

# THE EXISTENCE OF AN AHMAD TRIPLE

ISKANDER SH. KALIMULLIN, BART KASTERMANS,  
AND STEFFEN LEMPP

ABSTRACT. We construct a so-called *Ahmad triple* in the  $\Sigma_2^0$ -enumeration degrees; that is, we construct three  $\Sigma_2^0$ -sets  $A$ ,  $B$ , and  $C$  such that  $A \not\leq_e B$ ,  $B \not\leq_e C$ ,  $\forall X (X <_e A \Rightarrow X <_e B)$  and  $\forall X (X <_e B \Rightarrow X <_e C)$ .

This constitutes a first step toward showing the  $\forall\exists$ -theory of the  $\Sigma_2^0$ -enumeration degrees to be decidable.

## 1. INTRODUCTION

This paper is a contribution to the theory of the enumeration degrees. We first remind the reader of this structure, and then explain where our construction fits in.

We write  $\langle x, F \rangle$ , with  $x \in \mathbb{N}$  and finite  $F \subseteq \mathbb{N}$ , for the number that is a code for the pair  $(x, F)$  (using the standard pairing function, with a canonical code for  $F$ ). Then we say  $A = \Phi^B$  for a computably enumerable set  $\Phi$  (of such coded pairs) if for all  $x$  we have  $x \in A$  iff there exists an  $F \subseteq B$  such that  $\langle x, F \rangle \in \Phi$ . In this context, we call  $\Phi$  an *enumeration operator*. It is easy to see that  $\Phi$  effectively transforms any enumeration of  $B$  into an enumeration of  $A$ . Another way of looking at this is that  $\Phi$  is a Turing reduction of  $A$  to  $B$  that only uses positive information from  $B$  and computes only positive information about  $A$ . We write  $A \leq_e B$  iff there exists an enumeration operator  $\Phi$  such that  $A = \Phi^B$ .

Using  $\leq_e$ , we define the equivalence relation  $\equiv_e$  by  $A \equiv_e B$  iff  $A \leq_e B$  and  $B \leq_e A$ . We write  $\text{deg}_e(A)$  for the equivalence class of  $A$  and call it the *enumeration degree* of  $A$ . We write  $\text{deg}_e(A) \leq \text{deg}_e(B)$  iff  $A \leq_e B$ .

---

2010 *Mathematics Subject Classification.* 03D30.

Kalimullin's research was partially supported by an NSF U.S.-Russia binational grant. Lempp's research was supported by NSF grants DMS-0140120 and DMS-0555381 as well as Grant # 13407 by the John Templeton Foundation entitled "Exploring the Infinite by Finitary Means". The authors would like to thank Tom Kent for much helpful information and email discussion, and Uri Andrews and Selwyn Ng for a discussion about Remark 4(2).

The structure  $\mathcal{E} = (\mathcal{P}(\mathbb{N})/\equiv_e, \leq)$  is the structure of the enumeration degrees.

We will restrict our attention to a substructure of the enumeration degrees, the  $\Sigma_2^0$ -enumeration degrees. An enumeration degree  $\mathbf{a} \in \mathcal{E}$  is a  $\Sigma_2^0$ -enumeration degree if there is a  $\Sigma_2^0$ -set  $A \subseteq \mathbb{N}$  such that  $\mathbf{a} = \text{deg}_e(A)$ . In this paper, we will focus on the structure

$$\mathcal{D} = (\{\mathbf{a} \mid \mathbf{a} \text{ is a } \Sigma_2^0\text{-enumeration degree}\}, \leq)$$

of the  $\Sigma_2^0$ -enumeration degrees. Cooper [3] observed that a set  $A$  is  $\Sigma_2^0$  iff  $A \leq_e \overline{K}$  (the complement of the halting problem,  $K$ , the enumeration degree  $\mathbf{0}'_e$  of which serves as the counterpart in  $\mathcal{D}$  of  $\mathbf{0}'$  in the Turing degrees); thus the  $\Sigma_2^0$ -degrees are exactly the enumeration degrees  $\leq \mathbf{0}'_e$ , and each  $\Sigma_2^0$ -enumeration degree consists only of  $\Sigma_2^0$ -sets. (Note that the least enumeration degree  $\mathbf{0}_e$  consists of all the computably enumerable sets, including  $K$ .)

Slaman and Woodin [8] showed that the first-order theory of this structure is not decidable; more precisely, they showed the  $\exists\forall\exists\forall\exists$ -fragment to be undecidable. Kent [4] improved this to the  $\exists\forall\exists$ -fragment. From the embedding of the Turing degrees into the enumeration degrees (mapping a Turing degree  $\text{deg}_T(A)$  to the enumeration degree  $\text{deg}_e(A \oplus \overline{A})$ ), it is easy to see that the  $\exists$ -fragment is decidable. All this leaves open the decidability of the  $\forall\exists$ -fragment. The decidability of the  $\forall\exists$ -fragment can easily be seen to be equivalent to the existence of a decision procedure for the following question (put in prenex normal form with the matrix in disjunctive normal form).

**Question 1** (Extendibility Question). Let  $\mathbb{P}$  and  $\mathbb{Q}_i$  (with  $i \leq n$ ) be finite posets such that for all  $i \leq n$ ,  $\mathbb{P} \subseteq \mathbb{Q}_i$ . Under what conditions on  $\mathbb{P}$  and the  $\mathbb{Q}_i$  can any embedding of  $\mathbb{P}$  into the  $\Sigma_2^0$ -enumeration degrees be extended to an embedding of  $\mathbb{Q}_i$  into the  $\Sigma_2^0$ -enumeration degrees for some  $i$  (which may depend on the embedding of  $\mathbb{P}$ )?

Lempp, Slaman, and Sorbi [5] showed that the case  $n = 0$  (often called the Extension of Embeddings Problem) is decidable; this is, of course, the question of whether, given two finite posets  $\mathbb{P} \subseteq \mathbb{Q}$ , any embedding of  $\mathbb{P}$  into the enumeration degrees can be extended to an embedding of  $\mathbb{Q}$ .

In addition to the two obstacles to extendibility in the case of the c.e. Turing degrees (see, Slaman and Soare [7]), they had to overcome one additional obstacle, a generalization of the existence of an *Ahmad pair* of  $\Sigma_2^0$ -enumeration degrees.

- Definition 2.** (1) An *Ahmad pair* consists of two degrees  $\mathbf{a}$  and  $\mathbf{b}$  such that  $\mathbf{a} \not\leq \mathbf{b}$  and  $\forall \mathbf{x} (\mathbf{x} < \mathbf{a} \Rightarrow \mathbf{x} < \mathbf{b})$ . (We denote this by  $\mathcal{A}(\mathbf{a}, \mathbf{b})$ .)
- (2) An *Ahmad triple* consists of three degrees  $\mathbf{a}$ ,  $\mathbf{b}$ , and  $\mathbf{c}$  such that  $\mathcal{A}(\mathbf{a}, \mathbf{b})$  and  $\mathcal{A}(\mathbf{b}, \mathbf{c})$ .

In her thesis, Ahmad [1] (cf. Ahmad and Lachlan [2]) showed that there exist  $\Sigma_2^0$ -enumeration degrees  $\mathbf{a}$  and  $\mathbf{b}$  such that  $\mathcal{A}(\mathbf{a}, \mathbf{b})$ . She also showed  $\mathcal{A}(\mathbf{a}, \mathbf{b}) \Rightarrow \neg \mathcal{A}(\mathbf{b}, \mathbf{a})$ , i.e., that an Ahmad pair is necessarily “asymmetric”.

Toward deciding the Extendibility Question, several questions about different configurations of degrees need to be answered first. In the present paper, we show that one of these configurations, the Ahmad triple, can be realized in the  $\Sigma_2^0$ -enumeration degrees.

**Theorem 3.** *There exists an Ahmad triple of  $\Sigma_2^0$ -enumeration degrees.*

Other intermediate steps toward deciding the  $\forall\exists$ -fragment include the existence of a so-called “cupping Ahmad pair”, i.e., an Ahmad pair with join of degree  $\mathbf{0}'_e$ ; and the more general question of possible one-point extensions, i.e., the situation where  $|\mathbb{Q}_i - \mathbb{P}| = 1$ . On the former, there is on-going work by the first and third author suggesting that no such pair can exist; on the latter, there is progress in the special case where  $\mathbb{P}$  is an antichain and the  $\mathbb{Q}_i$  add one new point below points in  $\mathbb{P}$ , by the third author with several coauthors.

The remainder of the paper is devoted to proving Theorem 3. Our construction of an Ahmad triple is based on the work in Kent [4]. Our requirements are a subcollection of his requirements where, however, each requirement has two different types depending on which pair in the triple they relate to. The intuition about the construction is similar to Kent [4]; but we analyze the resulting construction differently (compare, e.g., the Tree Lemmas and their proofs). This different analysis combined with careful bookkeeping is why *A*, *B*- and *B*, *C*-requirements do not interfere with each other.

We conclude with a

- Remark 4.** (1) *It is not hard to extend the proof of Theorem 3 to obtain longer chains of Ahmad pairs  $(\mathbf{a}_0, \dots, \mathbf{a}_n)$  such that  $\mathcal{A}(\mathbf{a}_i, \mathbf{a}_{i+1})$  for each  $i < n$ .*
- (2) (Andrews, Lempp, Ng) *The asymmetry of Ahmad pairs extends to longer chains as well, i.e., we cannot have in addition to (1) that  $\mathcal{A}(\mathbf{a}_n, \mathbf{a}_0)$ : Otherwise, fix a “symmetric chain”  $(\mathbf{a}_0, \dots, \mathbf{a}_n, \mathbf{a}_0)$  of Ahmad pairs of minimal length. Now fix any degree  $\mathbf{x} < \mathbf{a}_n$ , which thus must be  $< \mathbf{a}_0$  and so  $< \mathbf{a}_1$ . But this*

means that we have a shorter symmetric chain  $(\mathbf{a}_1, \dots, \mathbf{a}_n, \mathbf{a}_1)$  of Ahmad pairs, a contradiction.

## 2. NOTATION

$a \nearrow A$  is our notation for a situation where  $a$  was a member of  $A$  before this time/stage, but now is no longer (sometimes it describes our action of removing  $a$  from  $A$ ).

$a \searrow A$  is our notation for a situation where  $a$  was not a member of  $A$  before this time/stage, but now is a member (sometimes it describes our action of putting  $a$  into  $A$ ).

If  $S$  is any set, then  $S^{<\mathbb{N}}$  denotes the tree of finite sequences of elements from  $S$  (i.e., functions  $n \rightarrow S$  for some  $n \in \mathbb{N}$ ), and  $\emptyset$  its root. For two elements  $\beta, \alpha \in S^{<\mathbb{N}}$  we write  $\beta \prec \alpha$  iff  $\beta$  is a proper initial segment of  $\alpha$ . We write  $\alpha^-$  for  $\alpha \upharpoonright (|\alpha| - 1)$ , the predecessor of  $\alpha$  (also by convention  $\emptyset^- = \emptyset$ ).

For  $\alpha, \beta \in S^{<\mathbb{N}}$ , say,  $\alpha : n \rightarrow S$  and  $\beta : m \rightarrow S$ , we write  $\alpha \frown \beta$  for the concatenation of the two:

$$(\alpha \frown \beta)(i) = \begin{cases} \alpha(i) & \text{if } i < n; \\ \beta(i - n) & \text{if } n \leq i < n + m. \end{cases}$$

## 3. THE REQUIREMENTS

Here we introduce the requirements used both for the construction of an Ahmad pair and an Ahmad triple.

To prove the theorem, we need to construct  $\Sigma_2^0$ -approximations to three sets  $A$ ,  $B$ , and  $C$  satisfying the requirements:

$$\begin{aligned} (\mathbf{S}_{A,B,\Omega}) & \quad \exists \Gamma (\Omega^A = \Gamma^B) \vee \exists \Delta (A = \Delta^{(\Omega^A)}) \\ (\mathbf{T}_{A,B,\Phi}) & \quad A \neq \Phi^B \\ (\mathbf{S}_{B,C,\Omega}) & \quad \exists \Gamma (\Omega^B = \Gamma^C) \vee \exists \Delta (B = \Delta^{(\Omega^B)}) \\ (\mathbf{T}_{B,C,\Phi}) & \quad B \neq \Phi^C \end{aligned}$$

We introduce the notations

$$\mathcal{R}_p = \{\mathbf{T}_{A,B,\Phi}, \mathbf{S}_{A,B,\Phi} \mid \Phi \text{ is an enumeration operator}\}$$

and

$$\mathcal{R} = \{\mathbf{T}_{A,B,\Phi}, \mathbf{T}_{B,C,\Phi}, \mathbf{S}_{A,B,\Phi}, \mathbf{S}_{B,C,\Phi} \mid \Phi \text{ is an enumeration operator}\}$$

for the set of requirements for an Ahmad pair and an Ahmad triple, respectively.

## 4. THE INTUITION

In this section we explain the intuition of the constructions. This gives us a good handle in understanding and following the formal constructions in the following sections. All the sets we are building will be  $\Delta_2^0$ , meaning that initial segments stabilize.

**4.1. A  $\mathbb{T}_{A,B,\Phi}$ -Requirement in Isolation.** For a  $\mathbb{T}_{A,B,\Phi}$ -requirement in isolation, a Friedberg-Muchnik strategy for enumeration degrees suffices: That is, you pick a witness  $a$  for  $A$ , ensure it is in  $A$ , and wait for  $a \searrow \Phi^B$ . If it never happens, the requirement is satisfied. If it does happen, say, at stage  $s$ , remove  $a$  from  $A$ , and protect  $B[s]$  (by ensuring no elements are removed). This ensures  $a \in \Phi^B \setminus A$ , satisfying the requirement.

**4.2. An  $\mathbb{S}_{A,B,\Omega}$ -Requirement.** For an  $\mathbb{S}_{A,B,\Omega}$ -requirement, we choose a coding point  $b_x$  in  $B$  for each element  $x$  in  $\Omega^A$ , enumerate  $\langle x, \{b_x\} \rangle$  into  $\Gamma$ , and, whenever active, correct the computation by adding or removing  $b_x$  to or from  $B$ . So in this case the only requirement on the coding points is that they are distinct; a coding point should be able to go into or out of  $B$  independently.

**4.3. Streams.** To help with the description of (and proofs about) the interaction of the different requirements, we use the notion of a *stream*. For the construction of an Ahmad pair, every requirement defines an  $A$ -stream consisting of elements  $a$  for which this requirement (and the requirements of higher priority) do not care about membership in  $A$ ; namely, the requirement does not care whether lower-priority strategies enumerate or extract  $a$  into or from  $A$  as long as certain side conditions are met. These side conditions will be violated (or at least no longer guaranteed to be satisfied) for elements  $a' > a$  in the stream as soon as the membership of  $a$  in  $A$  changes (actually, only if  $a \nearrow A$ , since we are using enumeration reductions), and so we will have to remove these  $a'$  from the  $A$ -stream (in the construction, this is done at steps  $(0_{\mathbb{T}_{A,B,\Phi}})$  and  $(0_{\mathbb{T}_{B,C,\Phi}})$ ). We use the notion of suitability (see Section 10) to ensure that actions by lower-priority requirements do not invalidate the side conditions too much. They need to be spaced far enough apart: Between “invalidations” there will be enough progress that in the end the whole reduction can be defined (if we can never diagonalize). We will write  $S_A(\alpha)$  for the  $A$ -stream associated with  $\alpha$ . (For an Ahmad triple, we will also need to define  $B$ -streams essentially sharing the properties for the  $A$ -streams. We write  $S_B(\alpha)$  for the  $B$ -stream associated with  $\alpha$ .)

**4.4. One T-Requirement Below One S-Requirement.** We again attempt to implement a standard Friedberg-Muchnik strategy for enumeration degrees. So we choose a witness  $a$  for  $A$  and enumerate it into  $A$ . Then we wait until  $a$  appears in  $\Phi^B$ —until  $a$  becomes a *realized witness*—at a stage  $s$ . At this time, we want to remove  $a$  from  $A$  to diagonalize.

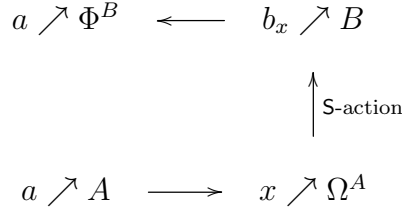


FIGURE 1. The action of a T-strategy below one S-strategy

Figure 1 illustrates the problem with this: Once we take  $a$  out of  $A$ , some (possibly several)  $x$  might be removed from  $\Omega^A$ . This change in  $\Omega^A$  requires correction for the higher-priority S-requirement, which then might change  $B$  (by removing some  $b_x$  from it) in such a way that  $a$  is removed from  $\Phi^B$ . Then clearly we have not succeeded in diagonalizing.

If it were the case that  $\Gamma^B \subseteq \Omega^A \setminus \{a\}$ , then this would not happen. So our strategy becomes to hand over  $a$  to lower-priority requirements to work with until this case appears. Then, when  $a$  becomes realized, we remove the killing points (these points are different from coding points and are explained more in Subsection (4.7)) from  $B$ , ensuring the  $\Gamma$ -axioms that involve these killing points will never be used in a computation from  $B$ . This is itself a change in  $B$ , but one that we'll reverse if ever  $a$  becomes  $\Gamma$ -cleared. Also note that it means we injure the action of the S-requirement of higher priority; this injury is only alright because we now take over the satisfaction of its requirement (changing to satisfying it through its second disjunct).

To carry out this strategy, we wait for (remember,  $s$  is the stage at which  $a$  became a realized witness):

$$(\Gamma\text{-cleared}) \quad \Gamma^B[s] \subseteq \Omega^A \quad \wedge \quad a \notin A$$

While waiting, we pick a new witness and attempt the diagonalization there. If ever one of our witnesses becomes  $\Gamma$ -cleared for the higher-priority  $S_{A,B,\Phi}$ -requirement, then we can diagonalize by restoring  $B$ , that is, adding  $B[s]$  to  $B$ . Now no higher-priority  $T_{A,B,\Phi}$ -requirement

(that is trying to diagonalize) is injured since these can only be injured by removing elements from  $B$ , and by  $\Gamma$ -clearance neither can the higher-priority  $S_{A,B,\Phi}$ -requirement.

There is, however, no reason for any of the witnesses to become  $\Gamma$ -cleared. In that case, we allow the  $S_{A,B,\Phi}$ -strategy to fail (in fact we make it fail by our removal of killing points) and have the  $T_{A,B,\Phi}$ -requirement construct an enumeration operator  $\Delta$  such that  $A = \Delta^{(\Omega^A)}$ , thus satisfying the  $S_{A,B,\Phi}$ -requirement. For this, notice that if the  $A$ -use of  $\Omega$  for the numbers in  $\Gamma^B[s]$  is unchanged other than at the witness  $a$  (which condition we call the side condition for  $a$ ), then we ensure the equivalence

$$a \in A \iff \Gamma^B[s] \subseteq \Omega^A.$$

This allows us to define the operator  $\Delta$  in the following way: Every time we see a witness  $a$  being enumerated into  $\Gamma^B$ , we enumerate an axiom  $\langle a, \Gamma^B[s] \rangle$  into  $\Delta$ . To ensure the side condition on the use of  $A$ , we have this  $T_{A,B,\Phi}$ -requirement generate an  $A$ -stream of elements for which it individually does not care about membership in  $A$ , but such that if an element in the  $A$ -stream is removed from  $A$ , then the side conditions for all larger elements currently in the  $A$ -stream are no longer met, and so these larger elements need to be removed from the  $A$ -stream.

Figure 2 shows the outcomes in this case.

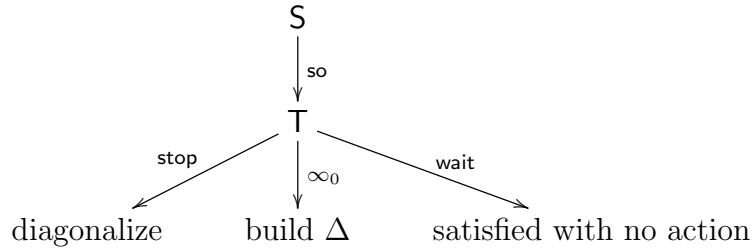


FIGURE 2. A  $T$ -strategy below and  $S$ -strategy with outcomes

**4.5. One  $T$ -requirement below multiple  $S$ -requirements.** Suppose we have  $\alpha_0 \prec \dots \prec \alpha_n \prec \beta$  with the  $T$ -requirement assigned to  $\beta$  and the  $\alpha_i$  being all its predecessors assigned to  $S$ -requirements. Each  $\alpha_i$  is building a  $\Gamma_i$ , trying to ensure  $\Omega_i^A = \Gamma_i^B$ . Now witnesses have different degrees of “clearance”. We look for a witness  $a$  and  $k \leq n$  such that for all  $j$  with  $j \geq k$  we have  $\Gamma_j^B[s] \subseteq \Omega_j^A$  and  $a \notin A$ , but (if  $k \neq 0$ )  $\Gamma_{k-1}^B[s] \not\subseteq \Omega_{k-1}^A$ . If  $k = 0$ , then the witness  $a$  is fully  $\Gamma$ -cleared and we can diagonalize; if  $k > 0$ , then we work at completing

the construction of  $\Delta_k$  in such a way that the  $\Gamma_l$  with  $l < k$  are not injured (which gives the additional restriction that  $a$  should be bigger than all the previous witnesses used for  $\Delta_l$  with  $l < k$ ), but initialize the work for  $\Delta_j$  with  $j > k$ . If we never diagonalize, then for the smallest  $k$  for which we find infinitely many witnesses, we successfully build  $\Delta_k$ . The  $S$ -requirements associated to  $\alpha_i$  with  $i < k$  are not injured, the  $S$ -requirement associated to  $\alpha_k$  is satisfied by this  $\Delta_k$ , and the  $S$ -requirements associated to  $\alpha_i$  with  $i > k$  will need to be reassigned to new nodes since they are terminally injured by this work.

Figure 3 shows the outcomes in this case.

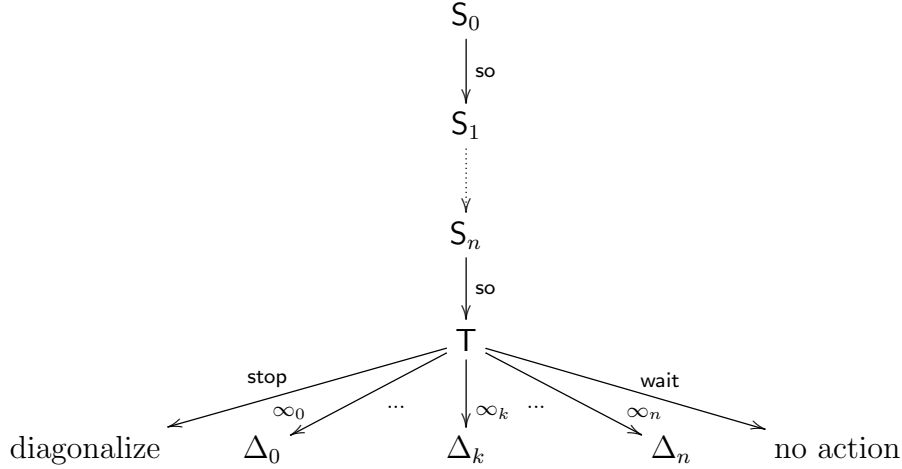


FIGURE 3. A  $T$ -strategy below multiple  $S$ -strategies

**4.6. Multiple  $T$ -requirements.** The case of multiple  $T$ -requirements is essentially the same as for one  $T$ -requirement, the only difference to point out here being that the witnesses for one  $T$ -strategy should be chosen from the  $A$ -stream of the previous  $T$ -strategies.

**4.7. Killing Points.** In Subsection 4.4, there were multiple references to killing points. Most explicitly, we mentioned that in case no witnesses become  $\Gamma$ -cleared, instead of satisfying the  $T$ -requirement, we satisfy the  $S$ -requirement by its other option (building  $\Delta$ ). Since clearly we cannot both have a correct computation  $\Gamma$  and a correct computation  $\Delta$  (because then  $A \leq_e \Omega^A$  witnessed by  $\Delta$  and  $\Omega^A \leq_e B$  witnessed by  $\Gamma$ ), we have to kill  $\Gamma$ . The way we kill  $\Gamma$  is by removal of the killing points. Aside from removing the immediate contradiction, it works as follows: When we add  $a$  to the stream, it will be used by (possibly many) lower priority requirements  $R$ , its coding points could be smaller

than elements from  $B$  used at requirements  $R'$  that are of lower priority but still higher priority than  $R$ . Then if  $R$  changes membership of  $a$  in  $A$ , it also injures  $R'$ . This type of injury is avoided with our use of killing points: When  $a$  is added to the  $A$ -stream, all of its connections ( $\Gamma$ -axioms) to  $B$  are reset (they involve a point not currently nor ever again in  $B$ ).

## 5. THE CONSTRUCTION OF THE AHMAD TRIPLE

**5.1. The tree of strategies.** Effectively order the set of requirements  $\mathcal{R}$  as  $R_e$  for  $e \in \mathbb{N}$ .

The set of outcomes is  $\text{Outc} = \{\text{stop}, \text{wait}, \text{so}\} \cup \{\infty_i \mid i \in \mathbb{N}\}$  ordered as

$$\text{stop} < \infty_0 < \cdots < \infty_i < \infty_{i+1} < \cdots < \text{wait} < \text{so}.$$

The tree of strategies  $\mathcal{T}$  will be the subset of  $\text{Outc}^{<\mathbb{N}}$  consisting of the nodes that have a requirement assigned to it as described below. Any node in the tree also has certain labels associated with it; these are labels to indicate that certain requirements are *satisfied* at  $\alpha$ , and there are also labels to indicate that certain requirements are *active* at  $\alpha$  via some predecessor  $\beta$ .

No strategy is active or satisfied at  $\emptyset$ , and  $\emptyset$  is assigned requirement  $R_0$ .

Given a node  $\alpha \in \mathcal{T}$  with a strategy  $R$  assigned to it, let  $R'$  be the least requirement in the enumeration  $\langle R_e \mid e \in \mathbb{N} \rangle$  that is not labeled active or satisfied at  $\alpha$ .

Now we make a case distinction depending on requirement  $R$ :

**Case  $R = \top_{A,B,\Phi}$ :** Let  $\beta_0 \prec \beta_1 \prec \cdots \prec \beta_i \prec \alpha$  be all the nodes such that there is an  $S_{A,B,\Phi}$ -requirement active at  $\alpha$  via  $\beta_j$ . Write  $S_{A,B,\Phi}^j$  for the requirement assigned to  $\beta_j$  for  $j$  such that  $0 \leq j \leq i$ . (We allow  $i = -1$ .)

Assign  $R'$  to  $\alpha \frown \text{wait}$ ,  $\alpha \frown \text{stop}$ , and  $\alpha \frown \infty_j$  for  $j$  such that  $0 \leq j \leq i$  (these are all the successors of  $\alpha$  that are assigned a requirement), and set it to be active there. Requirement  $R$  is satisfied at  $\alpha \frown \text{wait}$  and  $\alpha \frown \text{stop}$ . For any other requirement that is also different from all  $S_{A,B,\Phi}^j$ , it is active via  $\beta$ , or satisfied, at  $\alpha \frown o$ , for

$$o \in \{\text{stop}, \infty_0, \dots, \infty_i, \text{wait}\},$$

if it is so at  $\alpha$ .

At  $\alpha \frown \text{wait}$  and  $\alpha \frown \text{stop}$ , the requirements  $S_{A,B,\Phi}^j$  are active via  $\beta_j$  as they were at  $\alpha$ .

At  $\alpha \hat{\infty}_j$ , the requirements  $S_{A,B,\Phi}^l$  with  $l < j$  are active via  $\beta_l$ , the requirement  $S_{A,B,\Phi}^j$  is satisfied, and the requirements  $S_{A,B,\Phi}^l$  with  $l > j$  are neither satisfied nor active.

**Case**  $R = S_{A,B,\Phi}$ : Assign  $R'$  to  $\alpha \hat{\text{so}}$  and set  $R'$  to be active at  $\alpha \hat{\text{so}}$ . For any other requirement, it is active via  $\beta$ , or satisfied, at  $\alpha \hat{\text{so}}$  if it is so at  $\alpha$ .

**Case**  $R = T_{B,C,\Phi}$ : Similar to the case when  $R = T_{A,B,\Phi}$  but with  $S_{B,C,\Phi}$  instead of  $S_{A,B,\Phi}$ .

**Case**  $R = S_{B,C,\Phi}$ : Similar to the case when  $R = S_{A,B,\Phi}$ .

Note from the above: A T-requirement is only active at the node it is assigned to, after that it is not assigned a label (this is because the node works at satisfying some higher-priority S-requirement), or it is assigned the label satisfied. An S-requirement is active at and below the node it is assigned to, unless either a T-requirement injures it (satisfying a higher-priority S-requirement), in which case it has no label below the node associated by that T-requirement, or a T-requirement satisfies it, in which case it has a label satisfied assigned below that node.

We call a branch in  $\text{Outc}^{<\mathbb{N}}$  *valid* if every node in the branch has a requirement assigned to it. Now by an easy induction, the following lemma can be seen.

**Lemma 5.** *Every requirement is eventually active or satisfied along every valid branch.*

**5.2. Suitability.** At various times during the construction, we have to choose witnesses that can be used to satisfy certain requirements. We define *t-suitability* here to collect together some of the requirements these numbers need to satisfy.

We say  $x$  is *t-suitable for C* if  $x$  is bigger than any number seen before in the construction. We can take suitability for  $C$  to be this simple, since there are no  $C$ -streams; there are no diagonalizations with witnesses in  $C$ .

We say  $x$  is *t-suitable for  $X \in \{A, B\}$*  at node  $\alpha$  (or  $x$  is *t,  $\alpha$ -suitable for X*) iff

- (1)  $x$  is greater than any number previously or currently used as a witness for  $X$  at  $\alpha$  or any predecessor of  $\alpha$ ,
- (2)  $x$  is greater than  $|\alpha|$  and any stage  $s \leq t$  at which any  $\beta \succ \alpha$  has changed a set, picked a witness, or extended an enumeration operator,
- (3)  $x$  is in  $S_X(\alpha)$  and has not been dumped into  $X$ , and

- (4)  $x$  is greater than  $s$  many numbers satisfying (1)-(3), where  $s$  is the most recent stage at which  $\alpha$  was initialized, or  $\alpha$  previously picked a suitable number.

We have added requirement (1) in order to have the sequence of witnesses at a node be an increasing sequence (this might not be required, but certainly simplifies the picture). Also, this ensures that no two requirements are working with the same witness (in the construction, whenever a number is chosen, we ensure that it is suitable for the set for which it is chosen at the relevant node).

We have added requirement (2) to ensure that the work with this witness does not injure future requirements that are not initialized by this action.

We have added requirement (3) to ensure adding or removing  $x$  to or from  $X$  will not injure higher-priority requirements (it is the higher-priority strategies' responsibility to ensure that elements in the stream satisfy this).

We have added requirement (4) to pass on sufficiently many numbers for lower-priority strategies to work with. Also it ensures the stabilizing effect that we need, as described in Section 4.3.

Whenever a strategy asks for a  $t, \alpha$ -suitable number when it does not exist, finish the stage. This will iterate until a suitable number has been generated (or this node no longer becomes active); Lemma 6(3) (the Tree Lemma) shows that this wait is always finite.

In the construction we write "pick an  $x$  that is  $\alpha$ -suitable for  $A$  (or  $B$ , or  $C$ )". What this means is to pick an  $x$  that is  $t, \alpha$ -suitable for  $A$  (or  $B$ , or  $C$ ) where either  $t$  is the stage at which we started waiting for this suitable number, or, in case we are replacing a number that is no longer in the stream, the  $t$  that was used previously. Also all witnesses chosen by extensions in the time between picking the original witness and now are discarded (and dumped into their respective sets).

**5.3. Steps in the Construction.** The construction consists of infinitely many stages  $s \in \mathbb{N}$ . Stage  $s$  consists of substages  $t \leq s$ , although there are stages where one of the substages  $t$  *finishes* the stage; meaning the substages  $t'$  with  $t < t' \leq s$  are not performed. If  $t = s$ , then substage  $t$  always finishes the stage. During substage  $t$  of stage  $s$ , we build the current approximation  $f_{s,t}$  to the true path  $f_s$  by recursion: Either  $t$  finishes the construction, in which case  $f_s = f_{s,t}$ , or it gives an outcome  $o \in \text{Outc}$  and then sets  $f_{s,t+1} = f_{s,t} \hat{\ } o$ .

Initializing a strategy means making all local parameters (witnesses, killing points, coding points, streams) undefined.

At stage  $s$  we set  $S_A(\emptyset)$  and  $S_B(\emptyset)$  to  $[0, s)$ . (Without change to the construction we could have chosen  $S_A(\emptyset) = S_B(\emptyset) = \mathbb{N}$  for all stages.)

At substage  $t$  of stage  $s$ , we have defined  $\alpha := f_{s,t}$ . Depending on which requirement  $R$  is assigned to  $\alpha$ , we perform the following actions.

**Case**  $R = \mathsf{T}_{A,B,\Phi}$  ( $A \neq \Phi^B$ ): Let  $\beta_0 \prec \cdots \prec \beta_i \prec \alpha$  be all the predecessors of  $\alpha$  such that an  $\mathsf{S}_{A,B,\Phi}$ -requirement is active at  $\alpha$  via  $\beta_j$  (write  $\mathsf{S}_{A,B,\Phi}^j$  for that requirement).

First perform step  $(0_{\mathsf{T}_{A,B,\Phi}})$ :

- $(0_{\mathsf{T}_{A,B,\Phi}})$  If the least number,  $a_{\text{least}}$ , that has changed its membership status w.r.t.  $A$  since the last time at which  $\alpha$  was active is less than  $\text{wit}$ , the current witness, then
- remove the current killing point from  $B$ , and
  - remove all elements greater than  $a_{\text{least}}$  from  $S_A(\alpha \frown O)$  for  $O \in \text{Outc}$ , and
  - remove all elements greater than  $a_{\text{least}}$  from  $W_\alpha$ , and
  - make  $\text{wit}$  undefined.

Then perform the first step from  $(1_{\mathsf{T}_{A,B,\Phi}})$ — $(5_{\mathsf{T}_{A,B,\Phi}})$  that applies:

- $(1_{\mathsf{T}_{A,B,\Phi}})$  If, for some  $\beta_j$ ,  $\alpha$  does not have a current killing point or the current killing point is no longer in  $S_B(\alpha)$ , let  $j$  be least for which this is true, pick a killing point  $\text{kill}_j$  for  $\beta_j$  suitable for  $B$ , enumerate it into  $B$ , and finish the stage.
- $(2_{\mathsf{T}_{A,B,\Phi}})$  If there is no current witness, or the current witness is no longer in  $S_A(\alpha)$ , pick a witness,  $\text{wit}$ ,  $\alpha$ -suitable for  $A$ , put it into  $A$ , and finish the stage.
- $(3_{\mathsf{T}_{A,B,\Phi}})$  If the current witness satisfies  $\text{wit} \in A \setminus \Phi^B$ , do nothing other than setting the  $\alpha$ -outcome to  $\text{wait}$ , and setting  $S_A(\alpha \frown \text{wait}) = [s_0, s) \cap S_A(\alpha)$  and  $S_B(\alpha \frown \text{wait}) = [s_0, s) \cap S_B(\alpha)$ , where  $s_0$  is the oldest stage such that since then  $\alpha$  always has had outcome  $\text{wait}$ .
- $(4_{\mathsf{T}_{A,B,\Phi}})$  If the current witness satisfies that  $\text{wit} \in \Phi^B \setminus A$ , do nothing other than setting the  $\alpha$ -outcome to  $\text{stop}$ , and setting  $S_A(\alpha \frown \text{stop}) = [s_0, s) \cap S_A(\alpha)$  and  $S_B(\alpha \frown \text{stop}) = [s_0, s) \cap S_B(\alpha)$ , where  $s_0$  is the oldest stage such that since then  $\alpha$  always has had outcome  $\text{stop}$ .
- $(5_{\mathsf{T}_{A,B,\Phi}})$  If the current witness satisfies  $\text{wit} \in A \cap \Phi^B$ , add  $\text{wit}$  to  $W_\alpha$ , the set of *realized* witnesses, set  $s_{\text{wit}}$  to the current stage

(this makes  $n \mapsto s_n$  a mapping defined on  $W_\alpha$  giving for every realized witness the stage at which it was realized), and unset  $\text{wit}$  as the current witness. Also perform the following steps (here we try to satisfy the  $\Gamma$ -requirement, but towards that goal we do work that results—if we don't get a fully  $\Gamma$ -cleared number—in satisfying the highest-priority  $\mathcal{S}_{A,B,\Phi}^j$ -requirement we can, injuring the ones between  $\alpha$  and that  $\mathcal{S}_{A,B,\Phi}^j$ -requirement). Here we have defined some  $a_j$  for  $j \in [0, i]$  to be the largest number for which we have taken a step in the satisfaction of  $\mathcal{S}_{A,B,\Phi}^j$  (see  $(5_{\Gamma_{A,B,\Phi}} \text{d})$  below). Initially none of these are defined. We say  $a \in W_\alpha$  is  $j$ - $\Gamma$ -cleared iff

$$(\forall k. j \leq k \leq i \rightarrow \Gamma_k^B[s_a] \subseteq \Phi_k^A) \quad \wedge \quad a \notin A$$

and

$$\forall k. 0 \leq k \leq j \rightarrow \text{if } a_k \text{ is defined, then } a > a_k.$$

Find the least  $j$  with  $-1 \leq j \leq i$  such that there is an  $a \in W_\alpha$  that is  $(j+1)$ - $\Gamma$ -cleared. Let  $\tilde{a}$  be the least such  $a$ . If  $j = -1$ , then we have found an element of  $A$  that is fully  $\Gamma$ -cleared, and we can perform the diagonalization. We enumerate  $B[s_{\tilde{a}}]$  into  $B$ , and **stop**  $\alpha$ : We initialize all extensions of  $\alpha \hat{\ } \text{stop}$  and finish the stage (note that this means that if this requirement is not injured, then the next time it is active, step  $(4_{\Gamma_{A,B,\Phi}})$  applies).

Otherwise,

- (a) extract all  $\text{kill}_k$  with  $k \in [j, i]$  from  $B$ , and make the killing points  $\text{kill}_k$  for  $k \in [j, i]$  undefined,
- (b) enumerate  $\langle \tilde{a}, \Gamma_j^B[s] \rangle$  into  $\Delta_j$ ,
- (c) add  $\tilde{a}$  to  $S_A(\alpha \hat{\ } \infty_j)$ , and remove all  $a > \tilde{a}$  from  $S_A(\alpha \hat{\ } \infty_j)$ . Then enumerate all  $a > \tilde{a}$  that have  $a \in W_\alpha$  into  $A$  and remove them from  $W_\alpha$  (i.e. we dump these into  $A$ ),
- (d) set  $a_j$  to  $\tilde{a}$ ,
- (e) set the outcome of  $\alpha$  to  $\infty_j$ , and
- (f) set  $S_B(\alpha \hat{\ } \infty_j)$  to be  $S_B(\alpha) \cap [s_0, \infty)$ , where  $s_0$  is the first stage it was active since  $\alpha$  was initialized, or had outcome  $< \infty_j$ .

**Case**  $R = \mathcal{S}_{A,B,\Omega}$  ( $\exists \Gamma. \Omega^A = \Gamma^B$ )  $\vee$  ( $\exists \Delta. A = \Delta^{(\Omega^A)}$ ): We work at building a correct  $\Gamma$  in this case. Let  $F$  be the set of all

current killing points for  $B$  defined at extensions of  $\alpha$ . Then perform all of the following steps until you finish, or finish the stage:

- (1) If there are any coding points  $b_x$  that have been moved in and out of  $B$ ,  $b_x$  many times, then remove them from  $B$  and choose an  $\alpha$ -suitable number  $b'_x$  for  $B$  to put in their place: Enumerate it into  $B$  and enumerate axioms  $\langle x, F \cup \{b'_x\} \rangle$  into  $\Gamma$  for any axiom  $\langle x, F \cup \{b_x\} \rangle$  in  $\Gamma$ .
  - (2) If there are any coding points  $b_x$  no longer in the  $B$ -stream, for the least  $x$  where this is true pick a new coding point  $b_x$  in the  $B$ -stream.
  - (3) For each  $x \in \Omega^A$  that in all of its axioms  $\langle x, F' \rangle$  has a removed killing point in  $F'$ , enumerate a new axiom  $\langle x, F \cup \{b_x\} \rangle$  where  $b_x$  is a new coding point for  $x$  suitable for  $B$ .
  - (4) For all  $x \in \Gamma^B \setminus \Omega^A$  that have a coding point, remove the coding point from  $B$ , and finish the stage.
  - (5) For all  $x \in \Omega^A \setminus \Gamma^B$  that have a coding point, add the coding point to  $B$ , and finish the stage. (Note that when these two steps do not finish the stage,  $\Gamma^B$  and  $\Omega^A$  agree on all points that have coding points)
  - (6) Pick, if it exists, the least  $x \in \Omega^A$  that does not have a coding point, choose a number  $b_x \in B$  suitable for  $B$  as its coding point and enumerate  $\langle x, \tilde{F} \cup \{b_x\} \rangle$  into  $\Gamma$ , where  $\tilde{F}$  is the set of points in  $F$  from nodes  $\beta$  with  $|\beta| < b_x$ .
  - (7) Set  $S_A(\alpha \frown \text{so}) = [s_0, s) \cap S_A(\alpha)$  and  $S_B(\alpha \frown \text{so}) = [s_0, s) \cap S_B(\alpha)$ , where  $s_0$  is the last stage at which  $\alpha$  was initialized.
- Case**  $R = \top_{B,C,\Phi}$  ( $B \neq \Phi^C$ ): Let  $\beta_0 \prec \dots \prec \beta_i \prec \alpha$  be all the predecessors of  $\alpha$  such that an  $S_{B,C,\Phi}$ -requirement is active at  $\alpha$  via  $\beta_j$  (write  $S_{B,C,\Phi}^j$  for that requirement).

First perform step  $(0_{\top_{B,C,\Phi}})$ :

- $(0_{\top_{B,C,\Phi}})$  If the least number,  $b_{\text{least}}$ , that has changed its membership status w.r.t.  $B$  since the last time  $\alpha$  was active is less than  $b_c$ , the current witness, then
- remove the current killing point from  $C$ , and
  - remove all elements greater than  $b_{\text{least}}$  from  $S_B(\alpha \frown O)$  for  $O \in \text{Outc}$ , and
  - remove all elements greater than  $b_{\text{least}}$  from  $W_\alpha$ , and
  - make wit undefined.

Then perform the first step from  $(1_{\mathbb{T}_{B,C,\Phi}})$ — $(5_{\mathbb{T}_{B,C,\Phi}})$  that applies:

- ( $1_{\mathbb{T}_{B,C,\Phi}}$ ) If, for some  $\beta_j$ ,  $\alpha$  does not have a current killing point, pick a killing point  $\text{kill}_j$  for  $\beta_j$  suitable for  $C$ , enumerate it into  $C$ , and finish the stage (finishing the stage allows higher-priority requirements, requirements assigned to  $\beta \prec \alpha$ , to possibly correct enumerations).
- ( $2_{\mathbb{T}_{B,C,\Phi}}$ ) If there is no current witness, or the current witness is no longer in  $S_B(\alpha)$ , pick a witness,  $\text{wit}$ ,  $\alpha$ -suitable for  $B$ , put it into  $B$ , and finish the stage.
- ( $3_{\mathbb{T}_{B,C,\Phi}}$ ) If the current witness satisfies  $\text{wit} \in B \setminus \Phi^C$ , do nothing other than setting the  $\alpha$ -outcome to **wait**, and setting  $S_A(\alpha \frown \text{wait}) = [s_0, s) \cap S_A(\alpha)$  and  $S_B(\alpha \frown \text{wait}) = [s_0, s) \cap S_B(\alpha)$ , where  $s_0$  is the oldest stage such that since then  $\alpha$  always has had outcome **wait**.
- ( $4_{\mathbb{T}_{B,C,\Phi}}$ ) If the current witness satisfies that  $\text{wit} \in \Phi^C \setminus B$ , do nothing other than setting the  $\alpha$ -outcome to **stop**, and setting  $S_A(\alpha \frown \text{stop}) = [s_0, s) \cap S_A(\alpha)$  and  $S_B(\alpha \frown \text{stop}) = [s_0, s) \cap S_B(\alpha)$ , where  $s_0$  is the oldest stage such that since then  $\alpha$  always has had outcome **stop**.
- ( $5_{\mathbb{T}_{B,C,\Phi}}$ ) If the current witness satisfies  $\text{wit} \in B \cap \Phi^C$ , add  $\text{wit}$  to  $W_\alpha$ , the set of *realized* witnesses, set  $s_{\text{wit}}$  to the current stage (this makes  $n \mapsto s_n$  a mapping defined on  $W_\alpha$  giving for every realized witness the stage at which it was realized), and unset  $\text{wit}$  as the current witness. Also perform the following steps (here we try to satisfy the  $\mathbb{T}$ -requirement, but towards that goal we do work that results—if we don't get a fully  $\Gamma$ -cleared number—in satisfying the highest-priority  $\mathbb{S}_{B,C,\Phi}^j$ -requirement we can, injuring the ones between  $\alpha$  and that  $\mathbb{S}_{B,C,\Phi}^j$ -requirement). Here we have defined some  $b_j$  for  $j \in [0, i]$  to be the largest number for which we have taken a step in the satisfaction of  $\mathbb{S}_{B,C,\Phi}^j$  (see ( $5_{\mathbb{T}_{B,C,\Phi}}$  d) below). Initially none of these are defined. We say  $b \in W_\alpha$  is  $j$ - $\Gamma$ -cleared iff

$$(\forall k. j \leq k \leq i \rightarrow \Gamma_k^C[s_b] \subseteq \Phi_k^B) \quad \wedge \quad b \notin B$$

and

$$\forall k. j \leq k \leq i \rightarrow \text{if } b_k \text{ is defined, then } b > b_k.$$

Find the least  $j$  with  $-1 \leq j \leq i$  such that there is a  $b \in W_\alpha$  that is  $(j+1)$ - $\Gamma$ -cleared that has not previously been selected for that  $j$  in this step. Let  $\tilde{b}$  be the least such  $b$ .

If  $j = -1$ , then we have found an element of  $B$  that is fully  $\Gamma$ -cleared, and we can perform the diagonalization. We enumerate  $C[s_{\tilde{b}}]$  into  $C$ , and **stop**  $\alpha$ : We initialize all extensions of  $\alpha \frown \text{stop}$  and finish the stage (note that this means that if this requirement is not injured, then the next time it is active, step  $(4_{\top_{B,C,\Phi}})$  applies).

Otherwise,

- (a) extract all  $\text{kill}_k$  with  $k \in [j, i]$  from  $C$ , and make the killing points  $\text{kill}_k$  for  $k \in [j, i]$  undefined,
- (b) enumerate  $\langle \tilde{b}, \Gamma_j^C[s] \rangle$  into  $\Delta_j$ ,
- (c) add  $\tilde{b}$  to  $S_B(\alpha \frown \infty_j)$ , and remove all  $b > \tilde{b}$  from  $S_B(\alpha \frown \infty_j)$ . Then enumerate all  $b > \tilde{b}$  that have  $b \in W_\alpha$  into  $B$  and remove them from  $W_\alpha$  (i.e. we dump these into  $B$ ),
- (d) set  $b_j$  to  $\tilde{b}$ ,
- (e) set the outcome of  $\alpha$  to  $\infty_j$ , and
- (f) set  $S_A(\alpha \frown \infty_j)$  to be  $S_A(\alpha) \cap [s_0, \infty)$ , where  $s_0$  is the first stage it was active since  $\alpha$  was initialized, or had outcome  $< \infty_j$ .

**Case**  $R = S_{B,C,\Omega}$  ( $\exists \Gamma. \Omega^B = \Gamma^C$ )  $\vee$  ( $\exists \Delta. B = \Delta^{(\Omega^B)}$ ): We work at building a correct  $\Gamma$  in this case. Let  $F$  be the set of all current killing points for  $C$ . Then perform all of the following steps:

- (1) If there are any coding points  $c_x$  that have been moved in and out of  $C$ ,  $c_x$  many times, then remove them from  $C$  and wait for an  $\alpha$ -suitable number  $c'_x$  for  $C$  to put in their place: Enumerate it into  $C$  and enumerate axioms  $\langle x, F \cup \{c'_x\} \rangle$  into  $\Gamma$  for any axiom  $\langle x, F \cup \{c_x\} \rangle$  in  $\Gamma$ .
- (2) For all  $x \in \Gamma^C \setminus \Omega^B$  that have a coding point, remove the coding point from  $C$ .
- (3) For all  $x \in \Omega^B \setminus \Gamma^C$  that have a coding point, add the coding point to  $C$ . (Note that after these two steps,  $\Gamma^C$  and  $\Omega^B$  agree on all points that have coding points).
- (4) Pick the least  $x \in \Omega^B$  that does not have a coding point, choose a number  $c_x \in C$  suitable for  $C$  as its coding point and enumerate  $\langle x, F \cup \{c_x\} \rangle$  into  $\Gamma$ .

- (5) Set  $S_A(\alpha \smallfrown \mathbf{so}) = [s_0, s) \cap S_A(\alpha)$  and  $S_B(\alpha \smallfrown \mathbf{so}) = [s_0, s) \cap S_B(\alpha)$ , where  $s_0$  is the last stage at which  $\alpha$  was initialized.

Finally, initialize anything to the right of the current approximation of the true path, also if the stage was stopped by  $\alpha$  in step  $(\mathfrak{5}_{\mathbb{T}_{A,B,\Phi}})$  or  $(\mathfrak{5}_{\mathbb{T}_{B,C,\Phi}})$ , then initialize all extensions of  $\alpha$ .

## 6. VERIFICATION OF THE AHMAD TRIPLE CONSTRUCTION

We define the true path of the construction,  $f : \mathbb{N} \rightarrow \text{Outc}$ , by recursion

$$f(n) = \liminf_{\{s \in \mathbb{N}: f \smallfrown n \prec f_s \wedge |f_s| > n\}} f_s(n)$$

**Lemma 6** (Tree Lemma).

- (1) *Each  $\alpha \prec f$  is initialized finitely often, and if a  $\mathbb{T}$ -requirement is assigned to  $\alpha$ , it attains outcome **stop** through step  $(\mathfrak{5}_{\mathbb{T}_{A,B,\Phi}})$  or  $(\mathfrak{5}_{\mathbb{T}_{B,C,\Phi}})$  at most finitely often.*
- (2) *For each strategy  $\alpha \prec f$  there are infinitely many  $x$  such that*

$$\exists t \forall s \geq t. x \in S_A(\alpha)$$

*(i.e.,  $x$  is eventually in  $S_A(\alpha)$ ), and there are infinitely many  $x$  such that*

$$\exists t \forall s \geq t. x \in S_B(\alpha)$$

*(i.e.,  $x$  is eventually in  $S_B(\alpha)$ ).*

- (3) *For every  $\alpha \prec f$ ,  $X \in \{A, B\}$ , and any  $s$ , if  $\alpha$  needs a number  $r$ -suitable for  $X$  at stage  $s$ , then there is an  $\alpha$ -stage  $t \geq s$  and a number  $z > s$  such that  $z$  is  $r$ -suitable for  $X$  at  $\alpha$  and stage  $t$ .  
Moreover, there exists such  $z > s$  (a  $z$  that is  $r$ -suitable for  $X$  at  $\alpha$  and stage  $t$ ) that remains in  $S_X(\alpha)$  (i.e.,  $\forall t' \geq t. x \in S_X(\alpha)$ ).*
- (4)  *$f$  is total.*
- (5) *The status of all requirements reaches a limit along  $f$ : Every requirement is either satisfied along sufficiently long initial segments of  $f$ , or is active along sufficiently long initial segments of  $f$ .*

Note that for (1), the second part follows from the first; this overloading is needed for the induction.

*Proof.* We prove (1)–(4) by simultaneous induction.

Base step: First we observe they are all true at the root:

For (1): Note that a strategy  $\alpha$  is only initialized at the start of the construction, and after that only either when it is to the right of the current approximation to the true path (to the right of  $f_s$ ), or when some strategy  $\beta \prec \alpha$  has outcome **stop** by step  $(5_{\Gamma_{A,B,\Phi}})$  or  $(5_{\Gamma_{B,C,\Phi}})$ .

So the root is only initialized at the start of the construction. If it is a  $\Gamma$ -strategy, then the first time it has outcome **stop**, *all* other nodes are initialized and therefore cannot injure it; i.e., it will have outcome **stop** forever after.

For (2): This is true at the root since the root node is only initialized at the start of the construction, so at stage  $s$  we have  $S_X(\emptyset) = [0, s)$ .

For (3): Once the root node wants a suitable number, items (1), (2), and (4) in the definition of suitability represent fixed numbers. By having infinite and stable streams at the root, item (3) will eventually also be satisfied.

For (4): It is clear that  $f(0)$  will be defined.

Induction step: Now let  $\emptyset \prec \alpha$  be such that  $\alpha^- \prec f$  and (1)–(4) hold for  $\alpha^-$  and its predecessors. Observe that by this assumption,  $f(|\alpha| - 1)$  is defined ( $\alpha^-$  never is stuck permanently waiting for a suitable number), i.e., the induction step for (4) is satisfied. This means that for (1)–(3), we can assume that also  $\alpha \prec f$ .

For (2): Here we know that both  $S_A(\alpha^-)$  and  $S_B(\alpha^-)$  are infinite; in fact, their stable parts are infinite. Also, using (1), we have that  $s_0$  is the last stage at which  $\alpha^-$  is initialized. In all cases other than  $\alpha = (\alpha^-) \frown \infty_i$ , we have for both  $X = A$  and  $X = B$  that  $S_X(\alpha) = S_X(\alpha^-) \cap [s_0, s)$ , proving the claim.

It remains to cover the case  $\alpha = (\alpha^-) \frown \infty_i$ , i.e.,  $\alpha^-$  is assigned a  $\Gamma$ -requirement that never finds a fully  $\Gamma$ -cleared witness, but infinitely often finds a partially cleared witness (up to  $i$ , for least  $i$ ). We will first prove the result for a  $\Gamma_{A,B,\bullet}$ -requirement.

The result is now immediate for  $S_B(\alpha)$ , since  $S_B(\alpha) = [s_0, \infty) \cap S_B(\alpha^-)$ . So let  $SP_A(\alpha)$  be the set of elements permanently in  $S_A(\alpha)$ . We'll show by induction that  $SP_A := SP_A(\alpha)$  is infinite. Assume  $\{x_0 < \dots < x_{n-1}\}$  are the  $n$  least elements in  $SP_A$ . By the definition of suitability, all  $x_i$  ( $i < n$ ) can only be used as witnesses at finitely many nodes (since witnesses  $x$  at a node  $\alpha$  need to satisfy  $x > |\alpha|$ ). This means that their membership status can only change finitely often (a witness at a node is—by that node—inserted into the set and then removed from that set). So there is some stage  $s > s_0$  after which no element in  $\{x_0, \dots, x_{n-1}\}$  changes its membership status in  $A$ .

Look for the least stage  $s' > s$  for which there is a number  $x'$  in  $(S_A(\alpha) \setminus \{x_0, \dots, x_{n-1}\}) \cap SP_A(\alpha^-)$  at a stage  $s'$ . Note that  $(s', x')$

exists since, by induction, we are never stuck waiting for a suitable number. Now  $x' \in SP_A(\alpha)$  since it will not be removed from the stream by  $\alpha^-$  being initialized, or by it being removed from  $S_A(\alpha^-)$ , and it will not be removed by the action of  $\alpha$  since it will never be more than  $i$ -cleared, and therefore only works with numbers greater than  $x'$ .

Now for a  $\mathbb{T}_{B,C,\bullet}$ -requirement: The result is immediate for  $S_A(\alpha)$ . Also, the reasoning above up to the last sentence works with the additional observation that coding points also can only change their membership status finitely often. Now, however, a number can be removed from the stream by a  $B$ -restore from a lower-priority requirement (a  $B$ -restore is the action of enumerating all  $b$  that were in  $B$  at an earlier stage into  $B$  at this stage; restoring their membership to  $B$ ; this happens in  $(5_{\mathbb{T}_{A,B,\Phi}})$ ).

When the  $B$ -restore happens,  $x'$  and all larger elements are removed from the stream. The first number  $y$  that appears in the stream after this is larger than  $x'$  and has the property that the only  $B$ -restores that could injure it are ones for witnesses that were already realized before the  $B$ -restore mentioned above. This shows that this can happen only finitely many times, after which the first number  $y$  that appears in the stream will never be removed by further  $B$ -restorations, showing that it remains in the stream forever.

For (1): Let  $s$  be the minimal stage after which  $\alpha$  will never be to the right of the true path, and after which no  $\beta \prec \alpha$  is initialized or reaches outcome **stop** by step  $(5_{\mathbb{T}_{A,B,\Phi}})$  or  $(5_{\mathbb{T}_{B,C,\Phi}})$  (which we can assume by induction). Then after stage  $s$ ,  $\alpha$  is never initialized again, but  $\alpha$  is initialized at stage  $s$ .

We need to check that  $\alpha$  attains outcome **stop** only finitely often through step  $(5_{\mathbb{T}_{A,B,\Phi}})$  or  $(5_{\mathbb{T}_{B,C,\Phi}})$ . If  $\alpha$  is not a  $\mathbb{T}$ -strategy, this is automatic, so let it be a  $\mathbb{T}$ -strategy. If it never reaches outcome **stop** after stage  $s$ , then we are done, so assume it reaches outcome **stop** for the first time at stage  $s' > s$ . We want to check it never reaches outcome **stop** because of step  $(5_{\mathbb{T}_{A,B,\Phi}})$  or  $(5_{\mathbb{T}_{B,C,\Phi}})$  again.

At stage  $s'$ , all nodes to the right of  $\alpha$  and all extensions of  $\alpha$  are initialized; that is, all nodes that will be active at some point in the future—other than  $\alpha$  and its predecessors—are initialized.

Since none of the predecessors of  $\alpha$  will change membership of any element smaller than the current witness of any set (otherwise  $\alpha$  would be initialized; note that since any predecessors of  $\alpha$  that have current witnesses have outcome **wait** or **stop**) and by the definition of suitability, all new witnesses will be larger than the current witness, so the current

witness is permanently in the stream, and thus the outcome of  $\alpha$  is permanently **stop** by  $(4_{\mathbb{T}_{A,B,\Phi}})$  or  $(4_{\mathbb{T}_{B,C,\Phi}})$ .

For (3): Use induction and note that once a strategy starts waiting for an  $r$ -suitable number, (1), (2), and (4) of the definition of suitability represent fixed numbers.

Since any time we choose an  $r$ -suitable number, we choose again when it is removed from the stream, we can restrict our attention to numbers permanently in the stream. By item (4), sufficiently many numbers will be in the stream that have not been used by higher-priority strategies.

Now (2) of this lemma gives that we will get a permanently  $r$ -suitable number eventually.

For (5): This follows immediately from (5) and the fact that in the construction only nodes of  $\mathbb{T}$  to which requirements are assigned are used (i.e., only valid branches can occur).  $\square$

The following lemma is an important observation, in particular for  $A$  and  $B$ . Almost every time  $A$  and  $B$  change somewhere, the streams above this change are truncated (almost every time, since for instance the stream at the root is not so influenced). If initial segments would not stabilize, then many requirements would not have permanent witnesses.

**Lemma 7.**  *$A$ ,  $B$ , and  $C$  are  $\Delta_2^0$ -sets.*

The proof of this lemma for  $A$  and  $B$  is included in the proof of Lemma 6 (the Tree Lemma) above.  $C$  only contains coding points, and those by construction can only change membership finitely often.

**Lemma 8.** *Every  $\mathbb{T}_{A,B,\Phi}$ -requirement is satisfied.*

*Proof.* By Lemma 6(5) (the Tree Lemma) and the observation we made right after the assignment of requirements to the tree, there exists a least  $\alpha \prec f$  such that along all  $\gamma$  with  $\alpha \preceq \gamma \prec f$ , we have that  $\mathbb{T}_{A,B,\Phi}$  has label satisfied. This means that the node where  $\mathbb{T}_{A,B,\Phi}$  is assigned is an initial segment  $\alpha'$  of  $\alpha$  and has either  $(\alpha' \widehat{\text{stop}}) \prec f$  or  $(\alpha' \widehat{\text{wait}}) \prec f$ .

Using Lemma 6(1) and (3) (the Tree Lemma), we go past the stage where  $\alpha'$  is initialized and where it has a stable witness  $\text{wit}$  (one that is not removed from its stream). We see that in case  $(\alpha' \widehat{\text{stop}}) \prec f$ , we have that  $\text{wit} \in \Phi^B \setminus A$ , and in case  $(\alpha' \widehat{\text{wait}}) \prec f$ , we have that  $\text{wit} \in A \setminus \Phi^B$ ; in either case, the requirement is satisfied.  $\square$

**Lemma 9.** *Every  $\mathbb{T}_{B,C,\Phi}$ -requirement is satisfied.*

The proof is analogous to the proof of Lemma 8.

**Lemma 10.** *Every  $\mathsf{S}_{A,B,\Phi}$ -requirement is satisfied.*

*Proof.* By Lemma 6(5) (the Tree Lemma), there is a least  $\beta \prec f$  such that for all  $\gamma \prec f$  extending  $\beta$ , either

- (1)  $\mathsf{S}_{A,B,\Phi}$  is active at  $\gamma$  via  $\beta$ , or
- (2)  $\mathsf{S}_{A,B,\Phi}$  is satisfied at  $\gamma$  via  $\beta$ .

Case (1): This means  $\beta$  is assigned requirement  $\mathsf{S}_{A,B,\Phi}$  and so performs the steps as described in Case  $R = \mathsf{S}_{A,B,\Omega}$  of the construction. We will show that the  $\Gamma$  constructed satisfies

$$\Phi^A = \Gamma^B.$$

As  $A$  is  $\Delta_2^0$  (from Lemma 7), any initial segment of  $A$  stabilizes—in particular the membership status of any fixed element  $x$  of  $A$  stabilizes. This means that eventually  $x$  has a fixed coding number  $b_x$  (if it fluctuates often it might lose a coding number a couple of times, also a few times coding numbers might be removed from the stream; but by observations above, there is a stage after which neither of those happen).

Since  $\mathsf{S}_{A,B,\Phi}$  is active at all extensions on the true path, eventually the last killing point that will be removed by a  $\gamma$  extending  $\beta$  with  $|\gamma| < b_x$  is removed. After that stage an axiom with correct assumptions about the killing points will be enumerated for  $x$  (case (3) of case  $R = \mathsf{S}_{A,B,\Omega}$ ), and the membership of  $b_x$  in  $B$  corrected (either case 4 or case 5 of case  $R = \mathsf{S}_{A,B,\Omega}$ ).

Case (2): This means  $\beta$  is assigned a T-requirement and has outcome  $\infty_i$  where the  $i$ -th  $\mathsf{S}$ -requirement above it is  $\mathsf{S}_{A,B,\Phi}$ . We will show that the  $\Delta_i$  it constructs satisfies

$$A =^* \Delta_i^{(\Phi^A)}.$$

Since the equality is clearly true at elements dumped into  $A$  we only have to consider elements permanently in the  $A$ -stream. Let  $a$  be a witness chosen at a stage after which  $\beta$  will never have an outcome  $\infty_j$  with  $j < i$  or **stop**, and  $a$  is permanently in the  $A$ -stream of  $\beta \frown \infty_i$ . As elements in the stream are removed from the stream when there are smaller  $A$ -changes, we know there will be no  $A$ -changes smaller than  $a$ . Also by the definition of suitability (and our action when an earlier chosen suitable number is removed from the stream), there are no changes in  $A \upharpoonright m$  where  $m = \max \cup \{F \mid \exists x. \langle x, F \rangle \in \Omega_s \wedge F \subseteq A[s]\}$  (the part of  $A$  used in  $\Omega^A[s]$ ). At the stage  $s$  when  $a$  is added to the  $A$ -stream of  $\beta \frown \infty_i$ , we also enumerate an axiom  $\langle a, \Gamma^B[s] \rangle$  into  $\Delta_i$ . We need to verify that the equivalence

$$a \in A \quad \Leftrightarrow \quad \Gamma^B[s] \subseteq \Omega^A$$

holds.

If ever  $a \notin A$  and  $\Gamma^B[s] \subseteq \Omega^A$  holds then  $a$  is  $i$ -cleared in addition to being  $k$ -cleared for  $k > i$  (this in the context of  $R = \mathbb{T}_{A,B,\bullet}$ ), contradicting that  $\beta$  will not have outcomes  $\infty_j$  with  $j < i$  or outcome **stop** after choosing witness  $a$ .

For the other implication, at stage  $s$  we have  $\Gamma^B[s] \subseteq \Omega^A$ . Since we had already observed that there are no  $A$ -changes below  $a$ , we have that whenever  $a \in A$  we have that  $\Omega^A[s] \subseteq \Omega^A$ , which gives the other implication.  $\square$

**Lemma 11.** *Every  $S_{B,C,\Phi}$ -requirement is satisfied.*

The proof is analogous to the proof of Lemma 10.

This completes the verification, and therefore the proof of our main theorem.

## REFERENCES

- [1] Seema Ahmad. *Some results on the structure of the  $\Sigma_2$  enumeration degrees*. PhD thesis, Simon Fraser University (Canada), 1989.
- [2] Seema Ahmad and Alistair H. Lachlan. Some special pairs of  $\Sigma_2$  e-degrees. *Math. Log. Q.*, 44(4):431–449, 1998.
- [3] S. Barry Cooper. Partial degrees and the density problem. II. The enumeration degrees of the  $\Sigma_2$  sets are dense. *J. Symbolic Logic*, 49:503–513, 1984.
- [4] Thomas F. Kent. The  $\Pi_3$ -theory of the  $\Sigma_2^0$ -enumeration degrees is undecidable. *J. Symbolic Logic*, 71(4):1284–1302, 2006.
- [5] Steffen Lempp, Theodore A. Slaman, and Andrea Sorbi. On extensions of embeddings into the enumeration degrees of the  $\Sigma_2^0$ -sets. *J. Math. Log.*, 5(2):247–298, 2005.
- [6] Steffen Lempp and Andrea Sorbi. Embedding finite lattices into the  $\Sigma_2^0$  enumeration degrees. *J. Symbolic Logic*, 67(1):69–90, 2002.
- [7] Theodore A. Slaman and Robert I. Soare. Extension of embeddings in the computably enumerable degrees. *Ann. of Math. (2)*, 154(1):1–43, 2001.
- [8] Theodore A. Slaman and W. Hugh Woodin. Definability in the enumeration degrees. *Arch. Math. Logic*, 36(4-5):255–267, 1997. Sacks Symposium (Cambridge, MA, 1993).

DEPARTMENT OF MATHEMATICS, KAZAN STATE UNIVERSITY, KAZAN 420008,  
RUSSIA

*E-mail address:* Iskander.Kalimullin@ksu.ru

*URL:* <http://www.ksu.ru/persons/5211.ru.html>

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF COLORADO, BOULDER, CO  
80309-0395, USA

*E-mail address:* bart.kastermans@colorado.edu

*URL:* <http://www.bartk.nl/>

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF WISCONSIN, MADISON, WI  
53706-1388, USA

*E-mail address:* lempp@math.wisc.edu

*URL:* <http://www.math.wisc.edu/~lempp>