assignment of nodes to modules as before but assign $\mathcal{S}_{i,j,k}$ to the node α with $|\alpha| = 3\langle\langle i, j, k \rangle\rangle$ instead of the module $\mathcal{R}_{i,j}$ with $|\alpha| = 3\langle\langle i, j \rangle\rangle$.

As before we will ensure that our potential witnesses are spaced out so that if x_k enters A or B then it must result in some change in C below the least change caused by some $x_{k'}$ with $k' > k$. Like $\mathscr{R}_{i,j}$ the module tasked with satisfying $\mathscr{S}_{i,j,k}$ will delay changing to outcome $\ulcorner \Gamma \urcorner$ until we have seen enough convergence to ensure the desired ordering property. Note that we can continue to supply elements to the outcome $\ulcorner\neq\urcorner$ since, when we see sufficient agreement to guarantee the ordering property, all balls passed to the right are removed from the tree.

We only define our functionals Γ_α when we visit the outcome $\ulcorner\Gamma\urcorner$ where, thanks to the ordering property, we can be sure that (absent injury) the commitments we make at some node α trying to meet $\mathscr{S}_{i,j,k}$ are consistent.

4.1. **Module $\mathcal{S}_{i,j,k}$.** We now formally describe the action of $\mathcal{S}_{i,j,k}$ when assigned to node α (so that $|\alpha| = 3\langle\langle i,j,k\rangle\rangle$). As in our description of $\mathcal{R}_{i,j}$ in Section 3.4, this module has two outcomes $\ulcorner\Gamma\urcorner <_L \ulcorner\neq\urcorner$. The left outcome corresponds to a guess that there is some set $C = \Phi_k(A\oplus B)$ satisfying $\Phi_i(A\oplus C) = B$ and $\Phi_j(B\oplus C) = A$, requiring us to build a functional $\Gamma_{i,j,k}(C) = A\oplus B$. The right outcome corresponds to a guess that $\mathscr{S}_{i,j,k}$ is trivially satisfied either because $\Phi_k(A\oplus B)$ is partial, $\Phi_i(A\oplus C) \neq B$ or $\Phi_j(B\oplus C)\neq A$. The basic operation of this module will mimic that described in Section 3.4 for $\mathcal{R}_{i,j}$; however, we must redefine several of the functions used in that argument and include the rest for completeness.

$$
\begin{aligned}
C^\alpha_s(y) &= \begin{cases} \uparrow & \text{if } \exists(x<y)\,(C^\alpha_s(x)\uparrow)\,, \\ \Phi_{k,s_\alpha}(A_s\oplus B_s; y) & \text{otherwise,} \end{cases} \\
\widehat{X}^\alpha_s(z) &= \begin{cases} \uparrow & \text{if } (\exists y<z)\widehat{X}^\alpha_s(y)\uparrow\,, \\ \Phi_{i,s_\alpha}(X_s;z) & \text{otherwise, if } X=A, \\ \Phi_{j,s_\alpha}(X_s;z) & \text{otherwise, if } X=B, \end{cases} \\
\mathfrak{y}^\alpha_s &= x^\beta_{s_\beta-1} \text{ where } \beta = \alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle, \\
\widehat{\mathfrak{l}}_\alpha(s) &= \max\{x \mid [\forall y<x]\,(\Phi_{i,s_\alpha}(A^\alpha_s\oplus C^\alpha_s;x) = B_s(x)\,\wedge \\
&\qquad\qquad\qquad\qquad \Phi_{j,s_\alpha}(B^\alpha_s\oplus C^\alpha_s;x) = A_s(x))\}, \\
u_\alpha(x,s) &= \max\left\{\left\lfloor\frac{\mathfrak{u}\left[\Phi_{i,s_\alpha}(A^\alpha_s\oplus C^\alpha_s;z)\right]}{2}\right\rfloor, \left\lfloor\frac{\mathfrak{u}\left[\Phi_{j,s_\alpha}(B^\alpha_s\oplus C^\alpha_s;z)\right]}{2}\right\rfloor\right\}+1, \\
\widehat{u}_\alpha(x,s) &= \max_{z<x}\left\lfloor\frac{\mathfrak{u}\left[\Phi_{k,s_\alpha}(A_s\oplus B_s;z)\right]}{2}\right\rfloor+1, \\
\widehat{\mathfrak{r}}_\alpha(s) &= u_\alpha(\mathfrak{y}^\alpha_s+1,s), \\
\mathfrak{r}_\alpha(s) &= \max\{\widehat{u}_\alpha(\widehat{\mathfrak{r}}_\alpha(s),s), \widehat{\mathfrak{r}}_\alpha(s)\}.
\end{aligned}
$$

Note that these definitions have been modified so that the same basic relationship as in Section 3.4 is maintained. As before $\mathfrak{y}^\alpha_s$ represents the last ball that was passed to the left. If $\widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_s$ and $\mathfrak{y}^\alpha_s$ is placed into A_s but no other balls below $\mathfrak{r}_\alpha(s)$ are placed into B_s, then a change in C_s below $\widehat{\mathfrak{r}}_\alpha(s)$ must be observed, or the requirement will be trivially satisfied.

Suppose that α is active at stage s and thus receives ball $x^\alpha_{s_\alpha}$. We consider the following cases:

Case $s_\alpha = 0$: Discard x^α_0 from the tree and end execution. This is the first time α has been visited since last being initialized, so we need not specify an outcome.

Case $s_\alpha = 1$: Set the outcome to $\ulcorner\Gamma\urcorner$ and pass x^α_1 along to $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$.

Case $s_\alpha > 1$: Execution is broken up into the following subcases:

CASE $x^\alpha_{s_\alpha} \le \mathfrak{r}_\alpha(s) \vee \widehat{\mathfrak{l}}_\alpha(s) \le \mathfrak{y}^\alpha_s$: Select outcome $\ulcorner\neq\urcorner$ and pass $x^\alpha_{s_\alpha}$ to this outcome.

CASE $x^\alpha_{s_\alpha} > \mathfrak{r}_\alpha(s) \wedge \widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_s$: Then it appears that $\Phi_i(A \oplus C^\alpha) = B$ and $\Phi_j(B \oplus C^\alpha) = A$, forcing us to build Γ_α, so we must perform all of the following actions:

(1) Set the outcome to $\ulcorner\Gamma\urcorner$ and pass $x^\alpha_{s_\alpha}$ along to $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$.

(2) For any $x \in A_s \oplus B_s$, enumerate the axiom $\Gamma_\alpha(\langle\rangle; x) = 1$.

(3) For any ball x that has been removed from the tree without being enumerated into A (or B), enumerate the axiom $\Gamma_{i,j}(\langle\rangle; 2x) = 0$ (or $\Gamma_\alpha(\langle\rangle; 2x+1) = 0$, respectively). Recall that $2x \in A \oplus B$ iff $x \in A$, and $2x+1 \in A \oplus B$ iff $x \in B$.

(4) If the ball $\mathfrak{y}^\alpha_s$ remains on the tree at the end of this stage, enumerate the axioms $\Gamma_\alpha(C^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)}; 2x) = 0$ and $\Gamma_\alpha(C^\alpha_s\restriction_{\widehat{\mathfrak{r}}_\alpha(s)}; 2x+1) = 0$. Note that $\widehat{\mathfrak{r}}_\alpha(s) < \mathfrak{r}_\alpha(s) < \infty$.

4.2. **Verification.** The proof of Theorem 2 merely repeats the argument given in the proof of Theorem 1, but with the definitions from Section 4.1 used in place of those from Section 3.4.

We even note that both corollaries can be strengthened in exactly the same manner to yield the following results.

Corollary 2.1. *Theorem 2 can be strengthened to make the degrees of A and B a minimal pair.*

Corollary 2.2. *Theorem 2 and even Corollary 2.1 can be strengthened to make the degrees of A and B low.*

5. A STRONGER CONDITION?

The similarity of the proofs that we used in the preceding sections—indeed, a similarity so strong that we omitted the bulk of the second proof—naturally raises the question of whether the condition that if $\widehat{A} \equiv_{\mathbf{T}} A$ and $\widehat{B} \equiv_{\mathbf{T}} B$ then $\widehat{A}\,\Delta\,\widehat{B} \equiv_{\mathbf{T}} A \oplus B$ actually guarantees that every $C \le_{\mathbf{T}} A \oplus B$ with $A \oplus C \ge_{\mathbf{T}} B$ and $B \oplus C \ge_{\mathbf{T}} A$ also satisfies $C \equiv_{\mathbf{T}} A \oplus B$. We now demonstrate that this is not the case, even in the r.e. degrees.

Theorem 3. *There are r.e. sets A, B and C with $A \mid_{\mathbf{T}} B$ and $C \le_{\mathbf{T}} A \oplus B$ such that for any $\widehat{A}$ and $\widehat{B}$ with $\widehat{A} \equiv_{\mathbf{T}} A$ and $\widehat{B} \equiv_{\mathbf{T}} B$, we have $\widehat{A}\,\Delta\,\widehat{B} \equiv_{\mathbf{T}} A \oplus B$; however, we also have $A \oplus C \ge_{\mathbf{T}} B$ and $B \oplus C \ge_{\mathbf{T}} A$ but $C \not\ge_{\mathbf{T}} A \oplus B$.*

We prove this theorem by building A, B and C along with explicit computations $\Xi(A \oplus B) = C$, $\Upsilon_1(A \oplus C) = B$, and $\Upsilon_2(B \oplus C) = A$, while satisfying the requirements from Section 3 along with a new requirement $\mathscr{Q}_i$, ensuring that C does not compute $A \oplus B$.

$\mathscr{P}^A_e$: $\Phi_e(A) \neq B$

$\mathscr{P}^B_e$: $\Phi_e(B) \neq A$

$\mathscr{R}_{i,j}$: $\Phi_i(\widehat{A}_i) = A \wedge \Phi_j(\widehat{B}_j) = B \implies \Gamma_{i,j}(\widehat{A}_i \,\Delta\, \widehat{B}_j) = A \oplus B$

$\mathscr{Q}_i$: $\Phi_i(C) \neq A \times B$

As above, each requirement has an associated module responsible for meeting it, and we assign modules to nodes based on the height of the node on the tree. In particular, if $|\beta| = 4i + 3$, then β implements $\mathcal{Q}_i$; if $|\beta| = 4\langle\langle i, j\rangle\rangle$, then β implements $\mathcal{R}_{i,j}$; if $|\beta| = 4e+1$, then β implements $\mathcal{P}^A_e$; and if $|\beta| = 4e+2$, then β implements $\mathcal{P}^B_e$.

The basic approach here will be to take advantage of the fact that the requirements for Theorem 1 look at computations from A and B in isolation, while those for Theorem 2 look at a computation from $A \oplus B$. Thus, given some ball x_0 for potential enumeration into A or B, we seek to meet $\mathscr{R}_{i,j}$ while permitting both A and B to change without changing C. We first wait to observe A_s compute $\widehat{A}$ on a sufficiently long initial segment to freeze $\mathfrak{u}\left[\Phi_i(\widehat{A}; x_0)\right]$ and enumerate a small element into B and thereby C, pushing up the use of both $\Upsilon_1(A_s \oplus C_s; x_0)$ and $\Upsilon_2(B_s \oplus C_s; x_0)$ above that of the frozen initial segment of A. By repeating this trick (alternating between A and B), we become free to pick some x_1 above the region of A and B that we must freeze to meet $\mathscr{R}_{i,j}$, but below the A- and B-uses of $\Upsilon_1(A_s \oplus C_s; x_0)$ and $\Upsilon_2(B_s \oplus C_s; x_0)$. This allows us to enumerate x_0 into A and x_1 into B without changing C, opening up the opportunity for a diagonalization. This approach of enumerating x_0 into A and x_1 into B makes it easier to formulate the requirement $\mathscr{Q}_i$ as a condition about the ability of C to compute $A \times B$ than directly as a condition about computing $A \oplus B$, though it obviously entails the latter.

As in Section 3, we build A and B as recursively enumerable sets and let C contain all balls enumerated into either A or B by the modules $\mathcal{P}^X_e$ and $\mathcal{R}_{i,j}$. We will ensure that if β implements either $\mathcal{R}_{i,j}$ or $\mathcal{P}^X_e$ and β is visited at stage s, then C can determine if the left outcome of β is visited after stage s and, in turn, the motion of balls through these nodes. We will also ensure that if a node β implementing $\mathcal{Q}_i$ is visited infinitely often, then either A or B can determine if the left outcome of β is ever visited and thus the final disposition of the balls occupying β. Inductively, this will ensure that $A \oplus C$ and $B \oplus C$ can determine the eventual fate of any ball, ensuring $A \oplus C \equiv_{\mathbf{T}} B \oplus C \equiv_{\mathbf{T}} A \oplus B$. By the strategy outlined in the previous paragraph, we may allow enumerations of pairs a_i into A and b_i into B by $\mathcal{Q}_i$ without changing C, thus allowing $\mathscr{Q}_i$ to be met.

The operation of $\mathcal{P}^X_e$ remains unchanged, except only that balls enumerated into either A or B are also enumerated into C. However, the operation of $\mathcal{R}_{i,j}$ requires some adjustment.

5.1. **Module $\mathcal{R}_{i,j}$.** Given α with $|\alpha| = 4\langle\langle i, j\rangle\rangle$, we implement $\mathcal{R}_{i,j}$ at α. This module has the same two outcomes $\ulcorner\Gamma\urcorner <_L \ulcorner\neq\urcorner$ as it did in Section 3.4, and we will make use of the same functions defined in Equation (3.2), which we will augment by the following definition:

$$\mathfrak{r}_\alpha^X(s) = \widehat{u}_\alpha^X(\widehat{\mathfrak{r}}_\alpha(s), s) \tag{5.1}$$

Suppose that α is active at stage s and thus receives ball $x^\alpha_{s_\alpha}$. The basic operation of the module remains the same as in Section 3.4, with one main difference. Rather than working jointly to find sufficiently long regions of A and B to freeze, we alternate between A and B so that we may enumerate elements into the other set to convey our progress.

We first ensure that we pass at least one element to outcome $\ulcorner\neq\urcorner$ at each round and then wait until we see a computation from $\widehat{A}^\alpha_s$, correctly predicting A_s up through $\mathfrak{y}^\alpha_s$, and freeze the computation while enumerating an element into B to flag this progress.

We now wait until we see a computation from $\widehat{B}^\alpha_s$, correctly predicting B_s up through $\mathfrak{y}^\alpha_s$, and for $\widehat{B}^\alpha_s$ to be defined on the initial segment $\widehat{A}^\alpha$ used for its computation. We enumerate an element into A (above the frozen computations) and freeze a sufficiently long segment of B (i.e., up to $\mathfrak{r}^B_\alpha(s)$) to preserve $\widehat{B}^\alpha_s$ on both uses (i.e., up to $\widehat{\mathfrak{r}}_\alpha(s)$). Note that $\mathfrak{r}^B_\alpha(s)$ is infinite until all necessary computations converge.

Finally, we switch back to A and wait until $\widehat{A}^\alpha_s$ is defined on the initial segment used by $\widehat{B}^\alpha_s$ to compute B up through $\mathfrak{y}^\alpha_s$. At this point, we enumerate an element into B (larger than the use of the B-computations being preserved) and preserve both A and B on a sufficiently long initial segment, $\mathfrak{r}_\alpha(s)$. We then visit outcome $\ulcorner\Gamma\urcorner$ and pass a ball larger than $\mathfrak{r}_\alpha(s)$ to the nodes below it.

Ultimately, this process ensures that the balls passed along to $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$ have the same properties as they did in Theorem 1, while allowing C (since all balls enumerated into A or B by $\mathcal{R}_{i,j}$ enter C) to determine whether $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$ is ever visited again if α is visited again.

We now describe the behavior of this module more precisely by breaking up its operation into several states, $\mathscr{A}_0$, $\mathscr{B}_0$, $\mathscr{A}_1$, $\mathscr{A}^*_0$, $\mathscr{B}^*_0$, $\mathscr{A}^*_1$ and $\mathscr{G}$ and use the notation that $s \in I$ to indicate that at the start of stage s, the module is in state I. We further simplify the description of the module's operation by naming common actions that will be taken by the module. Note that the informal description above captures the real content of the action of this module, but the additional complexity is required to ensure that we do not retain later balls while enumerating earlier balls into A or B, thereby violating Condition 3.1.

Action $\mathcal{I}$: Select no outcome, discard $x^\alpha_{s_\alpha}$ from the tree, visit no descendant nodes and initialize all nodes extending $\ulcorner\neq\urcorner$.

Action $\mathcal{F}$: Select outcome $\ulcorner\neq\urcorner$ and pass $x^\alpha_{s_\alpha}$ to this outcome.

Note that we can think of Action $\mathcal{I}$ as one step towards the infinitary outcome $\ulcorner\Gamma\urcorner$, and Action $\mathcal{F}$ as the action that is performed if we believe the finitary outcome to be true.

Case $s_\alpha = 0$: Remove x^α_0 from the tree and end execution. This is the first time α has been visited since last being initialized, so we need not specify an outcome.

CASE $s_\alpha = 1$: Set the outcome to $\ulcorner\Gamma\urcorner$ and pass x_1^α along to $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$. Enter state $\mathscr{A}_0^*$.

CASE $s \in \mathscr{X}_i^*$: Set $w = x_{s_\alpha}^\alpha$ and hold for future use. Take action $\mathcal{I}$ and enter state $\mathscr{X}_i$.

CASE $s \in \mathscr{A}_0$: If α was not in state $\mathscr{A}_0$ last time it was visited, then take action $\mathcal{F}$. Otherwise, check if $x_{s_\alpha}^\alpha > \mathfrak{r}_\alpha^A(s)$ and $\widehat{\mathfrak{l}}_\alpha^A(s) > \mathfrak{y}_s^\alpha$. If so, then take action $\mathcal{I}$, enumerate w into B and enter state $\mathscr{B}_0^*$. Otherwise, take action $\mathcal{F}$.

CASE $s \in \mathscr{B}_0$: Check if $x_{s_\alpha}^\alpha > \mathfrak{r}_\alpha^B(s)$ and $\widehat{\mathfrak{l}}_\alpha^B(s) > \mathfrak{y}_s^\alpha$. If so, then take action $\mathcal{I}$, enumerate w into A and enter state $\mathscr{A}_1^*$. Otherwise, take action $\mathcal{F}$.

CASE $s \in \mathscr{A}_1$: Check if $x_{s_\alpha}^\alpha > \mathfrak{r}_\alpha(s)$ and $\widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}_s^\alpha$. If so, then take action $\mathcal{I}$, enumerate w into B and enter state $\mathscr{G}$. Otherwise, take action $\mathcal{F}$.

CASE $s \in \mathscr{G}$: In this case, we enter state $\mathscr{A}_0^*$ and perform all the actions from the final case in Section 3.4, i.e., select outcome $\ulcorner\Gamma\urcorner$, pass $x_{s_\alpha}^\alpha$ down to the child node for that outcome, and enumerate the appropriate axioms for Γ_α.

Note that if the nodes extending outcome $\ulcorner\neq\urcorner$ are initialized infinitely many times, then we visit $\ulcorner\Gamma\urcorner$ infinitely often. Moreover, if α is visited infinitely often, then we visit a descendant of α infinitely often. Thus, the basic structure of the priority tree argument remains intact (we can regard the stages where we initialize the nodes extending $\ulcorner\neq\urcorner$ as visiting another outcome between $\ulcorner\Gamma\urcorner$ and $\ulcorner\neq\urcorner$ which does nothing).

5.2. **Module $\mathcal{Q}_i$.** Given α with $|\alpha| = 4i+3$, we implement $\mathcal{Q}_i$ at α. This module has two outcomes, $\ulcorner w \in A \times B\urcorner <_L \ulcorner w \notin A \times B\urcorner$. Suppose we visit α at stage s. Then this module proceeds as follows.

CASE $s_\alpha = 0$: Let $a = x_0^\alpha$ and hold this ball for later use. Specify no outcome and execute no descendant nodes.

CASE $s_\alpha = 1$: Let $b = x_1^\alpha$, and hold this ball for later use. Also let $w = \langle\langle a, b\rangle\rangle$. Specify no outcome and execute no descendant nodes.

CASE $s_\alpha > 1 \wedge w \in A_s \times B_s$: Specify outcome $\ulcorner w \in A \times B\urcorner$ and pass x_s^α along to that outcome.

CASE $s_\alpha > 1 \wedge w \notin A_s \times B_s$: Consider the following subcases.

> CASE $\Phi_{i,s_\alpha}(C_s; 2a)\uparrow$ or $\Phi_{i,s_\alpha}(C_s; 2a)\downarrow = 1$: Specify outcome $\ulcorner w \notin A \times B\urcorner$ and pass x_s^α along to that outcome.
>
> CASE $\Phi_{i,s_\alpha}(C_s; 2a)\downarrow = 0$: Place a, b into A, B, respectively, and set the outcome to $\ulcorner w \in A \times B\urcorner$.

5.3. **Verification.** We first check that the conditions we specified in Section 3.2 are still maintained.

Condition 3.1. If x is placed into either A or B at stage s, then any $y > x$ placed on the tree before stage s is removed from the tree by the end of stage s.

Lemma 5.1. *Condition 3.1 is maintained in this construction.*

Proof. Assume, for a contradiction, that α is the $\prec_L$-least node to violate the condition. That is, α holds on to some ball w at some stage s_0, and either

(1) w remains on the tree after stage s_1 while some β enumerates some $w^- < w$ into A or B, or

(2) w is enumerated into A or B at s_1 while $w^+ > w$ remains on the tree at some node β.

If $\alpha <_L \beta$, then any w^- enumerated by β would be greater than w, and if $\alpha \preceq \mathbf{f}_{s_1}$, then β is initialized and cannot retain any $w^+ > w$. Thus $\alpha \not<_L \beta$. If $\beta <_L \alpha$, then β would violate the minimality condition on α. Thus we may assume, by minimality of α, that $\alpha \prec \beta$.

Suppose that α implements either $\mathcal{Q}_i$ or $\mathcal{P}_e^X$. Since neither module passes any balls to descendants before stage s_0 (i.e., before reserving any balls it will use), no $\beta \succ \alpha$ receives any ball $w^- < w$ and thus cannot satisfy case 1 above. Furthermore, as both modules $\mathcal{Q}_i$ and $\mathcal{P}_e^X$ visit a previously unvisited left outcome at any stage at which they enumerate a ball into A or B, the action of the tree removes $w^+ > w$ at nodes $\beta \succ \alpha$ from the tree.

Thus α must implement $\mathcal{R}_{i,j}$. Note that $\mathcal{R}_{i,j}$ only visits outcome $\ulcorner \neq \urcorner$ while holding a ball and thus in the interval $[s_0, s_1]$. As the nodes below the outcome $\ulcorner \neq \urcorner$ are initialized at stage s_0, any ball enumerated into A or B by any $\beta \succ \alpha$ in the interval $[s_0, s_1]$ must be larger than w. Thus, we must fall under case 2 above, and thus (for w to be enumerated into A or B), the nodes below the outcome $\ulcorner \neq \urcorner$ are initialized at stage s_1. Hence, no ball $w^+ > w$ remains at a node $\beta \succ \alpha$ after w is enumerated.

As α must implement one of the modules $\mathcal{Q}_i$, $\mathcal{P}_e^X$ or $\mathcal{R}_{i,j}$, this establishes the desired contradiction. □

We must also show that our second condition still holds.

Condition 3.2. Suppose that $\alpha \prec \beta$ or $\beta <_L \alpha$, and that α enumerates x into either A or B at stage s. Then either $x > s_\beta$, or β is initialized before being visited again.

Lemma 5.2. *Condition 3.2 is satisfied.*

Proof. The argument given in Lemma 3.2 remains valid regarding the case where $\beta <_L \alpha$, and when $\alpha \prec \beta$, its reasoning still applies when α implements $\mathcal{P}_e^X$ and, indeed, also applies when α implements $\mathcal{Q}_i$. However, we must supplement this argument by also considering the new case where $\mathcal{R}_{i,j}$ enumerates an element into A or B.

Assume $\alpha \prec \beta$, α implements $\mathcal{R}_{i,j}$, and at some stage s, α enumerates w into A or B. If $\beta \succeq \alpha^\frown\langle\ulcorner \neq \urcorner\rangle$, then β is initialized at s, satisfying the condition. Thus, assume $\beta \succeq \alpha^\frown\langle\ulcorner \Gamma \urcorner\rangle$.

For some earlier stage $s_0 < s$, w is received by α. As balls are received in order, and 0 is not used, at most $w - 1$ balls can have been passed to outcome $\ulcorner \Gamma \urcorner$ by stage s_0. As nodes are only executed at stages at which they receive a ball, it follows that $(s_0)_\beta \leq w - 1 < w$. Finally, as outcome $\ulcorner \Gamma \urcorner$ is not visited during $[s_0, s_1]$, it follows that $s_\beta = (s_0)_\beta < w$. □

We now verify that the same restraint on balls passed to the left outcome of $\mathcal{R}_{i,j}$ is obeyed. Note that elements which $\mathcal{R}_{i,j}$ enumerates into A or B pose no danger themselves as they, like balls extending the outcome $\ulcorner \neq \urcorner$, have always been removed from the tree by the time the next ball is passed to the outcome $\ulcorner \Gamma \urcorner$.

Lemma 5.3. *Suppose α implements $\mathcal{R}_{i,j}$ and x is passed to outcome $\ulcorner \Gamma \urcorner$ at stage s. Then $\widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_s$ and $x > \mathfrak{r}_\alpha(s)$ (where these are the same functions as in Section 3.4).*

Proof. The fact that $\mathfrak{r}_\alpha(s)$ is the same function as in the previous argument is obvious from the definitions. The main claim is trivial if $s_\alpha = 1$, so we may assume that at stage s, $\mathcal{R}_{i,j}$ is in state $\mathscr{G}$ and enumerates $x = x^\alpha_{s_\alpha}$. Thus, at the last stage $t < s$ at which α was active, the module was in state $\mathscr{A}_1$ and satisfied $\widehat{\mathfrak{l}}_\alpha(t) > \mathfrak{y}^\alpha_t$ and $x^\alpha_{t_\alpha} > \mathfrak{r}_\alpha(t)$.

Note that $\widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_t = \mathfrak{y}^\alpha_s$ and $\mathfrak{r}_\alpha(s) = \mathfrak{r}_\alpha(t)$, provided no elements enter A or B below $\mathfrak{r}^A_\alpha(t)$ and $\mathfrak{r}^B_\alpha(t)$, respectively.

As the element w enumerated by $\mathcal{R}_{i,j}$ at stage t satisfies $w > \mathfrak{r}^B_\alpha(t)$, enumerating this element into B violates neither condition. By Condition 3.2, we need only worry about elements enumerated by some $\beta \succeq \alpha$. As the outcome $\ulcorner \Gamma \urcorner$ remains unvisited in $[t, s)$, we need only concern ourselves with the nodes extending outcome $\ulcorner \neq \urcorner$. However, all such nodes were initialized at stage t, and as $x^\alpha_{t_\alpha} > \mathfrak{r}_\alpha(t)$, it follows that any ball b enumerated by such a node satisfies $b > \mathfrak{r}_\alpha(t) = \max\{\mathfrak{r}^B_\alpha(t), \mathfrak{r}^A_\alpha(t)\}$.

Thus, $\widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_t = \mathfrak{y}^\alpha_s$, and as

$$x = x^\alpha_{s_\alpha} > x^\alpha_{t_\alpha} > \mathfrak{r}_\alpha(t) = \mathfrak{r}_\alpha(s),$$

the claim is proved. □

This is enough to show that *when* $\mathcal{R}_{i,j}$ visits its left outcome, it still demonstrates the same behavior. However, to finish the proof that $\mathscr{R}_{i,j}$ is satisfied, we must demonstrate that the modifications to the operation of $\mathcal{R}_{i,j}$ do not prevent the left outcome from being taken when $\mathscr{R}_{i,j}$ is not trivially satisfied. That is, we must reprove Lemma 3.3 in this new context.

Lemma 5.4. *Suppose $\alpha \prec \mathbf{f}$ implements $\mathcal{R}_{i,j}$, $\Phi_i(\widehat{A}^\alpha) = A$ and $\Phi_j(\widehat{B}^\alpha) = B$. Then $\alpha^\frown\langle \ulcorner \Gamma \urcorner \rangle \prec \mathbf{f}$.*

Proof. Suppose not, and let s_0 be the last stage at which $\alpha^\frown\langle \ulcorner \Gamma \urcorner \rangle$ is visited. If there is some later stage at which α is in state $\mathscr{G}$, then the claim is proven. Now, obviously, α cannot occupy the starred states indefinitely, so it is enough to show that for each of the states $\mathscr{A}_0, \mathscr{B}_0, \mathscr{A}_1$, we eventually leave them. So suppose that for all stages $s_1 > s_0$, α remains in one of these three states.

By the assumptions made in the lemma, we can assume that there is some stage $s > s_1$ large enough so that $\widehat{\mathfrak{l}}_\alpha(s) > \mathfrak{y}^\alpha_{s_0+1}$, $l = \widehat{u}_\alpha(u_\alpha(\mathfrak{y}^\alpha_{s_0+1}, s), s)$ is finite, $A\restriction_l = A_s\restriction_l$, and $B_s\restriction_l = B\restriction_l$. Since s_0 is the last stage at which $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$ is visited, it follows that for any $s' > s_0$, $\mathfrak{y}^\alpha_{s'} = \mathfrak{y}^\alpha_{s_0+1}$, and thus $l = \mathfrak{r}_\alpha(s')$. It also follows that $\widehat{\mathfrak{l}}_\alpha(s'), \widehat{\mathfrak{l}}^X_\alpha(s') > \mathfrak{y}^\alpha_{s'}$.

Let $t > s$ be a stage at which $\mathbf{f}_t \succeq \alpha$ with $x^\alpha_{t_\alpha} > l = \mathfrak{r}_\alpha(t)$. By choosing t large enough, since $\alpha \prec \mathbf{f}$, we may also assume that for some $t_0 \in (s, t)$, we have $\mathbf{f}_{t_0} \preceq \alpha$. We consider the following cases:

Case $t \in \mathscr{A}_0$: As $t > t_0 > s_1$ and $\mathbf{f}_{t_0} \succeq \alpha$, it follows that α was in state $\mathscr{A}_0$ last time it was visited. As $x^\alpha_{t_\alpha} > \mathfrak{r}_\alpha(t) = \widehat{u}^A_\alpha(\widehat{\mathfrak{r}}^A_\alpha(t), t)$ and $\widehat{\mathfrak{l}}^A_\alpha(t) \geq \widehat{\mathfrak{l}}_\alpha(t) > \mathfrak{y}^\alpha_t$ it follows that α changes to a new state at stage t.

Case $t \in \mathscr{B}_0$: As $x^\alpha_{t_\alpha} > \mathfrak{r}_\alpha(t) \geq \mathfrak{r}^B_\alpha(t)$ and $\widehat{\mathfrak{l}}^B_\alpha(t) \geq \widehat{\mathfrak{l}}_\alpha(t) > \mathfrak{y}^\alpha_s$, it follows that α changes to a new state at stage t.

Case $t \in \mathscr{A}_1$: As $x^\alpha_{t_\alpha} > \mathfrak{r}_\alpha(t)$ and $\widehat{\mathfrak{l}}_\alpha(t) > \mathfrak{y}^\alpha_t$, it follows that α changes to a new state at stage t.

Thus, eventually, at some stage, α enters state $\mathscr{G}$ and visits outcome $\alpha^\frown\langle\ulcorner\Gamma\urcorner\rangle$, giving us the required contradiction. □

To verify that the set C produced has the property that $C \oplus A \geq_{\mathbf{T}} B$ and $C \oplus B \geq_{\mathbf{T}} A$, we now demonstrate that C has sufficient information to determine what pairs of balls are used by the modules of the form $\mathcal{Q}_i$.

Lemma 5.5. *There is a C-computable function $\zeta(x)$ such that if there is some stage s and node α implementing $\mathcal{Q}_i$, and α has reserved either the ordered pair $\langle\langle x, y\rangle\rangle$ or $\langle\langle y, x\rangle\rangle$, then $\zeta(x) = y$. If no such stage exists, then $\zeta(x) = 0$ (i.e., x is an unused ball).*

Proof. We compute $\zeta(x)$ as follows (setting $\zeta(0) = 0$). We wait until stage x, at which point x is placed onto the tree and is either held by some node α or discarded from the tree. If x is discarded or α does not implement some module of the form $\mathcal{Q}_i$, then output 0. If α implements $\mathcal{Q}_i$ and already holds some ball y, then output y.

So suppose α implements $\mathcal{Q}_i$ but that x is the first ball to be received by α for use. If we ever see a stage s at which a second ball y is received by α, then we output y. If we see a stage $s > x$ at which α is initialized or for some node β, all of the following hold, then we output 0:

(1) $\alpha \succeq \beta^\frown\langle\ulcorner\Gamma\urcorner\rangle$,

(2) at stage s, $\beta \prec \mathbf{f}_s$ is in one of the states $\mathscr{A}_0, \mathscr{B}_0, \mathscr{A}_1$, and

(3) at stage s, β is holding w for enumeration into either A or B and $w \notin C$.

Clearly, if the three conditions above are met, then β never again visits outcome $\ulcorner\Gamma\urcorner$ (and thus α never receives a second ball), since in each of those states, w is enumerated into A or B (and thus C) before moving on to another state. We now must prove that eventually α is initialized, we see a stage at which the above three conditions hold, or α receives a second ball.

Suppose that we do not eventually see α receive a second ball. Then, since nodes receive balls each time they are visited, there is some $\gamma \preceq \alpha$ that is the $\prec$-least ancestor of α not visited infinitely often. Clearly, if γ^- implements $\mathcal{Q}_i$ or $\mathcal{P}_e^X$, then this can only happen if γ extends the right outcome of γ^- and at some stage s, γ^- visits the left outcome, initializing α.

Thus, γ^- implements module $\mathcal{R}_{i,j}$. Now if γ extends outcome $\ulcorner \neq \urcorner$, then we are guaranteed that γ is visited infinitely often. If γ^- eventually settles permanently into one of the states $\mathscr{A}_0, \mathscr{B}_0, \mathscr{A}_1$, then this ensures (from that point on) that the outcome $\ulcorner \neq \urcorner$ is visited every time γ^- is visited. Otherwise, infinitely often, γ^- enters state $\mathscr{A}_0$ despite not having been in that state on the last visit and thus visits outcome $\ulcorner \neq \urcorner$.

So we may suppose that γ extends outcome $\ulcorner \Gamma \urcorner$ of γ^-. Now if γ^- infinitely often entered state $\mathscr{G}$, then γ would be visited infinitely often. Thus there must be some stage $s > x$ at which $\gamma^- \prec \mathbf{f}_s$ is in one of the states $\mathscr{A}_0, \mathscr{B}_0, \mathscr{A}_1$, and, for all $s' > s$, continues to occupy that state. However, in that case, the ball w which γ^- is holding is never enumerated into C, so the conditions are satisfied with $\beta = \gamma^-$. ☐

We also demonstrate that C cannot compute $A \oplus B$:

Lemma 5.6. $C \not\geq_{\mathbf{T}} A \oplus B$

Proof. Suppose that $\Phi_i(C) = A \oplus B$, and let $\alpha \prec \mathbf{f}$ implement $\mathcal{Q}_i$. Let s be large enough so that α is never again initialized and that $\Phi_{i,s_\alpha}(C; 2a) = A(a)$, where a is the first ball reserved by α.

If $a \notin A$, then, by construction, at the first stage after s at which α was visited, a would have been enumerated into A. Thus $a \in A$, and as balls are not reused, there was some stage t after the last initialization of α at which α acted to enumerate (a, b) into $A \times B$. At stage t, we had $\Phi_{i,t_\alpha}(C_t; 2a) = 0$, and by Condition 3.2, only a node $\beta \preceq \alpha$ could disrupt this computation. However, at stage t, all such nodes β that have ever been visited are initialized, ensuring that $C_t\restriction_{t_\alpha} = C\restriction_{t_\alpha}$. ☐

We are now finally in a position to prove the main theorem for this section.

Theorem 3. *There are r.e. sets A, B and C with $A \mid_{\mathbf{T}} B$ and $C \leq_{\mathbf{T}} A \oplus B$ such that for any $\widehat{A}$ and $\widehat{B}$ with $\widehat{A} \equiv_{\mathbf{T}} A$ and $\widehat{B} \equiv_{\mathbf{T}} B$, we have $\widehat{A} \mathbin{\Delta} \widehat{B} \equiv_{\mathbf{T}} A \oplus B$; however, we also have $A \oplus C \geq_{\mathbf{T}} B$ and $B \oplus C \geq_{\mathbf{T}} A$ but $C \not\geq_{\mathbf{T}} A \oplus B$.*

Proof. Lemmas 5.3 and 5.4, along with Conditions 3.1 and 3.2, demonstrate that the construction given in this section has all the same properties used in the proof of Theorem 1. The remainder of the proof, that if $\widehat{A} \equiv_{\mathbf{T}} A$ and $\widehat{B} \equiv_{\mathbf{T}} B$, then $\widehat{A} \mathbin{\Delta} \widehat{B} \equiv_{\mathbf{T}} A \oplus B$, is not interestingly different from that given in Section 3.5.

By Lemma 5.6, we have $C \not\geq_{\mathbf{T}} A \oplus B$. We now demonstrate that $C \oplus A \geq_{\mathbf{T}} B$ and $C \oplus B \geq A$. As the two arguments are symmetric, it is enough to demonstrate $C \oplus A \geq_{\mathbf{T}} B$.

Given C and A, we determine if $x \in B$ as follows. First, we check if $x \in C$ but $x \notin A$. If so, we conclude that $x \in B$. Otherwise, compute $y = \zeta(x)$. Now conclude that $x \in B$ iff $y \neq 0$ and $y \in A$.

To verify that this computation is correct, we note that if x does not end up held by some node α implementing $\mathcal{Q}_i$, then $x \in B$ iff $x \in C$. If x is held by some node α, then $x \notin C$, and x is only in B if α also holds some other ball y and enumerates the pair (y, x) into $A \times B$, i.e., if $\zeta(x) > 0$ and $\zeta(x) \in A$.

This completes the verification. □

References

[Co89] Cooper, S. Barry. *The strong anticupping property for recursively enumerable degrees.* Journal of Symbolic Logic **54** (1989), 527–539. 1

[Sa63] Sacks, Gerald E. *Degrees of unsolvability.* Princeton University Press, Princeton, N.J., 1963. ix+174 pp. 1

[SS89] Slaman, Theodore A. and Steel, John R. *Complementation in the Turing degrees.* J. Symbolic Logic **54** (1989), 160–176. 1

[Od99] Odifreddi, Piergiorgio. *Classical recursion theory. Vol. II.* Studies in Logic and the Foundations of Mathematics, 143. North-Holland Publishing Co., Amsterdam, 1999. 2

[So87] Soare, Robert I. *Recursively Enumerable Sets and Degrees.* Perspectives in Mathematical Logic, Omega Series. Springer–Verlag, Heidelberg, 1987. 3.5

Email address, Andrews: andrews@math.wisc.edu

Email address, Gerdes: gerdes@invariant.org

Email address, Lempp: lempp@math.wisc.edu

Email address, Miller: jmiller@math.wisc.edu

Email address, Schweber: schweber@berkeley.edu

(Andrews, Lempp, Miller, Schweber) Department of Mathematics, University of Wisconsin, Madison, WI 53706-1325, USA

(Gerdes) Department of Mathematics and Statistics, Mathematics and Science Center, Room 368, Oakland University, 146 Library Drive, Rochester, MI 48309-4479, USA