

The d.r.e. degrees are not dense

S. BARRY COOPER
LEO HARRINGTON
ALISTAIR H. LACHLAN
STEFFEN LEMPP
ROBERT I. SOARE

Abstract. By constructing a maximal incomplete d.r.e. degree, the nondensity of the partial order of the d.r.e. degrees is established. An easy modification yields the nondensity of the n -r.e. degrees ($n \geq 2$) and of the ω -r.e. degrees.

1. INTRODUCTION

Call a set $D \subseteq \omega$ *d.r.e. (difference of r.e. sets)* iff there are r.e. (recursively enumerable) sets $A_1, A_2 \subseteq \omega$ such that $D = A_1 - A_2$.

Another way of defining these sets is by recursive approximations to their characteristic functions: A is r.e. iff there is a recursive function f such that for all x :

- (i) $A(x) = \lim_s f(x, s),$
- (ii) $f(x, 0) = 0,$
- (iii) $|\{s \mid f(x, s) \neq f(x, s + 1)\}| \leq 1.$

(That is to say, we can change our guess about the membership of x in A at most once, namely when we enumerate x into A .) Now a set D is d.r.e. iff it satisfies the same definition where (iii) is replaced by

- (iii') $|\{s \mid f(x, s) \neq f(x, s + 1)\}| \leq 2.$

This explains also why d.r.e. sets are sometimes called 2-r.e., and it naturally leads to the definition of an n -r.e. set E (for $n \in \omega$) and an ω -r.e. set F satisfying the same definition where (iii) is replaced by

- (iii'') $|\{s \mid f(x, s) \neq f(x, s + 1)\}| \leq n,$

The authors would like to thank T. Slaman for helpful discussions. The first author was partially supported by the University of Illinois at Chicago, the University of Chicago, and T. Slaman's Presidential Young Investigator Award DMS-8451748 during the first author's visit in 1985-86. The second author was partially supported by NSF Grant DMS-8910312. The third author was partially supported by NSERC (Canada) Grant A3040. The fourth author was partially supported by NSF Grants DMS-8701891 and DMS-8901529 and fellowships of the Deutsche Forschungsgemeinschaft and the Mathematical Sciences Research Institute. The fifth author was partially supported by NSF Grant DMS-8807389 and a grant of the Deutsche Forschungsgemeinschaft. The last two authors were also partially supported by NSF Grant INT-8722296.

and by

$$(iii''') \quad |\{s \mid f(x, s) \neq f(x, s + 1)\}| \leq x,$$

respectively. (Of course, the 1-r.e. sets are exactly the r.e. sets.)

The d.r.e. and n -r.e. sets were first introduced and studied by Putnam [12] and Ershov [6] as a natural extension of the concept of r.e. sets. (For an extensive survey, see Epstein, Haas, Kramer [5].) The d.r.e. degrees (Turing degrees of d.r.e. sets) were first studied by Cooper (early 1970's, unpublished) and Lachlan (1968, unpublished), who showed that there is a *properly d.r.e. degree* (a d.r.e. degree that does not contain an r.e. set), and that every nonrecursive d.r.e. degree bounds a nonrecursive r.e. degree, respectively. The latter result established the downward density of the d.r.e. degrees (by the Sacks Density Theorem).

The main interest in the n -r.e. and, most of all, the d.r.e. degrees, lies in comparing them to the r.e. degrees, a structure that is now well understood. A lot of work has been done in this direction in the past decade. One example is the extension of the Arslanov Completeness Criterion to n -r.e. (and even n -REA) degrees by Jockusch, Lerman, Soare, and Solovay [7]. One main question in this area is whether the partial orders of the r.e. degrees and the d.r.e. degrees (or n -r.e. degrees in general) are elementarily equivalent. This was first refuted by Arslanov, and later by Downey using a different argument.

THEOREM (Arslanov [1,2]). *For all $n \geq 2$, every nonrecursive n -r.e. degree \mathbf{d} cups to \mathbf{O}' in the n -r.e. degrees, i.e. there is an incomplete n -r.e. degree \mathbf{e} such that $\mathbf{d} \cup \mathbf{e} = \mathbf{O}'$.*

THEOREM (Downey [4]). *For all $n \geq 2$, there are nonrecursive n -r.e. degrees \mathbf{d} and \mathbf{e} such that $\mathbf{d} \cap \mathbf{e} = \mathbf{O}$ and $\mathbf{d} \cup \mathbf{e} = \mathbf{O}'$.*

Since both of these statements fail in the r.e. degrees by a result of Yates and Cooper (unpublished, see D. Miller [11]), and by the Lachlan Nondiamond Theorem [9], respectively, either of the above theorems establish the following

COROLLARY (Arslanov [1,2]). *For all $n \geq 2$, the partial orders of the r.e. degrees and of the n -r.e. degrees are not elementarily equivalent.*

A related issue is the comparison between n -r.e. and m -r.e. degrees for distinct n and m . We restate the following

OPEN QUESTION: Are the partial orders of the n -r.e. and the m -r.e. degrees (for distinct $n, m \geq 2$) elementarily equivalent?

2. THE THEOREMS

Probably the most fundamental result about the r.e. degrees is the Sacks Density Theorem [13], which states that for any two r.e. degrees $\mathbf{a} < \mathbf{b}$, there is an r.e. degree \mathbf{c} such that

$\mathbf{a} < \mathbf{c} < \mathbf{b}$. The question of whether the d.r.e. degrees are also dense therefore generated a lot of interest and conjectures among recursion theorists. Partial evidence for density was Lachlan's observation that they are downward dense, and the following "weak" density theorem:

THEOREM (Cooper, Lempp, Watson [3]). *For any r.e. degrees $\mathbf{a} < \mathbf{b}$, there is a properly d.r.e. degree \mathbf{d} such that $\mathbf{a} < \mathbf{d} < \mathbf{b}$.*

Furthermore, by an easy modification of the observation by Jockusch and Soare [8] that $\mathbf{O}^{(n)}$ is not a minimal cover for any $n > 0$, it can be seen that no nonrecursive n -REA and thus no nonrecursive n -r.e. degree can be a minimal cover in the Turing degrees.

However, by a proof that crystallized over several years in discussions between the authors, the density conjecture is refuted by our main theorem:

D.R.E. NONDENSITY THEOREM. *There is a maximal incomplete d.r.e. degree, i.e. a d.r.e. degree $\mathbf{d} < \mathbf{O}'$ such that there is no d.r.e. degree \mathbf{e} with $\mathbf{d} < \mathbf{e} < \mathbf{O}'$. Thus the partial order of the d.r.e. degrees is not dense.*

An easy modification yields the

n -R.E./ ω -R.E. NONDENSITY THEOREM. *There is an incomplete d.r.e. degree maximal in the n -r.e. degrees (for all $n \geq 2$) as well as in the ω -r.e. degrees. Thus the n -r.e. degrees (for $n \geq 2$) and the ω -r.e. degrees are not dense.*

(We prove this last theorem in Section 11.)

Our nondensity result and Downey's diamond theorem both obviously imply the following

COROLLARY (Downey [4]). *The partial orders of the r.e. degrees and the n -r.e. degrees (for $n \geq 2$) are not Σ_2 -elementarily equivalent (i.e. they do not satisfy the same Σ_2 -statements).*

Since the two structures both allow the embedding of any finite partial order, they are Σ_1 -equivalent. The following is therefore the remaining

OPEN QUESTION (Slaman): Do the r.e. degrees form a Σ_1 -substructure of the d.r.e. degrees?

This question arose through the corresponding negative answer about the r.e. and the Δ_2 -degrees given by Slaman [14].

Our notation is standard and generally follows Soare [15]. $X \upharpoonright [m, n]$ has the set-theoretic meaning, i.e. $X \upharpoonright (n + 1) - X \upharpoonright m$. The use (largest number *actually used* in a computation) of a functional Γ , Θ , etc. is denoted by the corresponding lower-case Greek letters γ , ϑ , etc. (Thus changing X at some $y \leq \gamma(x)$ will allow $\Gamma^X(x)$ to change, etc.) If the oracle is a join of two sets we assume the use is computed on the two sets separately, i.e. $\Gamma^{(X \oplus Y) \upharpoonright (\gamma(x)+1)}$ will mean $\Gamma^X \upharpoonright (\gamma(x)+1) \oplus Y \upharpoonright (\gamma(x)+1)$. All use functions are

assumed to be increasing in the argument and nondecreasing in the stage. When describing the construction, all parameters A , Γ , x , etc. are assumed to be taken at the current (sub)stage.

We assume familiarity with \mathbf{O}''' -arguments (see e.g. Soare [15]).

3. THE D.R.E. NONDENSITY THEOREM: THE REQUIREMENTS AND THE BASIC STRATEGIES

We have to construct a d.r.e. set $D <_T K$ such that there is no d.r.e. set U with $D <_T U <_T K$. This is ensured by the following two types of requirements:

For all d.r.e. sets U , we build a partial recursive functional $\Gamma = \Gamma_U$ satisfying:

$$(4) \quad \mathcal{S}_U : K = \Gamma^{D \oplus U} \text{ or } \exists \Delta (U = \Delta^D).$$

To ensure that D is incomplete, we also build an auxiliary r.e. set A and satisfy for all partial recursive functionals Θ :

$$(5) \quad \mathcal{R}_\Theta : A \neq \Theta^D.$$

The basic strategy for \mathcal{S}_U in isolation is to build $\Gamma^{D \oplus U}$, ensuring that it is total, and that it computes K correctly by enumerating $\gamma(w)$ into D when w enters K . The strategy will always insist on the correctness of $\Gamma^{D \oplus U}$ on its domain but it may allow $\Gamma^{D \oplus U}$ to be partial if some lower-priority strategy builds a partial recursive functional Δ such that $U = \Delta^D$. An \mathcal{S}_U -strategy itself always has only one outcome on the tree of strategies.

The basic strategy for \mathcal{R}_Θ in isolation is the one developed by Friedberg and Muchnik:

1. Pick an unused witness x and keep it out of A .
2. Wait for $\Theta^D(x) \downarrow = 0$.
3. Put x into A and restrain $D \upharpoonright (\vartheta(x) + 1)$.

While the strategies for the requirements in isolation are thus very simple, there are obviously severe conflicts between them. The \mathcal{S}_U -strategy threatens to make D complete while the \mathcal{R}_Θ -strategy would like to preserve initial segments of D . The main difficulty of the \mathbf{O}''' -priority argument for the D.r.e. Nondensity Theorem therefore lies in combining them. We will slowly lead up to the full construction, trying to explain the intuition behind each feature as we add more and more elements of the construction.

4. THE \mathcal{R}_Θ -REQUIREMENTS BELOW ONE \mathcal{S}_U -STRATEGY

The Intuition. The fundamental idea here is the following. Since an \mathcal{R}_{Θ_0} -strategy obviously cannot prevent a higher-priority \mathcal{S}_U -strategy from correcting $\Gamma^{D \oplus U}$, the former will instead try to clear its Θ_0^D -computation of Γ -uses (i.e. of values $\gamma(w)$ for almost all w). The \mathcal{R}_{Θ_0} -strategy picks a fixed number w_0 and allows the \mathcal{S}_U -strategy to initialize the \mathcal{R}_{Θ_0} -strategy whenever $K \upharpoonright w_0$ changes and $\Gamma^{D \oplus U} \upharpoonright w_0$ needs to be corrected (which

constitutes only finite injury to the \mathcal{R}_{Θ_0} -strategy). Whenever a computation $\Theta_0^D(x_0) \downarrow = 0$ appears but is threatened by $\gamma(w_0) \leq \vartheta_0(x_0)$ then the \mathcal{R}_{Θ_0} -strategy, instead of putting x_0 into A , will put the current value of $\gamma(w_0)$ into D and request that the new use $\gamma(w_0)$ be very big, at least greater than the current value $\vartheta_0(x_0)$. (This action is often called “capricious destruction” of the computation $\Theta_0^D(x_0)$ (that we would really like to preserve) and was first used by Lachlan in the proof of his Nonsplitting Theorem [10].) If this happens infinitely often then our attempt of “finitarily” diagonalizing $\Theta_0^D(x_0) \downarrow \neq A(x_0)$ fails. Furthermore $\Gamma^{D \oplus U}(w_0) \uparrow$ because $\gamma(w_0) \rightarrow \infty$, so the \mathcal{R}_{Θ_0} -strategy must assume responsibility to define and keep correct a functional $\Delta^D = U$ on larger and larger initial segments in order to satisfy \mathcal{S}_U . Now keeping Δ^D correct involves changing $D \uparrow (\delta(w) + 1)$ whenever $U(w)$ changes. So we still satisfy \mathcal{S}_U ; but what have we gained for an \mathcal{R}_{Θ_1} -strategy below the infinite outcome of the \mathcal{R}_{Θ_0} -strategy since the \mathcal{R}_{Θ_1} -strategy now has to deal with Δ -uses? The trick is that its Θ_1 -computation is only threatened by Δ -uses if $\delta(w) \leq \vartheta_1(x_1)$ and $U(w)$ changes later. But then the \mathcal{R}_{Θ_0} -strategy should not have put $\gamma(w_0)$ into D since $\gamma(w_0)$ could have been increased past $\vartheta_0(x_0)$ using the U -change alone. It is here that we need to use the fact that D is d.r.e. since we can now extract the old value $\gamma(w_0)$ from D , stop destroying $\Gamma^D(w_0)$ (and forcing $\gamma(w_0) \rightarrow \infty$), and stop correcting Δ . This lets \mathcal{S}_U be satisfied by Γ again, and \mathcal{R}_{Θ_0} is satisfied finitarily since $\gamma(w_0) > \vartheta_0(x_0)$.

An extra complication here is that U can change back and forth many times on w 's with $\delta(w) \leq \vartheta_1(x_1)$, so we have to ensure that D can change below the old value of $\gamma(w_0)$ more often. Technically, we will reserve an entire interval $B = [\gamma(w_0) - n, \gamma(w_0)]$ (for some n) solely for destroying and restoring $\Gamma^{D \oplus U}(w_0)$. Every time $\Gamma^{D \oplus U}(w_0)$ is destroyed, it will be redefined by the \mathcal{S}_U -strategy using not only larger $\min B$ but also a longer such “use block”. We will assume that $|B|$ (the length of the interval B) is less than the least element of the interval B . (Thus $\vartheta(x_0) < |B|$ implies $\vartheta(x_0) < \min B$.) Furthermore, the \mathcal{R}_{Θ_1} -strategy will ensure the satisfaction of \mathcal{R}_{Θ_0} at its own witness x_1 .

With this intuition in mind, we now describe the \mathcal{R}_{Θ} -strategies more precisely:

The \mathcal{R}_{Θ_0} -strategy. It works with a fixed fresh witness x_0 for Θ_0 and a fixed “killing point” w_0 for Γ and proceeds as follows (where i is the number of times that the \mathcal{R}_{Θ_0} -strategy has put $\gamma_0(w_0)$ into D so far):

1. Wait for $\Theta_0^D(x_0) \downarrow = 0$ and $\Theta_0^D \uparrow (i + 1) \downarrow = A \uparrow (i + 1)$. (The second clause slows down the \mathcal{R}_{Θ_0} -strategy.)
2. If $\min(\gamma(w_0)$ -use block) $> \vartheta_0(x_0)$ then put x_0 into A , restrain $D \uparrow (\vartheta_0(x_0) + 1)$, and stop.
3. Otherwise, define B_i to be the current $\gamma(w_0)$ -use block, put $\gamma(w_0)$ into D , request that the \mathcal{S}_U -strategy choose the new $\gamma(w_0)$ -use block “very big”, i.e. of length greater than any number mentioned so far in the construction, define $\Delta^D(i) = U(i)$, keep it defined and correct from now on (unless stopped), and go back to 1. (Notice that x_0 is not yet

put into A .)

The \mathcal{R}_{Θ_0} -strategy will be initialized whenever $K \upharpoonright w_0$ changes but the parameter w_0 remains unchanged then. (This causes only finite injury to the \mathcal{R}_{Θ_0} -strategy.)

The \mathcal{R}_{Θ_0} -strategy will continue as above unless it is stopped by some other strategy (as specified below).

Our \mathcal{R}_{Θ_0} -strategy can thus have either a finite or an infinite outcome, depending on whether it is eventually always in 1. or in 2., or whether it goes from 3. to 1. infinitely often. (Note that by the below, the finite outcome of the \mathcal{R}_{Θ_0} -strategy can also be achieved by being stopped by another strategy.)

An \mathcal{R}_{Θ} -strategy of lower priority. If the \mathcal{R}_{Θ} -strategy assumes a finite outcome of the \mathcal{R}_{Θ_0} -strategy then the \mathcal{R}_{Θ} -strategy proceeds exactly as the \mathcal{R}_{Θ_0} -strategy itself since it is in the same position as the \mathcal{R}_{Θ_0} -strategy once the latter has stopped. Of course, this \mathcal{R}_{Θ} -strategy then has to use a fixed “killing point” $w > w_0$ so that the finite outcomes of \mathcal{R}_{Θ} -strategies destroy $\Gamma^{D \oplus U}(w)$ only finitely often for fixed w .

The situation is different for an \mathcal{R}_{Θ_1} -strategy assuming the infinite outcome of the \mathcal{R}_{Θ_0} -strategy.

The \mathcal{R}_{Θ_1} -strategy. It works with a fixed fresh witness $x_1 > x_0$ for Θ_1 and Θ_0 , and at stages at which the \mathcal{R}_{Θ_0} -strategy enters 3., it proceeds as follows:

1. Wait for $\Theta_1^D(x_1) \downarrow = \Theta_0^D(x_1) \downarrow = 0$, and $|B_i| > \vartheta_1(x_1) + 3$ at some stage s_* . (Here B_i is the $\gamma(w_0)$ -use block now being used by the \mathcal{R}_{Θ_0} -strategy to destroy $\Gamma^{D \oplus U}(w_0)$. Recall $|B_i| < \min B_i$.)
2. Set $y_1 = \vartheta_1(x_1)$, $y_0 = \max\{\vartheta_1(x_1), \vartheta_0(x_1)\}$, set $i^* = i$, put x_1 into A , and restrain $D \upharpoonright (y_0 + 1)$ from lower-priority strategies. (We pause briefly to analyze the situation now when we say that the \mathcal{R}_{Θ_1} -strategy is *taking charge* of the \mathcal{R}_{Θ_0} -strategy. If $U \upharpoonright (y_1 + 1) = U_{s_*} \upharpoonright (y_1 + 1)$ then we can ensure $D \upharpoonright (y_1 + 1) = D_{s_*} \upharpoonright (y_1 + 1)$ without making Δ^D incorrect so \mathcal{R}_{Θ_1} is satisfied finitarily without disturbing the \mathcal{S}_U - or the \mathcal{R}_{Θ_0} -strategy. Otherwise, since $y_1 < |B_{i^*}|$ and hence $< \min B_{i^*}$, we have $U \upharpoonright (\gamma_{s_*}(w_0) + 1) \neq U_{s_*} \upharpoonright (\gamma_{s_*}(w_0) + 1)$, so the \mathcal{R}_{Θ_1} -strategy will restore $D \upharpoonright (y_0 + 1) = D_{s_*} \upharpoonright (y_0 + 1)$ (by removing from D any element from B_{i^*} and any $\delta(y)$ from after stage s_*), thus restore $\Theta_0^D(x_1) \downarrow = \Theta_0^D(x_1)[s_*]$, and prevent the \mathcal{R}_{Θ_0} -strategy from acting since \mathcal{R}_{Θ_0} is satisfied via x_1 . Notice that now, after taking charge, the \mathcal{R}_{Θ_1} -strategy will stop and restart the \mathcal{R}_{Θ_0} -strategy when necessary.)
3. From now on ensure (using $|B_{i^*}| > y_1 + 3$):
 - a. Whenever $U \upharpoonright (y_1 + 1) = U_{s_*} \upharpoonright (y_1 + 1)$ then let the \mathcal{R}_{Θ_0} -strategy act and have $D \cap B_{i^*} \neq D_{s_*} \cap B_{i^*}$ by enumerating an element into D if necessary. (\mathcal{R}_{Θ_1} is satisfied and Δ^D is correct.)
 - b. Otherwise, prevent the \mathcal{R}_{Θ_0} -strategy from acting and restore $D \upharpoonright (y_0 + 1) = D_{s_*} \upharpoonright (y_0 + 1)$ by possibly extracting elements from D . (\mathcal{R}_{Θ_0} is satisfied via the Γ -cleared

computation $\Theta_0^D(x_1)$.)

The possible combined outcomes. The possible outcomes of an \mathcal{R}_{Θ_0} - and an \mathcal{R}_{Θ_1} -strategy below one \mathcal{S}_U -strategy are now as follow:

- A. The \mathcal{R}_{Θ_1} -strategy never reaches 2.
 - a. The \mathcal{R}_{Θ_0} -strategy eventually stops at 2. or waits at 1. forever. Then $\Theta_0^D(x_0) \neq A(x_0)$, and \mathcal{R}_{Θ_0} is satisfied finitarily (and the \mathcal{R}_{Θ_1} -strategy has an incorrect guess about the \mathcal{R}_{Θ_0} -strategy).
 - b. The \mathcal{R}_{Θ_0} -strategy goes from 3. to 1. infinitely often. Then $\Delta^D = U$, and \mathcal{R}_{Θ_1} is satisfied finitarily by $\Theta_1^D(x_1) \neq A(x_1)$, using the slow-down feature of the \mathcal{R}_{Θ_0} -strategy in 1.
- B. The \mathcal{R}_{Θ_1} -strategy passes 2. Since $U \upharpoonright (y_1 + 1)$ can only change finitely often, the \mathcal{R}_{Θ_1} -strategy will eventually be in 3a. or in 3b. forever:
 - a. The \mathcal{R}_{Θ_1} -strategy is eventually in 3a. Then, since $U \upharpoonright (y_1 + 1) = U_{s_*} \upharpoonright (y_1 + 1)$, $\Delta^D = U$ is correct, so eventually the \mathcal{R}_{Θ_0} -strategy is not affected by the \mathcal{R}_{Θ_1} -strategy. Furthermore, since $D \upharpoonright (y_1 + 1) = D_{s_*} \upharpoonright (y_1 + 1)$, \mathcal{R}_{Θ_1} is satisfied by $\Theta_1^D(x_1) \downarrow = 0 \neq A(x_1)$.
 - b. Otherwise. Then, since $D \upharpoonright (y_0 + 1) = D_{s_*} \upharpoonright (y_0 + 1)$, \mathcal{R}_{Θ_0} is satisfied finitarily by $\Theta_0^D(x_1) \downarrow = 0 \neq A(x_1)$, a computation that is cleared of $\gamma(w_0)$ by the $U \upharpoonright (y_1 + 1)$ -change. (The \mathcal{R}_{Θ_1} -strategy is “to the left of the true path” since the \mathcal{R}_{Θ_0} -strategy has finite outcome. But, of course, the \mathcal{R}_{Θ_1} -strategy did all the work.)

We have purposely overlooked one minor complication so far.¹ Our assumption at 3b. in the \mathcal{R}_{Θ_1} -strategy was that we would be able to redefine $\Gamma^{D \oplus U}(w_0)$ with $\gamma(w_0) > y_0$ using the $U \upharpoonright (y_1 + 1)$ -change. If that U -change, however, is caused by the extraction of some $y \leq y_1$ from U , then some definition of $\Gamma^{D \oplus U}(w_0)$ may already exist with a small use $\gamma(w_0)$. The problem here is obviously that U had changed “too early” at y since y was enumerated into U before the \mathcal{R}_{Θ_1} -strategy passed 2. We get around this complication by rephrasing 1. of the \mathcal{R}_{Θ_1} -strategy as follows:

- 1'. Wait for $\Theta_1^D(x_1) \downarrow = \Theta_0^D(x_1) \downarrow = 0$, $|B_i| > \vartheta_1(x_1) + 3$, and $U \upharpoonright (\vartheta_1(x_1) + 1) = U_{s_*} \upharpoonright (\vartheta_1(x_1) + 1)$ at some stage s_* (where s_* is the stage at which use block B_{i-1} was defined).

This minor change does not affect the \mathcal{R}_{Θ_1} -strategy since $U \upharpoonright (\vartheta_1(x_1) + 1)$ must settle down if $\Theta_1(x_1)$ is defined. If some $y \in U_{s_*} \upharpoonright (\vartheta_1(x_1) + 1)$ now leaves U after stage s_* then it must have entered U before stage s_* . Since stage s_* , however, D was permanently changed on use block B_{i-1} by the \mathcal{R}_{Θ_0} -strategy, so $D \oplus U$ will, through the extraction of y from U , not return to an initial segment of $D \oplus U$ previously used to define $\Gamma^{D \oplus U}(w_0)$.

Summarizing the situation below one \mathcal{S}_U -strategy (see Diagram 1), we can distinguish *two types* of \mathcal{R}_{Θ} -strategies. First, there are the strategies of type \mathcal{R}_{Θ_0} , which deal with Γ -uses by threatening to destroy $\Gamma^{D \oplus U}$ at their “killing point” w_0 . Below the finite outcomes

¹The authors would like to thank R. Shore for pointing this out.

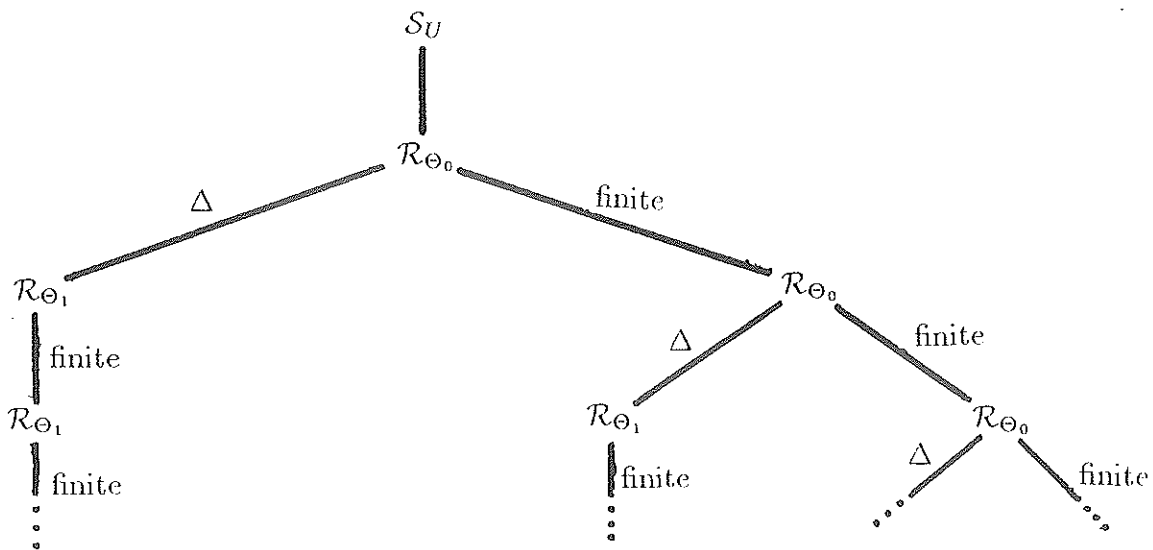


Diagram 1. The \mathcal{R}_Θ -strategies below one S_U -strategy

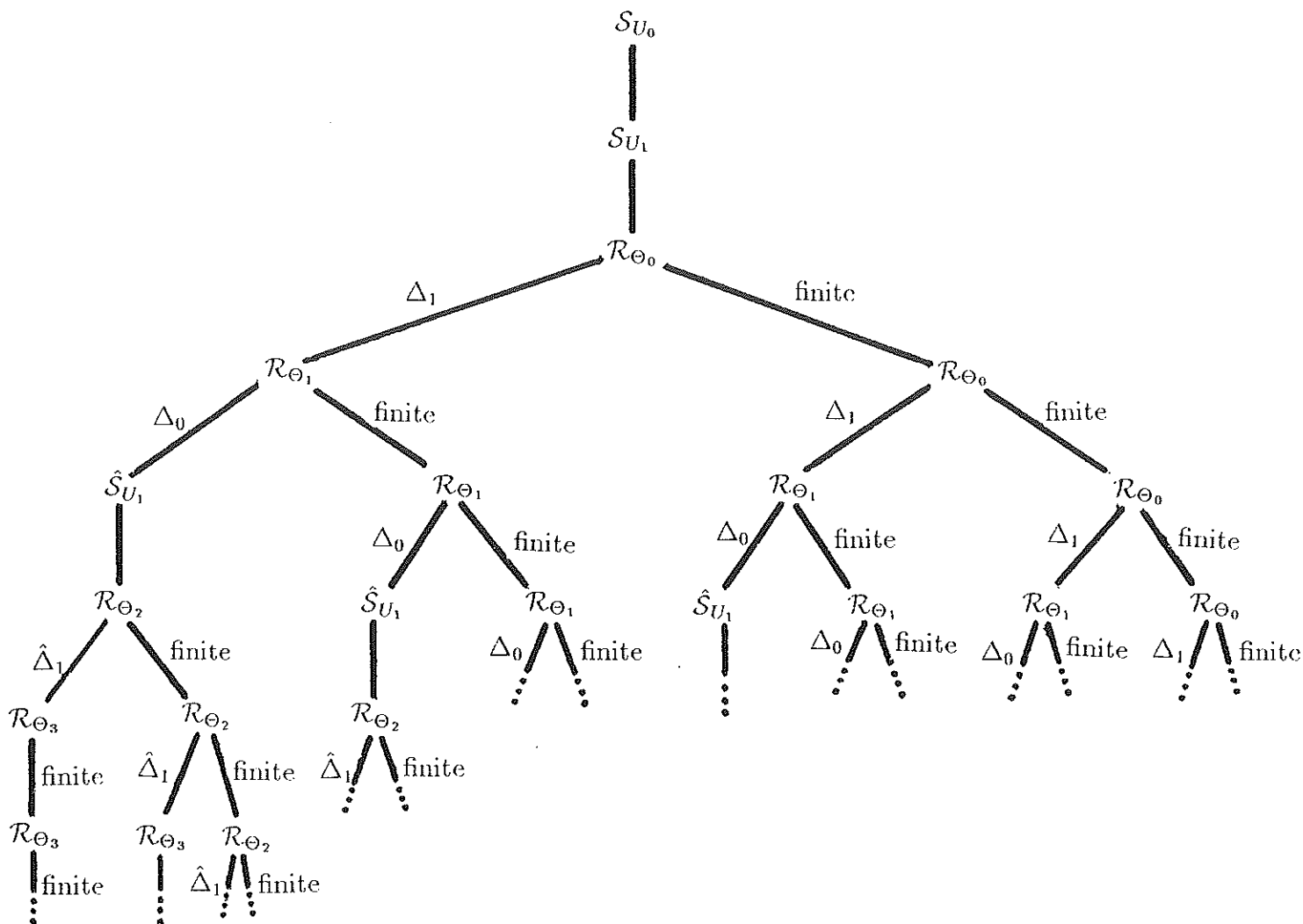


Diagram 2. The \mathcal{R}_Θ -strategies below two S_U -strategies

of these, we are still in the same situation and continue with \mathcal{R}_{Θ_0} -strategies. (“The \mathcal{S}_U -flip from Γ to Δ has not occurred.”) Below the infinite outcome of an \mathcal{R}_{Θ_0} -strategy, we only use strategies of type \mathcal{R}_{Θ_1} , which either satisfy their requirement finitarily without disturbing Δ (“The \mathcal{S}_U -flip from Γ to Δ has occurred.”), or else satisfy \mathcal{R}_{Θ_0} and therefore force themselves to the left of the true path (“The \mathcal{S}_U -flip from Γ to Δ turned out to be fake, and \mathcal{S}_U continues with Γ .”)

5. THE \mathcal{R}_{Θ} -REQUIREMENTS BELOW TWO \mathcal{S}_U -STRATEGIES

The Intuition. It would be accurate to say that this case just involves nesting the above combination of strategies; however, this nesting is very complicated and will therefore be presented in detail. We hope that the case of two higher-priority \mathcal{S}_U -strategies will give the intuition for the general case of arbitrarily many.

Just as we had two types of \mathcal{R}_{Θ} -strategies above (handling Γ - and Δ -uses, respectively), we here have four types of \mathcal{R}_{Θ} -strategies (called $\mathcal{R}_{\Theta_0^-}$, $\mathcal{R}_{\Theta_1^-}$, $\mathcal{R}_{\Theta_2^-}$, and $\mathcal{R}_{\Theta_3^-}$ -strategies, respectively) depending on the Σ_3 -outcomes of the \mathcal{S}_{U_0} - and the \mathcal{S}_{U_1} -strategy (see Diagram 2).

An $\mathcal{R}_{\Theta_0^-}$ -type strategy has only finitary outcomes of \mathcal{R}_{Θ} -strategies above itself. Its job is to destroy $\Gamma_1^{D \oplus U_1}$ unless it happens to find a Θ_0 -computation cleared of *both* Γ_0 - and Γ_1 -uses. Its outcome is either finite, or it destroys $\Gamma_1^{D \oplus U_1}$ and builds $\Delta_1^D = U_1$ (even if Γ_0 is the “culprit”).

An $\mathcal{R}_{\Theta_1^-}$ -type strategy works below the infinite outcome of an $\mathcal{R}_{\Theta_0^-}$ -strategy. It will destroy both $\Gamma_0^{D \oplus U_0}$ and Δ_1^D and build $\Delta_0^D = U_0$ unless it happens to find a Θ_1 -computation cleared of both Γ_0 - and Δ_1 -uses. Its outcome is either finite, or it destroys both $\Gamma_0^{D \oplus U_0}$ and Δ_1^D and builds $\Delta_0^D = U_0$.

Below the infinite outcome of an $\mathcal{R}_{\Theta_1^-}$ -strategy, \mathcal{S}_{U_1} is not satisfied since both $\Gamma_1^{D \oplus U_1}$ and Δ_1^D have been destroyed. So we first have to introduce another version of an \mathcal{S}_{U_1} -strategy, say an $\hat{\mathcal{S}}_{U_1}$ -strategy, building $\hat{\Gamma}_1^{D \oplus U_1}$. Thus an \mathcal{R}_{Θ} -strategy below the infinite outcome of an $\mathcal{R}_{\Theta_1^-}$ -strategy has to deal with Δ_0^D and $\hat{\Gamma}_1^{D \oplus U_1}$.

An $\mathcal{R}_{\Theta_2^-}$ -type strategy works below the infinite outcome of an $\mathcal{R}_{\Theta_1^-}$ -strategy and therefore also below an $\hat{\mathcal{S}}_{U_1}$ -strategy. It will destroy $\hat{\Gamma}_1^{D \oplus U_1}$ and build $\hat{\Delta}_1^D$ unless it happens to find a Θ_2 -computation cleared of both Δ_0 - and $\hat{\Gamma}_1$ -uses.

An $\mathcal{R}_{\Theta_3^-}$ -type strategy works below the infinite outcome of an $\mathcal{R}_{\Theta_2^-}$ -strategy. It has to deal with Δ_0 - and $\hat{\Delta}_1$ -uses in the same way the $\mathcal{R}_{\Theta_1^-}$ -strategy of the previous section dealt with Δ -uses. Namely, roughly speaking, if neither U_0 nor U_1 changes then the Δ_0 - and $\hat{\Delta}_1$ -uses will not cause a problem. If exactly one of U_0 and U_1 changes then this will clear the Θ -computation of the $\mathcal{R}_{\Theta_1^-}$ - or the $\mathcal{R}_{\Theta_2^-}$ -strategy of Γ_0 - or $\hat{\Gamma}_1$ -uses, respectively. And if both U_0 and U_1 change then the $\mathcal{R}_{\Theta_0^-}$ -strategy has a cleared Θ -computation.

We now give a detailed description of the various \mathcal{R}_{Θ} -strategies:

The \mathcal{R}_{Θ_0} -strategy. It works with a fresh witness x_0 for Θ_0 and a “killing point” w_0 for Γ_1 (here i_0 is the number of previous Γ_1 -killings of the \mathcal{R}_{Θ_0} -strategy) and proceeds as follows:

1. Wait for $\Theta_0^D(x_0) \downarrow = 0$ and $\Theta_0^D \uparrow (i_0 + 1) \downarrow = A \uparrow (i_0 + 1)$.
2. If $\min(\gamma_0(w_0)\text{-}, \gamma_1(w_0)\text{-use blocks}) > \vartheta_0(x_0)$ then put x_0 into A , restrain $D \uparrow (\vartheta_0(x_0)+1)$, and stop.
3. Otherwise, define $B_{i_0}^1$ to be the current $\gamma_1(w_0)$ -use block, put $\gamma_1(w_0)$ into D , request that the new $\gamma_1(w_0)$ -use block be very big, define $\Delta_1^D(i_0) = U_1(i_0)$, keep it defined and correct from now on (unless stopped), and go back to 1.

The \mathcal{R}_{Θ_1} -strategy. It is below the infinite outcome of an \mathcal{R}_{Θ_0} -strategy and works with a fresh witness $x_1 > x_0$ for Θ_1 and a “killing point” $w_1 > w_0$ (here i_1 is the number of previous Γ_0 -killings performed by the \mathcal{R}_{Θ_1} -strategy), and at stages at which the \mathcal{R}_{Θ_0} -strategy passes 2., the \mathcal{R}_{Θ_1} -strategy proceeds as follows:

1. Wait for $\Theta_1^D(x_1) \downarrow = 0$, $\Theta_1^D \uparrow (i_1 + 1) = A \uparrow (i_1 + 1)$, $|B_{i_0}^1| > i_1 \cdot \vartheta_1(i_1)$, and $U_j \uparrow (\vartheta_1(i_1) + 1) = U_{j,s^*} \uparrow (\vartheta_1(i_1) + 1)$ for $j = 0, 1$. (Here $B_{i_0}^1$ is the $\gamma_1(w_0)$ -use block just being used by the \mathcal{R}_{Θ_0} -strategy to destroy $\Gamma_1^{D \oplus U_1}(w_0)$, and s^* is the stage at which use block $B_{i_0-1}^1$ was defined. The last three clauses of 1. all slow down the \mathcal{R}_{Θ_1} -strategy.)
2. If $\min(\gamma_0(w_1)\text{-}, \delta_1(w_1)\text{-use blocks}) > \vartheta_1(w_1)$ then put x_1 into A , restrain $D \uparrow (\vartheta_1(x_1)+1)$, and stop.
3. Otherwise, define $B_{i_1}^0$ and $C_{i_1}^1$ to be the current $\gamma_0(w_1)$ - and $\delta_1(w_1)$ -use blocks, respectively, put $\gamma_0(w_1)$ and $\delta_1(w_1)$ into D , request that the new $\gamma_0(w_1)$ - and $\delta_1(w_1)$ -use blocks be very big, define $\Delta_0^D(i_1) = U_0(i_1)$, keep it defined and correct from now on (unless stopped), and go back to 1.

The \mathcal{R}_{Θ_2} -strategy. It is below the infinite outcomes of an \mathcal{R}_{Θ_0} - and an \mathcal{R}_{Θ_1} -strategy and below an \hat{S}_{U_1} -strategy. It works with a fresh witness $x_2 > x_1$ for Θ_2 and a “killing point” $w_2 > w_1$ (here i_2 is the number of previous $\hat{\Gamma}_1$ -killings of the \mathcal{R}_{Θ_2} -strategy), and at stages at which the \mathcal{R}_{Θ_1} -strategy passes 2., the \mathcal{R}_{Θ_2} -strategy proceeds as follows:

1. Wait for $\Theta_2^D(x_2) \downarrow = 0$; $\Theta_2^D \uparrow (i_2 + 1) = A \uparrow (i_2 + 1)$; all of $|B_{i_0}^1|$, $|B_{i_1}^0|$, $|C_{i_1}^1| > i_2 \cdot \vartheta(i_2)$; and $U_j \uparrow (\vartheta_2(i_2) + 1) = U_{j,s^*} \uparrow (\vartheta_2(i_2) + 1)$ for $j = 0, 1$. (Here $B_{i_0}^1$, $B_{i_1}^0$, and $C_{i_1}^1$ are the current $\gamma_1(w_0)$ -, $\gamma_0(w_1)$ -, and $\delta_1(w_1)$ -use blocks, respectively, and s^* is the stage at which $B_{i_1-1}^0$ and $C_{i_1-1}^1$ were defined.)
2. If $\min(\delta_0(w_2)\text{-}, \hat{\gamma}_1(w_2)\text{-use blocks}) > \vartheta_2(x_2)$ then put x_2 into A , restrain $D \uparrow (\vartheta_2(x_2)+1)$, and stop.
3. Otherwise define $\hat{B}_{i_2}^1$ to be the current $\hat{\gamma}_1(w_2)$ -use block, put $\hat{\gamma}_1(w_2)$ into D , request that the new $\hat{\gamma}_1(w_2)$ -use be very big, define $\hat{\Delta}_1^D(i_2) = U_1(i_2)$, keep it defined and correct from now on (unless stopped), and go back to 1.

The \mathcal{R}_{Θ_3} -strategy. It is below the infinite outcome of an \mathcal{R}_{Θ_2} -strategy and works with

a fresh witness x_3 for $\Theta_3, \Theta_2, \Theta_1$, and Θ_0 . At stages at which the \mathcal{R}_{Θ_2} -strategy passes 2., the \mathcal{R}_{Θ_3} -strategy proceeds as follows:

1. Wait for $\Theta_n^D(x_3) \downarrow = 0$ for $n = 0, 1, 2, 3$; $i_0, i_1, i_2 > x_3$; $\min\{|B_{i_0}^1|, |B_{i_1}^0|, |C_{i_1}^1|, |\hat{B}_{i_2}^1|\} > 2 \cdot \vartheta_3(x_3) + 4$; and $U_j \uparrow (\vartheta_3(x_3) + 1) = U_{j, s^*} \uparrow (\vartheta_3(x_3) + 1)$ for $j = 0, 1$ at some stage s_* . (Here $\hat{B}_{i_2}^1$ is the $\hat{\gamma}_1(w_2)$ -use block currently used by the \mathcal{R}_{Θ_2} -strategy to destroy $\hat{\Gamma}_1^{D \oplus U_1}(w_0)$, and s^* is the stage at which $\hat{B}_{i_2-1}^1$ was defined.)
2. Set $y_n = \max_{n \leq m \leq 3} \{\vartheta_m(x_3)\}$ for $n = 0, 1, 2, 3$; set $i_n^* = i_n$ for $n = 0, 1, 2$; put x_3 into A ; and restrain $D \uparrow (y_0 + 1)$ from lower-priority strategies. (We say the \mathcal{R}_{Θ_3} -strategy is *taking charge* of the $\mathcal{R}_{\Theta_0^-}$, $\mathcal{R}_{\Theta_1^-}$, and $\mathcal{R}_{\Theta_2^-}$ -strategies.)
3. From now on ensure (using $|B| > 2y_3 + 4$ in a., $|B| > 2y_2 + 4$ in b., and $|B_{i_0}^1| > 2y_1 + 4$ in c.):
 - a. Whenever $U_0 \uparrow (y_3 + 1) = U_{0, s_*} \uparrow (y_3 + 1)$ and $U_1 \uparrow (y_3 + 1) = U_{1, s_*} \uparrow (y_3 + 1)$ then let the \mathcal{R}_{Θ_n} -strategies act for $n = 0, 1, 2$; and have $D \cap B \neq D_{s_*} \cap B$ for $B = B_{i_0}^1, B_{i_1}^0, C_{i_1}^1, \hat{B}_{i_2}^1$. (\mathcal{R}_{Θ_3} is satisfied, and $\hat{\Delta}_1^D$ and Δ_0^D are correct.)
 - b. Otherwise, whenever $U_0 \uparrow (y_2 + 1) = U_{0, s_*} \uparrow (y_2 + 1)$ and $U_1 \uparrow (y_3 + 1) \neq U_{1, s_*} \uparrow (y_3 + 1)$ then let the \mathcal{R}_{Θ_n} -strategies act for $n = 0, 1$; prevent the \mathcal{R}_{Θ_2} -strategy from acting; restore $D \uparrow (y_2 + 1) = D_{s_*} \uparrow (y_2 + 1)$; and have $D \cap B \neq D_{s_*} \cap B$ for $B = B_{i_0}^1, B_{i_1}^0, C_{i_1}^1$. (\mathcal{R}_{Θ_2} is satisfied and cleared of Γ_1, Γ_0 , and $\hat{\Gamma}_1$; and Δ_0^D is correct.)
 - c. Otherwise, whenever $U_0 \uparrow (y_2 + 1) \neq U_{0, s_*} \uparrow (y_2 + 1)$ and $U_1 \uparrow (y_1 + 1) = U_{1, s_*} \uparrow (y_1 + 1)$ then let the \mathcal{R}_{Θ_0} -strategy act; prevent the \mathcal{R}_{Θ_n} -strategies from acting for $n = 1, 2$; restore $D \uparrow (y_1 + 1) = D_{s_*} \uparrow (y_1 + 1)$; and have $D \cap B_{i_0}^1 \neq D_{s_*} \cap B_{i_0}^1$. (\mathcal{R}_{Θ_1} is satisfied and cleared of Γ_1 and Γ_0 while $\hat{\Gamma}_1$ is not active; and Δ_1^D is correct.)
 - d. Otherwise (i.e. whenever $U_0 \uparrow (y_2 + 1) \neq U_{0, s_*} \uparrow (y_2 + 1)$ and $U_1 \uparrow (y_3 + 1) \neq U_{1, s_*} \uparrow (y_3 + 1)$; or $U_0 \uparrow (y_3 + 1) \neq U_{0, s_*} \uparrow (y_3 + 1)$ and $U_1 \uparrow (y_1 + 1) \neq U_{1, s_*} \uparrow (y_1 + 1)$), prevent the \mathcal{R}_{Θ_n} -strategies from acting for $n = 0, 1, 2$; and restore $D \uparrow (y_0 + 1) = D_{s_*} \uparrow (y_0 + 1)$. (\mathcal{R}_{Θ_0} is satisfied and cleared of Γ_0 and Γ_1 while $\hat{\Gamma}_1$ is not active.)

The possible combined outcomes. The possible outcomes of the \mathcal{R}_{Θ_n} -strategies for $n = 0, 1, 2, 3$ are as follow:

- A. The \mathcal{R}_{Θ_3} -strategy never reaches 2.
 - a. For some $n \leq 2$, the \mathcal{R}_{Θ_n} -strategy eventually stops at 2. or waits at 1. forever. For the least such n , $\Theta_n^D \neq A$ (notwithstanding the slow-down features, as in the previous section); also the Δ built by the $\mathcal{R}_{\Theta_{n-1}}$ -strategy is correct if $n > 0$.
 - b. Otherwise the \mathcal{R}_{Θ_3} -strategy cannot proceed to 2. because of 1., and \mathcal{R}_{Θ_3} is satisfied while Δ_0^D and $\hat{\Delta}_1^D$ are correct.
- B. The \mathcal{R}_{Θ_3} -strategy passes 2. Since $U_0 \uparrow (y_1 + 1)$ and $U_1 \uparrow (y_1 + 1)$ can only change finitely often, the \mathcal{R}_{Θ_3} -strategy will eventually be in one of 3a. through 3d. forever:
 - a. Since $D \uparrow (y_3 + 1) = D_{s_*} \uparrow (y_3 + 1)$, \mathcal{R}_{Θ_3} is satisfied by $\Theta_3^D(x_3) \downarrow = 0 \neq A(x_3)$; since $U_0 \uparrow (y_3 + 1) = U_{0, s_*} \uparrow (y_3 + 1)$ and $U_1 \uparrow (y_3 + 1) = U_{1, s_*} \uparrow (y_3 + 1)$, both $\Delta_0^D = U_0$

and $\hat{\Delta}_1^D = U_1$ are correct.

- b. Since $D \uparrow (y_2 + 1) = D_{s_*} \uparrow (y_2 + 1)$, \mathcal{R}_{Θ_2} is satisfied by $\Theta_2^D(x_3) \downarrow = 0 \neq A(x_3)$, a computation that is cleared of $\hat{\gamma}_1(w_2)$ by the $U_1 \uparrow (y_3 + 1)$ -change and of $\gamma_0(w_1)$, $\delta_1(w_1)$, and $\gamma_1(w_0)$ by the D -change; since $U_0 \uparrow (y_2 + 1) = U_{0,s_*} \uparrow (y_2 + 1)$, $\Delta_0^D = U_0$ is correct.
- c. Since $D \uparrow (y_1 + 1) = D_{s_*} \uparrow (y_1 + 1)$, \mathcal{R}_{Θ_1} is satisfied by $\Theta_1^D(x_3) \downarrow = 0 \neq A(x_3)$, a computation that is cleared of $\gamma_0(w_1)$ by the $U_0 \uparrow (y_2 + 1)$ -change and of $\gamma_1(w_0)$ by the D -change; since $U_1 \uparrow (y_1 + 1) = U_{1,s_*} \uparrow (y_1 + 1)$, $\Delta_1^D = U_1$ is correct.
- d. Since $D \uparrow (y_0 + 1) = D_{s_*} \uparrow (y_0 + 1)$, \mathcal{R}_{Θ_0} is satisfied by $\Theta_0^D(x_3) \downarrow = 0 \neq A(x_3)$, a computation that is cleared of $\gamma_1(w_0)$ and $\gamma_0(w_1)$ by the $U_1 \uparrow (y_1 + 1)$ - and $U_0 \uparrow (y_2 + 1)$ -changes, respectively.

There is an easier variant of \mathcal{R}_{Θ} -strategies working below two \mathcal{S}_U -strategies, namely when Γ_0 is destroyed and Δ_0 is built by an \mathcal{R}_{Θ} -strategy before the \mathcal{S}_{U_1} -requirement is introduced. In this case, the \mathcal{R}_{Θ_0} - and \mathcal{R}_{Θ_1} -strategies above collapse into one strategy, namely the \mathcal{R}_{Θ_0} -strategy from the previous section, and the \mathcal{R}_{Θ_2} -strategy works below the infinite outcome of it and the \mathcal{S}_{U_1} -strategy as described above, using Γ_1 in place of $\hat{\Gamma}_1$. For the \mathcal{R}_{Θ_3} -strategy, cases 3c. and 3d. collapse into one, namely “whenever $U_0 \uparrow (y_3 + 1) \neq U_{0,s_*} \uparrow (y_3 + 1)$, or $U_0 \uparrow (y_2 + 1) \neq U_{0,s_*} \uparrow (y_2 + 1)$ and $U_1 \uparrow (y_3 + 1) \neq U_{1,s_*} \uparrow (y_3 + 1)$ ”, in which case \mathcal{R}_{Θ_0} is satisfied by $\Theta_0^D(x_3) \downarrow = 0 \neq A(x_3)$, a computation cleared of $\gamma_0(w_0)$ by the $U_0 \uparrow (y_2 + 1)$ -change while Γ_1 is not active since the \mathcal{R}_{Θ_0} -strategy has finite outcome.

6. THE GENERAL \mathcal{R}_{Θ} -STRATEGY

In general, an \mathcal{R}_{Θ} -strategy will have to deal with a finite number of Γ 's built by higher-priority \mathcal{S}_U -strategies (and not destroyed yet), and a finite number of Δ 's built by higher-priority \mathcal{R}_{Θ} -strategies (and not destroyed yet). There are two cases:

Case 1. There is such a Γ . Then the \mathcal{R}_{Θ} -strategy will destroy the lowest-priority one of them, say Γ_* , and also the Δ 's of lower priority than Γ_* , and it will build $\Delta_*^D = U_*$, unless it happens to find a Θ -computation cleared of all γ - and δ -uses. This \mathcal{R}_{Θ} -strategy will thus work like the \mathcal{R}_{Θ_0} -strategy, or the \mathcal{R}_{Θ_0} -, \mathcal{R}_{Θ_1} -, and \mathcal{R}_{Θ_2} -strategies, from the previous two sections, respectively. We call this type of an \mathcal{R}_{Θ} -strategy an *\mathcal{R}_{Θ} -destroyer strategy*.

An \mathcal{R}_{Θ} -destroyer strategy has a fixed “killing number” w which is only increased whenever it appears to be to the right of the true path. Whenever $K \uparrow w$, or $U \uparrow w$ for some higher-priority \mathcal{S}_U -strategy, changes, the \mathcal{R}_{Θ} -destroyer strategy is initialized while keeping w fixed. (The strategy thus assumes that $K \uparrow w$ and the $U \uparrow w$ have settled down, which is true after finite injury.) The strategy will make $\Gamma_*^{D \oplus U}$ and those Δ^D 's of lower priority than Γ_* undefined at w unless \mathcal{R}_{Θ} can be satisfied finitarily.

Case 2. The \mathcal{R}_{Θ} -strategy has to deal only with higher-priority Δ 's. Then the \mathcal{R}_{Θ} -strategy will wait for computations at some x for its own Θ and all the $\bar{\Theta}$ of the higher-priority $\mathcal{R}_{\bar{\Theta}}$ -

strategies with infinite outcome (*including* the ones whose $\bar{\Delta}$ has been destroyed already), in such a way that $\bar{\vartheta}(x)$ (for each such $\bar{\vartheta}$) and $\vartheta(x)$ is less than the length of the use blocks of $\bar{\Gamma}$'s and $\bar{\Delta}$'s destroyed by $\mathcal{R}_{\bar{\Theta}}$ -strategies of higher priority than the \mathcal{R}_{Θ} -strategy. If our \mathcal{R}_{Θ} -strategy fails to find all these computations as specified, then \mathcal{R}_{Θ} is satisfied without taking any action, else our \mathcal{R}_{Θ} -strategy will attack using its witness x (“take charge of all higher-priority infinitary $\mathcal{R}_{\bar{\Theta}}$ -destroyer strategies”) and ensure that it itself or one of these higher-priority $\mathcal{R}_{\bar{\Theta}}$ -destroyer strategies satisfies its requirement finitarily. This \mathcal{R}_{Θ} -strategy will thus work like the \mathcal{R}_{Θ_1} -strategy, or the \mathcal{R}_{Θ_s} -strategy, from the previous two sections, respectively. We call this type of \mathcal{R}_{Θ} -strategy an \mathcal{R}_{Θ} -*controller strategy*.

There is one minor difference between an \mathcal{R}_{Θ} -destroyer strategy satisfying its requirement and a lower-priority $\mathcal{R}_{\bar{\Theta}}$ -controller strategy satisfying \mathcal{R}_{Θ} : The latter may satisfy \mathcal{R}_{Θ} using the “killing number” \bar{w} of an $\mathcal{R}_{\bar{\Theta}}$ -destroyer strategy of lower priority than the \mathcal{R}_{Θ} -strategy. Thus $\bar{w} > w$, and \mathcal{R}_{Θ} may turn out to be not satisfied because K or some U changes on $[w, \bar{w})$. In that case, the \mathcal{R}_{Θ} -destroyer strategy should really have continued acting instead of relying on the $\mathcal{R}_{\bar{\Theta}}$ -controller strategy. But nothing is lost if the \mathcal{R}_{Θ} -destroyer strategy now just “catches up” on all the actions it failed to perform since the $\mathcal{R}_{\bar{\Theta}}$ -controller strategy stopped it.

7. DEFINING THE Γ 'S AND Δ 'S

We have thus far been very vague about how the Γ 's and Δ 's are defined apart from saying that their uses, and the lengths of their use blocks, should be increased to clear Θ -computations, and to allow to destroy and restore computations a greater and greater number of times, respectively. The problem is to balance these demands against the need to make the Γ 's and Δ 's total (unless deliberately destroyed by one fixed lower-priority \mathcal{R}_{Θ} -destroyer strategy), and therefore to let their uses come to finite limits. Surprisingly, the technique is the same for the construction of a Γ by an \mathcal{S}_U -strategy and of a Δ by an \mathcal{R}_{Θ} -destroyer strategy. The only difference is that an \mathcal{S}_U -strategy will act whenever it appears to be on the true path while an \mathcal{R}_{Θ} -destroyer strategy may, even when apparently on the true path, be prevented from acting by an $\mathcal{R}_{\bar{\Theta}}$ -controller strategy. In the following, we thus restrict ourselves to explaining the definition of Γ .

When an \mathcal{S}_U -strategy defines Γ on some argument w , it reserves a *use block* B for its use, i.e. an interval of ω of length less than its least element, reserved solely for correcting $\Gamma^{D \oplus U}(w)$ and for destroying and restoring $\Gamma^{D \oplus U}(w)$. We call a $\Theta^D(x)$ -computation *cleared of $\gamma(w)$* when $\vartheta(x) < \text{least element of the use block for } \Gamma^{D \oplus U}(w)$. (Consistently with standard notation, however, we will denote by $\gamma(w)$ itself the greatest element of this use block.) We agree that whenever some element needs to enter this use block, it will be the leftmost unused element of it.

For the initial definition of $\Gamma^{D \oplus U}(w)$, the \mathcal{S}_U -strategy will pick a use block beyond any number, and longer than any use block, mentioned thus far in the construction.

Occasionally, the oracle $D \oplus U$ may change below a use $\gamma(w)$, and no previous definition of $\Gamma^{D \oplus U}(w)$ may apply. The \mathcal{S}_U -strategy now has to decide whether or not to increase the use and proceed to a new use block.

If D has not changed on any $\gamma(w')$ -use block for any $w' \leq w$ then we will keep the rightmost use block previously used for $\Gamma^{D \oplus U}(w)$ and define $\Gamma^{D \oplus U}(w) = K(w)$ with use $\gamma(w)$ equal to the rightmost element of that use block.

Otherwise, we will pick a new use block with leftmost element, and of length, greater than any number mentioned thus far in the construction, and we will define $\Gamma^{D \oplus U}(w) = K(w)$ with use equal to the rightmost element of that new use block.

Now if infinitely often some use block of $\Gamma^{D \oplus U}(w)$ changes for fixed w and so $\lim_s \gamma_s(w) = \infty$, then we have to ensure that one fixed \mathcal{R}_Θ -destroyer strategy wants to destroy $\Gamma^{D \oplus U}$. Otherwise, the use $\gamma(w)$ will come to a finite limit, and $\Gamma^{D \oplus U}(w)$ will eventually be defined permanently. Furthermore, the \mathcal{S}_U -strategy can correct $\Gamma^{D \oplus U}(w)$ by enumerating an element into the use block of $\Gamma^{D \oplus U}(w)$ when $K(w)$ changes. Thus at any stage at which the \mathcal{S}_U -strategy can act, its $\Gamma^{D \oplus U}$ will correctly compute K on its domain.

We are now ready for the formal description of the construction.

8. THE TREE OF STRATEGIES

As usual in \mathbf{O}''' -priority arguments, the construction uses a tree of strategies. For the sake of simplicity, we use a binary tree $T \subseteq 2^{<\omega}$ where we interpret 0 as the infinite and 1 as the finite outcome of a strategy $\xi \in T$. Of course, \mathcal{S}_U -strategies only have infinite, and \mathcal{R}_Θ -controller strategies only finite outcome, while \mathcal{R}_Θ -destroyer strategies may have either outcome.

We start with a definition about satisfaction of requirements on the tree:

DEFINITION 1: (i) The *priority ranking of the requirements* will be $\mathcal{S}_{U_0}, \mathcal{R}_{\Theta_0}, \mathcal{S}_{U_1}, \mathcal{R}_{\Theta_1}$, etc. where $\{U_e\}_{e \in \omega}$ and $\{\Theta_e\}_{e \in \omega}$ are effective enumerations of all d.r.e. sets and all partial recursive functionals, respectively.

Let $\xi \in T$.

(ii) A requirement \mathcal{S}_U is *active at* ξ iff there is an \mathcal{S}_U -strategy $\alpha \subset \xi$ and there is no \mathcal{R}_Θ -destroyer strategy β with $\alpha \subset \beta \hat{\ } 0 \subseteq \xi$ that is targeted to destroy α 's Γ (as defined below).

(iii) A requirement \mathcal{S}_U is *satisfied at* ξ iff there is an \mathcal{R}_Θ -destroyer strategy β such that $\beta \hat{\ } 0 \subseteq \xi$, β is targeted to destroy the Γ of an \mathcal{S}_U -strategy $\alpha \subset \beta$, and there is no \mathcal{R}_Θ -destroyer strategy $\tilde{\beta}$ such that $\beta \subset \tilde{\beta} \hat{\ } 0 \subseteq \xi$ and $\tilde{\beta}$ is targeted to destroy a Γ of an $\mathcal{S}_{\tilde{U}}$ -strategy $\tilde{\alpha} \subset \alpha$. (Note that here $\mathcal{S}_{\tilde{U}}$ would be a higher-priority requirement than \mathcal{S}_U by Definition 2.)

(iv) A requirement \mathcal{R}_Θ is *satisfied at* ξ iff there is an \mathcal{R}_Θ -(destroyer or controller) strategy η with $\eta \hat{\ } 1 \subseteq \xi$.

(The intuition is that in cases (iii) and (iv), \mathcal{S}_U and \mathcal{R}_Θ are satisfied by some Δ or finitarily, respectively, whereas in case (ii) the Γ for \mathcal{S}_U could still be destroyed at or below ξ .)

We can now define the tree T and the assignment of requirements to nodes of T :

DEFINITION 2: We proceed by induction on $|\xi|$ (starting with $\emptyset \in T$). Let $\xi \in T$.

(i) The node ξ *works on* (or, *is assigned to*) the highest-priority requirement that is neither active nor satisfied at ξ .

(ii) If ξ is an \mathcal{R}_Θ -strategy (i.e. works on \mathcal{R}_Θ) then ξ is an \mathcal{R}_Θ -*controller strategy* iff no requirement \mathcal{S}_U is active at ξ ; otherwise ξ is an \mathcal{R}_Θ -*destroyer strategy* targeted to destroy the Γ of the longest $\alpha \subset \xi$ such that α 's \mathcal{S}_U -requirement is active at ξ .

(iii) The *immediate successors of ξ on T* are $\xi \hat{\ } 0$ if ξ is an \mathcal{S}_U -strategy, $\xi \hat{\ } 1$ if ξ is an \mathcal{R}_Θ -controller strategy, and both $\xi \hat{\ } 0$ and $\xi \hat{\ } 1$ if ξ is an \mathcal{R}_Θ -destroyer strategy.

We are now in a position to prove a lemma about the formal structure of T :

LEMMA 1 (Finite Injury and Satisfaction Along Any Path Lemma). *Let p be a path through T and \mathcal{R} a requirement. Then \mathcal{R} is assigned to only finitely many nodes $\xi \subset p$. If ξ_0 is the longest such, then either \mathcal{R} is satisfied at $p \upharpoonright n$ via ξ_0 for all $n > |\xi_0|$, or \mathcal{R} is active at $p \upharpoonright n$ via ξ_0 for all $n > |\xi_0|$.*

PROOF: Let \mathcal{R} be the least requirement for which the lemma fails. Fix the least n_0 such that all higher-priority requirements are either satisfied at $p \upharpoonright n$ for all $n \geq n_0$ or active at $p \upharpoonright n$ for all $n \geq n_0$.

Thus some strategy $\xi \subseteq p \upharpoonright n_0$ must work on our fixed requirement. Pick ξ maximal such. If it is an \mathcal{R}_Θ -requirement then, by maximality of ξ and our assumption on n_0 , $\xi \hat{\ } 1 \subseteq p$, and then \mathcal{R}_Θ is satisfied at $p \upharpoonright n$ via $\xi_0 = \xi$ for all $n > |\xi_0|$. If it is an \mathcal{S}_U -requirement then either it is active at $p \upharpoonright n$ via $\xi_0 = \xi$ for all $n > |\xi_0|$; or there is a minimal $n_1 > |\xi|$ such that it is satisfied at $p \upharpoonright n_1$ via some \mathcal{R}_Θ -strategy η . By maximality of ξ and minimality of n_0 , necessarily $n_1 > n_0$. Thus by our assumption on n_0 , no higher-priority \mathcal{S}_U -requirement can become satisfied at $p \upharpoonright n$ for $n > n_1$, and so \mathcal{S}_U is satisfied at $p \upharpoonright n$ via $\xi_0 = \eta$ for all $n > |\xi_0|$. \square

For the remainder of this section, we fix an arbitrary $\mathcal{R}_{\bar{\Theta}}$ -controller strategy γ and develop some notation for the \mathcal{S}_U -strategies $\alpha \subset \gamma$ and the \mathcal{R}_Θ -destroyer strategies β with $\beta \hat{\ } 0 \subseteq \gamma$. This notation is relative to our fixed γ and will be useful in the description of the construction and the verification.

Denote by $\mathcal{S}_{U_0}, \mathcal{S}_{U_1}, \dots, \mathcal{S}_{U_{j_0}}$ the \mathcal{S}_U -requirements of higher priority than $\mathcal{R}_{\bar{\Theta}}$ (in decreasing order of priority). Denote by $\alpha_0 \subset \alpha_1 \subset \dots \subset \alpha_{k_0}$ the \mathcal{S}_U -strategies $\alpha \subset \gamma$. Denote by $\beta_0 \subset \beta_1 \subset \dots \subset \beta_{l_0-1}$ the \mathcal{R}_Θ -destroyer strategies β with $\beta \hat{\ } 0 \subseteq \gamma$, and set $\beta_{l_0} = \gamma$. For each β_l ($0 \leq l \leq l_0$), define a guess $\sigma_l \in 2^{\leq j_0+1}$ of β_l on the Σ_3 -outcomes of

higher-priority \mathcal{S}_U -strategies as follows (for $0 \leq j \leq j_0$):

$$\sigma_l(j) = \begin{cases} 0 & \text{if } \mathcal{S}_{U_j} \text{ is active at } \beta_l, \\ 1 & \text{if } \mathcal{S}_{U_j} \text{ is satisfied at } \beta_l, \\ \uparrow & \text{otherwise.} \end{cases}$$

(The intuition is that, in the first case, β_l assumes $\Gamma_{\alpha_k}^{D \oplus U_j} = K$ (where \mathcal{S}_{U_j} is active at β_l via α_k); in the second case, β_l assumes $\Delta_{\beta_{l'}}^D = U_j$ (where \mathcal{S}_{U_j} is satisfied at β_l via $\beta_{l'}$ for some $l' < l$); and in the third case, \mathcal{S}_{U_j} has lower priority than the requirement of β_l .)

We next define parameters that will help γ decide whether or not to stop a certain β_l . For $l \leq l_0$, $j < |\sigma_l|$, we set

$$L(l, j) = \begin{cases} l' + 1 & \text{if } \sigma_l(j) = 0, \text{ where } l' \geq l \text{ is minimal such that} \\ & \beta_{l'} \text{ is targeted to destroy the } \Gamma \text{ of some } \mathcal{S}_{U_j}\text{-strategy,} \\ l & \text{if } \sigma_l(j) = 1. \end{cases}$$

We now prove a combinatorial lemma, which will show that the $\mathcal{R}_{\bar{\Theta}}$ -controller strategy γ can act as prescribed in the next section:

LEMMA 2 (\mathcal{R}_{Θ} -Controller Strategy Decision Lemma). (i) If $l < l' \leq l_0$ then $|\sigma_l| \leq |\sigma_{l'}$ and $\sigma_l <_L \sigma_{l'}$.

(ii) The set $\{\sigma_l \mid l \leq l_0\}$ forms a maximal antichain in $2^{\leq j_0+1}$.

(iii) For all $l \leq l_0$, all $j \leq j_0$, if $\sigma_l(j)$ is defined then so is $L(l, j)$.

(iv) If $j \leq j_0$ and $l < \tilde{l} \leq l_0$, and both $L(l, j)$ and $L(\tilde{l}, j)$ are defined then $L(l, j) \leq L(\tilde{l}, j)$.

(v) Let U_0, U_1, \dots, U_{j_0} be the d.r.e. sets from above. For all stages $s_* < s$ and all numbers $y_0 \geq y_1 \geq \dots \geq y_{l_0}$, there is $l \leq l_0$ such that for all $j < |\sigma_l|$

$$(1a) \quad \sigma_l(j) = 0 \rightarrow U_{j,s} \upharpoonright (y_{L(l,j)} + 1) \neq U_{j,s_*} \upharpoonright (y_{L(l,j)} + 1),$$

and

$$(1b) \quad \sigma_l(j) = 1 \rightarrow U_{j,s} \upharpoonright (y_{L(l,j)} + 1) = U_{j,s_*} \upharpoonright (y_{L(l,j)} + 1).$$

(Intuitively speaking, an \mathcal{R}_{Θ} -controller β_{l_0} will think that it is β_l that has finite outcome, so it will let $\beta_{l'}$ act iff $l' < l$.)

PROOF: (i) Obvious by the way requirements are assigned to nodes of T .

(ii) By (i), the set is an antichain.

It is easy to verify that $\sigma_0 \subseteq 0^{j_0+1}$ and $\sigma_{l_0} = 1^{j_0+1}$. It therefore remains to verify that there is no τ with $\sigma_l <_L \tau <_L \sigma_{l+1}$ for some $l < l_0$. Since β_l is an \mathcal{R}_{Θ} -destroyer strategy, some \mathcal{S}_U is active at β_l , and so $\sigma_l = \sigma \hat{0} \hat{1}^m$ (for some σ and some $m \geq 0$). If we had applied the definition of σ_l to $\beta_l \hat{0}$ instead of β_l , we would have obtained $\sigma \hat{1}$.

Now between β_l and β_{l+1} , there are no \mathcal{R}_Θ -destroyer strategies β with $\beta \hat{0} \subseteq \beta_{l+1}$. Thus $\sigma_{l+1} = \sigma \hat{1} \hat{0}^{m'}$ (for some $m' \geq m$). Now observe that there is no τ with $\sigma \hat{0} \hat{1}^m <_L \tau <_L \sigma \hat{1} \hat{0}^{m'}$.

(iii) It suffices to show that if $\sigma_l(j) = 0$ then l' in the definition of $L(l, j)$ exists. But \mathcal{S}_{U_j} is active at β_l (since $\sigma_l(j) = 0$) and is satisfied at γ , so there must be some $\beta_{l'}$ with $\beta_l \subseteq \beta_{l'} \subset \beta_{l'} \hat{0} \subseteq \gamma$ that is targeted to kill the Γ of some \mathcal{S}_{U_j} -strategy, so $L(l, j) = l' + 1$.

(iv) The claim is clear by the definition of $L(-, j)$ if $\sigma_l(j) = 1$ or $\sigma_{\tilde{l}}(j) = 0$. So assume $\sigma_l(j) = 0$ and $\sigma_{\tilde{l}}(j) = 1$. Then \mathcal{S}_{U_j} is active at β_l and satisfied at $\beta_{\tilde{l}}$, so there must be some β with $\beta_l \subseteq \beta \subset \beta \hat{0} \subseteq \beta_{\tilde{l}}$ that is targeted to kill the Γ of some \mathcal{S}_{U_j} -strategy. Since $\beta = \beta_{l'}$ for some $l \leq l' < \tilde{l}$ we have $L(l, j) \leq l' + 1 \leq \tilde{l} = L(\tilde{l}, j)$.

(v) We construct a string $\sigma \in 2^{\leq j_0+1}$ by induction, satisfying for all $m \leq |\sigma|$, all $l \leq l_0$, and all $j \leq m$,

$$\sigma \upharpoonright m \subseteq \sigma_l \ \& \ \sigma(j) = 0 \rightarrow U_{j,s} \upharpoonright (y_{L(l,j)} + 1) \neq U_{j,s_*} \upharpoonright (y_{L(l,j)} + 1)$$

and

$$\sigma \upharpoonright m \subseteq \sigma_l \ \& \ \sigma(j) = 1 \rightarrow U_{j,s} \upharpoonright (y_{L(l,j)} + 1) = U_{j,s_*} \upharpoonright (y_{L(l,j)} + 1).$$

We will continue the definition of σ until $\sigma = \sigma_l$ for some $l \leq l_0$. This process must terminate by (ii).

Suppose $\sigma \upharpoonright m$ has been defined and $\sigma \upharpoonright m \neq \sigma_l$ for all $l \leq l_0$. Then, by (ii), there must be σ_l extending $(\sigma \upharpoonright m) \hat{0}$ and $\sigma_{l'}$ extending $(\sigma \upharpoonright m) \hat{1}$. Let \tilde{l} be maximal such that $(\sigma \upharpoonright m) \hat{0} \subseteq \sigma_{\tilde{l}}$. Now, by (iv), $L(\tilde{l}, m) \leq L(\tilde{l} + 1, m)$. Thus $y_{L(\tilde{l}, m)} \geq y_{L(\tilde{l}+1, m)}$; and so $U_{m,s} \upharpoonright (y_{L(\tilde{l}, m)} + 1) \neq U_{m,s_*} \upharpoonright (y_{L(\tilde{l}, m)} + 1)$ or $U_{m,s} \upharpoonright (y_{L(\tilde{l}+1, m)} + 1) = U_{m,s_*} \upharpoonright (y_{L(\tilde{l}+1, m)} + 1)$ holds true.

Set $\sigma(m) = 0$ in the first case, and $\sigma(m) = 1$ otherwise. Then $\sigma \upharpoonright (m + 1)$ satisfies the inductive condition. \square

9. THE CONSTRUCTION

We use the tree of strategies defined above to describe the construction.

We construct one d.r.e. set D and one r.e. set A . Each \mathcal{S}_U -strategy $\alpha \in T$ builds its own p.r. functional Γ_α , and each \mathcal{R}_Θ -destroyer strategy $\beta \in T$ builds its own p.r. functional Δ_β . (By abuse of notation, we may leave off the indices α and β .) Whenever a strategy ξ is *initialized*, all its parameters become undefined, its functional (if any) becomes totally undefined, and ξ no longer takes charge of any other strategy. The same holds when a strategy is *reset* except that then the killing number w of an \mathcal{R}_Θ -destroyer strategy does not become undefined. (Intuitively, we will initialize a strategy that appears to be to the right of the true path, and we will reset a strategy if there is a K - or U -change below its killing point.)

At stage 0, all strategies are initialized, and D and A are set to \emptyset .

At a stage $s + 1$, we first reset every strategy $\xi \in T$ for which there is an \mathcal{R}_Θ -destroyer strategy $\beta \leq \xi$ such that $K_{s+1} \upharpoonright w_\beta \neq K_s \upharpoonright w_\beta$, or such that $U_{\alpha, s+1} \upharpoonright w_\beta \neq U_{\alpha, s} \upharpoonright w_\beta$ for some \mathcal{S}_U -strategy $\alpha \subset \beta$.

Next, we let certain strategies on T be *eligible to act* as follows: First we let \emptyset be eligible to act. Given ξ that is eligible to act, we allow an immediate successor of ξ (determined by ξ 's action as defined below) to be eligible to act next. We proceed to stage $s + 2$ and initialize all strategies $\eta > \xi$ (if $|\xi| = s$), or initialize all strategies $\eta > \xi'$ (if the stage is ended by some strategy ξ').

For the remainder of this section, we describe the action of an individual strategy ξ . This description, of course, splits into three cases, depending on what type of strategy ξ is.

Case 1. ξ is an \mathcal{S}_U -strategy. For each $w \leq s$ (in increasing order), proceed according to the subcase that applies:

Case 1a. $\Gamma^{D \oplus U}(w) \downarrow = K(w)$. Do nothing.

Case 1b. $\Gamma^{D \oplus U}(w) \downarrow \neq K(w)$. Put the least unused element of the current $\gamma(w)$ -use block into D , and continue as in Case 1d.

Case 1c. $\Gamma^{D \oplus U}(w) \uparrow$, $\Gamma^{D \oplus U}(w)[s'] \downarrow$ for $s' \leq s$ maximal, and D has not changed on any of the $\gamma(w')$ -use blocks between stage s' and now (for any $w' \leq w$). Redefine $\Gamma^{D \oplus U}(w) = K(w)$ with the largest $\gamma(w)$ -use block defined so far as the new $\gamma(w)$ -use block. (This need not be the $\gamma(w)$ -use block from stage s' .)

Case 1d. Otherwise. (Re)define $\Gamma^{D \oplus U}(w) = K(w)$ with a use block of length (and thus with least element) greater than any number mentioned thus far in the construction.

In any case, $\xi \hat{\ } 0$ is eligible to act next.

Case 2. ξ is an \mathcal{R}_Θ -destroyer strategy targeted to destroy the Γ_α of some \mathcal{S}_U -strategy $\alpha \subset \xi$.

Let $i = i_\xi$. First, check if ξ 's killing point $w = w_\xi$ or ξ 's witness $x = x_\xi$ is undefined. If so then redefine it/them to a number greater than any number mentioned thus far in the construction, and let $i = 0$.

Next, check if ξ has stopped by itself (as defined below) since it was last initialized or reset. If so, end ξ 's action at this stage by letting $\xi \hat{\ } 1$ be eligible next.

Next, check if one or more $\mathcal{R}_{\Theta'}$ -controller strategies $\gamma \supset \xi$ are taking charge of ξ (as defined below). If so, then let these γ 's act now in decreasing order of priority (with respect to the ordering \leq on T) according to Case 3b below. If one of these γ 's stops ξ then end ξ 's action at this stage by letting $\xi \hat{\ } 1$ be eligible next. (Of course, if γ ends the stage then neither ξ nor $\xi \hat{\ } 1$ will act at this stage.)

Finally, let \mathcal{C} be the set of all \mathcal{R}_Θ -destroyer strategies β with $\beta \hat{\ } 0 \subseteq \xi$, and let \mathcal{B}_0 be the set of all use blocks used by some $\beta \in \mathcal{C}$ to destroy a functional at the current stage.

Check if

- (2) $\Theta^D(x) \downarrow = 0,$
- (3) $\Theta^D \uparrow (x + i + 1) \downarrow = A \uparrow (x + i + 1),$
- (4) $\forall \beta \in \mathcal{C}(i_\beta > \vartheta(x + i)),$
- (4') $\forall B \in \mathcal{B}_0(|B| > i \cdot (\vartheta(x + i) + 1)),$ and
- (5) $\forall \mathcal{S}_U\text{-strategies } \alpha \subset \beta \forall s'$
 $(s^* \leq s' \leq s + 1 \rightarrow U_{\alpha, s'} \uparrow (\vartheta(x + i) + 1) = U_{\alpha, s^*} \uparrow (\vartheta(x + i) + 1))$

where $s^* = \max\{s' \leq s \mid s' = 0 \text{ or } \xi \text{ was eligible to act at stage } s'\}.$

If no then end ξ 's action at this stage by letting $\xi \hat{\ } 1$ be eligible next.

Otherwise, let \mathcal{B}_1 be the set of all current $\gamma_\alpha(w)$ -use blocks (such that some requirement $\mathcal{S}_{\bar{\gamma}}$ is active at ξ via α) and of all current $\delta_\beta(w)$ -use blocks (such that some requirement $\mathcal{S}_{\bar{\delta}}$ is satisfied at ξ via β). Check if

- (6) $\forall B \in \mathcal{B}_1(|B| > \vartheta(x)).$

If yes, say ξ *stops by itself*, put x into A , and let ξ end the stage.

Otherwise, let \mathcal{B}_2 be the set of all use blocks $B \in \mathcal{B}_1$ corresponding to functionals built by α or $\mathcal{R}_{\bar{\delta}}$ -destroyer strategies β with $\alpha \subset \beta \hat{\ } 0 \leq \xi$. Put the least unused element of each $B \in \mathcal{B}_2$ into D (we say ξ *uses* B to destroy the functional corresponding to B); increment i by $+1$; and handle $\Delta^D \uparrow (i + 1)$ by subcases the same way an \mathcal{S}_U -strategy handles $\Gamma^{D \oplus U} \uparrow (s + 1)$ in Case 1. End ξ 's action at this stage by letting $\xi \hat{\ } 0$ be eligible next.

Case 3. ξ is an \mathcal{R}_Θ -controller strategy. (Recall the definitions of the β_l 's, etc., at the end of the previous section.) We distinguish two subcases:

Case 3a. ξ is currently not taking charge of other strategies. First, check if ξ 's witness x is undefined. If so then redefine it to a number greater than any number mentioned thus far in the construction.

Next, denote by i_0, \dots, i_{l_0-1} the parameters i of $\beta_0, \dots, \beta_{l_0-1}$, respectively; let \mathcal{B}_3 be the set of all use blocks used by some β_l (for $0 \leq l < l_0$) to destroy a functional at the current stage; and check if

- (7) $\forall l < l_0(i_l > \max\{x, 3j_0 + 1\}),$
- (8) $\Theta^D(x) \downarrow = 0,$
- (9) $\forall B \in \mathcal{B}_3(|B| > 3(j_0 + 1)(\vartheta(x) + 1) + 2),$ and
- (10) $\forall \mathcal{S}_U\text{-strategies } \alpha \subset \gamma \forall s'$
 $(s^* \leq s' \leq s + 1 \rightarrow U_{\alpha, s'} \uparrow (\vartheta(x) + 1) = U_{\alpha, s^*} \uparrow (\vartheta(x) + 1))$

where $s^* = \max\{s' \leq s \mid s' = 0 \text{ or } \xi \text{ was eligible to act at stage } s'\}$.

If not then end ξ 's action at this stage by letting $\xi \wedge 1$ be eligible to act next. Otherwise, say ξ is taking charge of $\beta_0, \dots, \beta_{l_0-1}$; put x into A ; set $y_l = \max\{\vartheta_{\beta_{l'}}(x) \mid l \leq l' \leq l_0\}$ (for $l \leq l_0$); set $s_* = s$; say ξ does not stop any of $\beta_0, \dots, \beta_{l_0-1}$; set $l_* = l_0$; and end the stage.

Case 3b. ξ is currently taking charge of $\beta_0, \dots, \beta_{l_0-1}$. First reset l_* to be the greatest $l \leq l_0$ such that for all $j < |\sigma_l|$

$$(1a) \quad \sigma_l(j) = 0 \rightarrow U_{j,s} \upharpoonright (y_{L(l,j)} + 1) \neq U_{j,s_*} \upharpoonright (y_{L(l,j)} + 1)$$

and

$$(1b) \quad \sigma_l(j) = 1 \rightarrow U_{j,s} \upharpoonright (y_{L(l,j)} + 1) = U_{j,s_*} \upharpoonright (y_{L(l,j)} + 1).$$

(Such l_* exists by the \mathcal{R}_Θ -Controller Strategy Decision Lemma at the end of the previous section. The intuition is that β_{l_*} 's requirement can be satisfied finitarily. If l_* has not changed since the last stage at which γ was eligible to act then no action will be taken by γ at this stage.)

Let $\mathcal{B}_4, \mathcal{B}_5$ be the sets of all use blocks used by some β_l (for $0 \leq l < l_*$ and $l_* \leq l < l_0$, respectively) to destroy a functional at stage s_* . Now restore

$$(11) \quad D \upharpoonright (y_{l_*} + 1) = D_{s_*} \upharpoonright (y_{l_*} + 1)$$

(by possibly extracting elements from D), and ensure

$$(12) \quad \forall B \in \mathcal{B}_4(D \cap B \neq D_{s_*} \cap B)$$

(by possibly putting the least unused element of B into D).

(We pause to verify in a tedious counting argument that it is possible to achieve (11) and (12). First of all, since at stage $s_* + 1$, ξ ended the stage, no other \mathcal{R}_Θ -controller strategy $\tilde{\xi}$ ever uses any use block of $\mathcal{B}_4 \cup \mathcal{B}_5$ as each β_l works on different use blocks when $\tilde{\xi}$ first takes charge (except that once some \mathcal{R}_Θ -controller strategy $\tilde{\xi} < \xi$ may restore D on these use blocks at which time ξ is initialized so that these use blocks are never again worked on). Furthermore, for any use block $B \in \mathcal{B}_4 \cup \mathcal{B}_5$ used by β_l , say, to destroy a functional, we have that the action of ξ on $D \cap B$ depends only on whether $l_* > l$, i.e. on whether (1a) and (1b) hold for some $l_* > l$. This in turn depends on whether certain initial segments of $U_j \upharpoonright (y_{l+1} + 1)$ for d.r.e. sets U_j are equal to initial segments of $U_{j,s_*} \upharpoonright (y_{l+1} + 1)$. For fixed j , this can become false and true again at most $y_{l+1} + 1$ many times, i.e. $(j_0 + 1) \cdot (y_{l+1} + 1)$ many times for all $j \leq j_0$ combined. For each such time, an S_U -strategy or an \mathcal{R}_Θ -strategy may put an element into $D \cap B$ at most twice to correct

$\Gamma^{D \oplus U}$ or Δ^D which must later be extracted. Finally, some \mathcal{R}_Θ -destroyer strategy β_l puts one element into $D \cap B$ to initially destroy $\Gamma^{D \oplus U}(w_{\beta_l})$ or $\Delta^D(w_{\beta_l})$. All this adds up to $3(j_0 + 1)(y_{l+1} + 1) + 1$ many times that D must be changed back and forth on B .

Now by (4') for β_{l+1} and (7),

$$(13) \quad |B| > i_{\beta_{l+1}} \cdot (\vartheta_{\beta_{l+1}}(i_{\beta_{l+1}}) + 1) \geq 3(j_0 + 1) \cdot (\vartheta_{\beta_{l+1}}(i_{\beta_{l+1}}) + 1),$$

or by (9), $|B| > 3(j_0 + 1)(\vartheta(x) + 1) + 2$. In the latter case, we are done; in the former case, it remains to show

$$(14) \quad \vartheta_{\beta_{l+1}}(i_{\beta_{l+1}}) \geq y_{l+1} = \max\{\vartheta_{\beta_{l'}}(x) \mid l < l' \leq l_0\}.$$

Now (14) follows by (7) for $l' = l + 1$, and by (4) and (7) for $l' > l + 1$ since

$$\vartheta_{\beta_{l+1}}(i_{\beta_{l+1}}) \geq i_{\beta_{l+1}} > \vartheta_{\beta_{l'}}(i_{\beta_{l'}}) \geq \vartheta_{\beta_{l'}}(x).$$

This concludes our counting argument.)

Furthermore, say ξ stops β_l for $l_* \leq l < l_0$, and ξ does not stop β_l for $0 \leq l < l_*$. If l_* has changed from the last time when ξ (re)set l_* then ξ ends the stage.

Otherwise, if some \mathcal{R}_Θ -destroyer strategy β let ξ act first then return to β 's action. Otherwise, end ξ 's action at this stage by letting $\xi \hat{=} 1$ be eligible to act next.

10. THE VERIFICATION

We need to verify that the above construction satisfies both the \mathcal{S}_U - and the \mathcal{R}_Θ -requirements. We start with the definition of, and a lemma about, the true path.

DEFINITION 3: The *true path* f of the construction is the path through T defined inductively as follows: Suppose $\xi = f \upharpoonright n$. Then:

- i) $f(n) = 0$ if ξ is an \mathcal{S}_U -strategy.
- ii) $f(n) = 1$ if ξ is an \mathcal{R}_Θ -controller strategy.
- iii) $f(n) = 0$ if ξ is an \mathcal{R}_Θ -destroyer strategy and $\xi \hat{=} 0$ is eligible to act at infinitely many stages; otherwise $f(n) = 1$.

LEMMA 3 (Finite Initialization Along the True Path Lemma). Any strategy $\xi \subset f$ is initialized or reset at most finitely often, and it is eligible to act at infinitely many stages.

PROOF: We proceed by induction on $|\xi|$. The lemma clearly holds for the \mathcal{S}_{U_0} -strategy \emptyset .

So suppose $|\xi| > 0$ and no $\eta \subset \xi$ is ever initialized or reset after stage s_0 . Let $\xi^- = \xi \upharpoonright (|\xi| - 1)$. Then ξ cannot be initialized at a stage $s > s_0$ unless $|\xi^-| = s$, ξ^- is an \mathcal{R}_Θ -controller strategy or \mathcal{R}_Θ -destroyer strategy and ends the stage s , or ξ^- is an \mathcal{R}_Θ -destroyer strategy, $\xi = \xi^- \hat{=} 1$, and $\xi^- \hat{=} 0$ is eligible to act at stage s . The first and second cases can occur at most finitely often after stage s_0 (since ξ^- can end the stage at most

finitely often before being initialized or reset again); and the third case can occur at most finitely often by the definition of f . Thus ξ cannot be initialized after some stage $s_1 \geq s_0$, and ξ is eligible to act infinitely often.

If ξ is not an \mathcal{R}_Θ -destroyer strategy then ξ cannot be reset after stage s_1 . If it is an \mathcal{R}_Θ -destroyer strategy then its parameter w cannot change after stage s_1 . Thus ξ cannot be reset once $K \upharpoonright w$ and $U_\alpha \upharpoonright w$ (for all \mathcal{S}_U -strategies $\alpha < \xi$ which have acted before stage s_1) have settled down. \square

In the next lemma, we will prove that the \mathcal{S}_U -requirements are satisfied. Recall that, by the Finite Injury and Satisfaction Along Any Path Lemma from Section 8, each requirement \mathcal{S}_U is either active at $f \upharpoonright n$ for almost all n or satisfied at $f \upharpoonright n$ for almost all n .

LEMMA 4 (\mathcal{S}_U -Satisfaction Lemma). i) *If \mathcal{S}_U is active at $f \upharpoonright n$ for almost all n then the requirement \mathcal{S}_U is satisfied.*

ii) *If \mathcal{S}_U is satisfied at $f \upharpoonright n$ for almost all n then the requirement \mathcal{S}_U is satisfied.*

PROOF: i) We will show $K = \Gamma^{D \oplus U}$ where $\Gamma = \Gamma_\alpha$ for the \mathcal{S}_U -strategy $\alpha \subset f$ such that \mathcal{S}_U is active at $f \upharpoonright n$ via α for all $n > |\alpha|$.

Since α is initialized or reset only finitely often, Γ will eventually not be made completely undefined by α . Since α is eligible to act at infinitely many stages and will correct $\Gamma^{D \oplus U} \upharpoonright (s+1)$ at each of these stages we see that $\Gamma^{D \oplus U}$ correctly computes K on its domain. It remains to show that $\Gamma^{D \oplus U}$ is total.

So suppose $\Gamma^{D \oplus U}(x) \uparrow$ for minimal x . Let s_0 be a stage by which $\Gamma^{D \oplus U} \upharpoonright x$ has been defined ($D \oplus U$)-correctly. Since $\Gamma^{D \oplus U}(x)$ becomes defined infinitely often and thus $\gamma(x) \rightarrow \infty$, D must change infinitely often on a $\gamma(x)$ -use block. Now the only strategies changing D on a $\gamma(x)$ -use block are α itself (to correct $\Gamma^{D \oplus U}(x)$), at most one \mathcal{R}_Θ -destroyer strategy $\beta \supset \alpha$ (for which $x = w_\beta$), and any \mathcal{R}_Θ -controller strategies γ with $\gamma <_L \beta$ or $\gamma \supseteq \beta \hat{\ } 0$. Since $\gamma(x) \rightarrow \infty$, α will eventually not correct $\Gamma^{D \oplus U}(x)$ any more. If β or any of the γ 's change D on a $\gamma(x)$ -use block infinitely often then $\beta \subset f$. (Note here that $\beta >_L f$ implies $w_\beta \rightarrow \infty$.) Now $\beta \hat{\ } 0 \subset f$ is impossible since \mathcal{S}_U is active at $f \upharpoonright n$ via α for all $n > |\beta|$. Thus $\beta \hat{\ } 1 \subset f$, and once $\beta \hat{\ } 1$ is no longer initialized, neither β nor any γ will change D on a $\gamma(x)$ -use block.

ii) We will show $U = \Delta^D$ where $\Delta = \Delta_\beta$ for the \mathcal{R}_Θ -destroyer strategy $\beta \subset f$ such that \mathcal{S}_U is satisfied at $f \upharpoonright n$ via β for all $n > |\beta|$.

Since β is initialized or reset only finitely often, Δ will eventually not be made completely undefined by β , and $i = i_\beta$ will not be set to 0. Since $\beta \hat{\ } 0 \subset f$ we have $i \rightarrow \infty$, and β can define and correct $\Delta^D \upharpoonright (i+1)$ infinitely often. Thus Δ^D correctly computes U on its domain. It remains to show that Δ^D is total.

So suppose $\Delta^D(x) \uparrow$ for minimal x . Let s_0 be a stage by which $\Delta^D \upharpoonright x$ has been defined D -correctly. Since $\Delta^D(x)$ becomes undefined infinitely often and thus $\delta(x) \rightarrow \infty$, D must

change infinitely often on a $\delta(x)$ -use block. Now the only strategies changing D on a $\delta(x)$ -use block are β itself (to correct $\Delta^D(x)$), at most one $\mathcal{R}_{\bar{\Theta}}$ -destroyer strategy $\bar{\beta} \supset \beta$ (trying to kill Δ with killing point $w_{\bar{\beta}} = x$), and any $\mathcal{R}_{\bar{\Theta}}$ -controller strategies γ with $\gamma <_L \bar{\beta}$ or $\gamma \supseteq \bar{\beta} \hat{\ } 0$ for this $\bar{\beta}$. Since $\delta(x) \rightarrow \infty$, β will eventually not correct $\Delta^D(x)$ any more. If $\bar{\beta}$ or any of the γ 's change D on a $\delta(x)$ -use block infinitely often then $\bar{\beta} \subset f$. (Note here that $\bar{\beta} >_L f$ implies $w_{\bar{\beta}} \rightarrow \infty$.) Now $\bar{\beta} \hat{\ } 0 \subset f$ is impossible since S_U is satisfied at $f \upharpoonright n$ via β for all $n > |\beta|$. Thus $\bar{\beta} \hat{\ } 1 \subset f$, and once $\bar{\beta} \hat{\ } 1$ is no longer initialized, neither $\bar{\beta}$ nor any γ will change D on any $\delta(x)$ -use block.

This completes the proof of the lemma. \square

We now turn to the \mathcal{R}_{Θ} -requirements. Recall that, by the Finite Injury and Satisfaction Along Any Path Lemma, each requirement \mathcal{R}_{Θ} is satisfied at $f \upharpoonright n$ via an \mathcal{R}_{Θ} -strategy for almost all n .

LEMMA 5 (\mathcal{R}_{Θ} -Satisfaction Lemma). *If \mathcal{R}_{Θ} is satisfied at $f \upharpoonright n$ via an \mathcal{R}_{Θ} -strategy ξ for all $n > |\xi|$ then the requirement \mathcal{R}_{Θ} is satisfied.*

PROOF: By hypothesis, we have $\xi \hat{\ } 1 \subset f$. Let s_0 be minimal such that ξ is not initialized or reset after stage s_0 . We distinguish five cases:

Case 1. ξ is an \mathcal{R}_{Θ} -destroyer strategy and stops by itself after stage s_0 , say, at stage $s_1 > s_0$. By initialization and our assumption on s_0 , only some S_U -strategy $\eta \subset \xi$ or some $\mathcal{R}_{\bar{\Theta}}$ -destroyer strategy η with $\eta \hat{\ } 0 \subseteq \xi$ can destroy $\Theta_{\xi, s_1}^D(x_{\xi}) \downarrow = 0 \neq 1 = A(x_{\xi})$.

So fix such an η . By (4) and our assumption on s_0 , the only way η could possibly destroy $\Theta_{\xi}^D(x_{\xi})[s_*]$ is by correcting Γ_{η} or Δ_{η} on some argument $x \geq w_{\xi}$. If no requirement is satisfied at ξ via η then some $\mathcal{R}_{\bar{\Theta}}$ -destroyer strategy β with $\eta \subseteq \beta \subset \beta \hat{\ } 0 \subseteq \xi$ destroys Γ_{η} or Δ_{η} . By (4'), we have $\min B > |B| > \vartheta_{\xi}(x_{\xi})$ at stage s_1 for the $\gamma_{\eta}(w_{\xi})$ - or $\delta_{\eta}(w_{\xi})$ -use block B . β changes D on B at stage s_1 , and by initialization and our assumption on s_0 , this change is permanent. Thus η will never want to change $D \upharpoonright (\vartheta_{\xi}(x_{\xi}) + 1)[s_*]$.

If some requirement is satisfied at ξ via η then by (6), $\min B > |B| > \vartheta_{\xi}(x_{\xi})$ at stage s_1 for the $\gamma_{\eta}(w_{\xi})$ - or $\delta_{\eta}(w_{\xi})$ -use block B , and by initialization and our assumption on s_0 , no $\mathcal{R}_{\bar{\Theta}}$ -controller strategy will ever cause the $\gamma_{\eta}(w_{\xi})$ - or $\delta_{\eta}(w_{\xi})$ -use to be lower at a stage $s > s_1$ than it was at stage s_1 . Thus again η will never want to change $D \upharpoonright (\vartheta_{\xi}(x_{\xi}) + 1)[s_*]$.

Case 2. ξ is an \mathcal{R}_{Θ} -destroyer strategy and neither stops by itself, nor is permanently stopped by some fixed $\mathcal{R}_{\bar{\Theta}}$ -controller strategy $\gamma \supseteq \xi \hat{\ } 0$, after stage s_0 . We first claim that ξ is eligible to act at infinitely many stages without being stopped by any $\mathcal{R}_{\bar{\Theta}}$ -controller strategy $\gamma \supseteq \xi \hat{\ } 0$. For the sake of a contradiction, assume there are only finitely many, and that s_1 is the largest such. Certainly $s_1 \geq s_0$ by initialization or resetting. Then after stage s_1 , no strategy $\eta \supseteq \xi \hat{\ } 0$ is eligible to act, and only $\mathcal{R}_{\bar{\Theta}}$ -controller strategies $\gamma \supseteq \xi \hat{\ } 0$ are allowed to act first by ξ or some $\xi' \subset \xi$. Whenever some such γ no longer stops ξ then all strategies $\eta > \gamma$ are initialized, and the next γ' to stop ξ must therefore satisfy

$\gamma' < \gamma$ and must have been eligible to act before stage s_1 . But there are only finitely many such γ ; so either one fixed such γ eventually stops ξ forever or ξ is no longer stopped, a contradiction.

Thus ξ must eventually be stuck waiting for (2) through (5) to happen for a fixed $x = x_\xi \notin A$. Suppose $\Theta_\xi^D = A$. Thus $\lim i_\xi < \infty$ and $\lim \vartheta_\xi(x_\xi + i_\xi) < \infty$ while $\lim s^* = \infty$, $\lim i_\beta = \infty$ for all $\beta \in \mathcal{C}$, and $\lim \min\{|B| \mid B \in \mathcal{B}_0\} = \infty$ since $\xi \subset f$. So (2) through (5) hold true at cofinitely many stages, a contradiction.

Case 3. ξ is an \mathcal{R}_Θ -controller strategy that never takes charge of other strategies after stage s_0 . Suppose $\Theta_\xi^D = A$. Thus $x = x_\xi \notin A$ and $\lim \vartheta_\xi(x) < \infty$ while $\lim i_l = \infty$ for all $l < l_0$, $\lim s^* = \infty$, and $\lim \min\{|B| \mid B \in \mathcal{B}_3\} = \infty$ since $\xi \subset f$. So (7) through (10) must hold at cofinitely many stages, a contradiction.

Cases 4 and 5. ξ is an \mathcal{R}_Θ -destroyer strategy that is permanently stopped by some fixed \mathcal{R}_Θ -controller strategy $\gamma \supseteq \xi \hat{\ } 0$ after stage s_0 ; or ξ is an \mathcal{R}_Θ -controller strategy γ that takes charge of other strategies after stage s_0 . We adopt γ 's notation in the following. Since $\gamma \leq f$, γ will be initialized or reset after stage s_0 only finitely often, say never after (a least) stage $s_1 \geq s_0$. Then γ takes charge of other strategies forever at some stage $s_* + 1 \geq s_1$. As in the counting argument in Case 3b of the construction, γ 's parameter l_* can change at most finitely often once γ takes charge of other strategies, so say l_* will not change after (a least) stage $s_2 \geq s_* + 1$.

By the minimality of s_2 , γ will ensure (11) and (12) for l_* at stage s_2 . By (11), we have $\Theta_\xi^D(x_\gamma)[s_2] \downarrow = 0 \neq 1 = A(x_\gamma)$. By initialization and our assumption on s_1 , only \mathcal{S}_U -strategies $\alpha \subset \xi$ and \mathcal{R}_Θ -destroyer strategies β with $\beta \hat{\ } 0 \subseteq \xi$ can possibly destroy the computation $\Theta_\xi^D(x_\gamma)$ after stage s_2 , and then only when correcting Γ_α or Δ_β on an argument $\geq \max\{w_\beta, \mid l < l_0\}$. Denote the set of these strategies by \mathcal{C}_0 . We will show that no $\eta \in \mathcal{C}_0$ will destroy $\Theta_\xi^D(x_\gamma)$ after stage s_2 .

Let \mathcal{C}_1 be the set of all \mathcal{S}_U -strategies $\alpha \in \mathcal{C}_0$ such that α 's requirement is not active at ξ via α , and of all \mathcal{R}_Θ -destroyer strategies $\beta \in \mathcal{C}_0$ (targeted to destroy some Γ_α for $\alpha \subset \xi$, say) such that α 's requirement is not satisfied at ξ via β . (These are exactly the $\eta \in \mathcal{C}_0$ the functional of which has been "destroyed before ξ ".) Let $\mathcal{C}_2 = \mathcal{C}_0 - \mathcal{C}_1$.

First consider a strategy $\eta \in \mathcal{C}_1$. Then the $\gamma_\alpha(w_{\beta_{l'}})$ - or $\delta_\beta(w_{\beta_{l'}})$ -use block B (for some $l' < l_*$) is in ξ 's set \mathcal{B}_0 (in Case 4), or \mathcal{B}_3 (in Case 5), at stage s_* ; and by (4') and (7) (in Case 4), or by (9) (in Case 5), we have $\min B > |B| > \vartheta_{\xi, s_*}(x_\gamma)$. By initialization, no \mathcal{R}_Θ -controller strategy $\bar{\gamma} \neq \gamma$ can restore D on B after stage s_* . Now B is in γ 's set \mathcal{B}_4 (since Γ_α or Δ_β is destroyed by $\beta_{l'}$). Thus by (12), D is permanently changed on B by γ at stage s_2 , and thus any $\gamma_\alpha(w)$ - or $\delta_\beta(w)$ -use block (for $w \geq w_{\beta_{l'}}$) that applies after stage s_2 exceeds $\vartheta_{\xi, s_*}(x_\gamma)$. Thus no $\eta \in \mathcal{C}_1$ can destroy $\Theta_\xi^D(x_\gamma)$ after stage s_2 .

Next consider an \mathcal{S}_{U_j} -strategy $\alpha \in \mathcal{C}_2$ (for some $j \leq j_0$). Then $\sigma_{l_*}(j) = 0$, and so by (1a), $U_{j, s} \upharpoonright (y_{\nu+1} + 1) \neq U_{j, s_*} \upharpoonright (y_{\nu+1} + 1)$ for all $s \geq s_2$ where $l' \geq l_*$ is minimal such

that $\beta_{l'}$ is targeted to destroy Γ_α . By (4') (for $\beta_{l'}$, $l' < l < l_0$), and by (7) and (9) for γ , $y_{l'+1} = \max\{\vartheta_{\beta_{l'}}(x_\gamma) \mid l' < l \leq l_0\} < |B| < \min B$ for the $\gamma_\alpha(w_{\beta_{l'}})$ -use block B that $\beta_{l'}$ uses to destroy Γ_α at stage $s_* + 1$.

Notice that whenever α newly defines $\Gamma_\alpha^{D \oplus U_j}(w)$ for some w then the $\gamma_\alpha(w)$ -use block is at least as big as the $\gamma_\alpha(w)$ -use block used at the last definition of $\Gamma_\alpha^{D \oplus U_j}(w)$, and bigger if D has changed on the previous $\gamma_\alpha(w)$ -use block. Thus any new definition of $\Gamma_\alpha^{D \oplus U_j}(w)$ (for any $w \geq w_{\beta_{l'}}$) after stage s_* must use a $\gamma_\alpha(w)$ -use block B' with $\min B' > |B'| > \vartheta_{\xi, s_*}(x_\gamma)$. For any $\gamma_\alpha(w_{\beta_{l'}})$ -use block B' with $\min B' < \min B$, $\beta_{l'}$ has destroyed that definition, and if it was ever restored by some $\mathcal{S}_{\bar{U}}$ -strategy $\bar{\gamma} \supseteq \beta_{l'} \hat{\ } 0$ then that restoration was only temporary (else $\beta_{l'}$ would be permanently stopped by $\bar{\gamma} \neq \gamma$). Any definition of $\Gamma_\alpha^{D \oplus U_j}(w_{\beta_{l'}})$ using use block B must have occurred between $\beta_{l'+1}$'s stage s^* (as measured at stage $s_* + 1$) and stage $s_* + 1$ (since any definition from before stage s^* has been permanently destroyed by stage s_2). Now by (5) or (10) for $\beta_{l'+1}$

$$\forall s \forall t (s^* \leq s \leq s_* + 1 \ \& \ t \geq s_2 \rightarrow U_{j,s} \upharpoonright (y_{l'+1} + 1) \neq U_{j,t} \upharpoonright (y_{l'+1} + 1)),$$

and $y_{l'+1} < \min B$ by the above; so any definition using use block B never applies after stage s_2 . Thus no \mathcal{S}_{U_j} -strategy $\alpha \in \mathcal{C}_2$ can destroy $\Theta_\xi^D(x_\gamma)$ after stage s_2 .

Finally, consider an \mathcal{R}_{Θ} -destroyer strategy $\beta \in \mathcal{C}_2$. Then $\sigma_{l_*}(j) = 1$, and so by (1b), $U_{j,s} \upharpoonright (y_{l_*} + 1) = U_{j,s_*} \upharpoonright (y_{l_*} + 1)$ for all $s \geq s_2$. Thus when γ restores $D \upharpoonright (y_{l_*} + 1)$ at stage s_2 (by (11)) then $\Delta^D \upharpoonright (y_{l_*} + 1)$ is correct, and β will not correct it after stage s_2 , so it will not change $D \upharpoonright (\delta(y_{l_*}) + 1)$ or, a fortiori, $D \upharpoonright (\vartheta_\xi(x_\gamma) + 1)$. Thus no \mathcal{R}_{Θ} -destroyer strategy $\beta \in \mathcal{C}_2$ can destroy $\Theta_\xi^D(x_\gamma)$ after stage s_2 .

We have thus established $\Theta_\xi^D(x_\gamma) \neq A(x_\gamma)$ in all five cases.

This concludes the proof of the lemma. □

Lemmas 1, 4, and 5 establish the D.r.e. Nondensity Theorem. □

Notice, incidentally, that, since $D \oplus U \geq_T K$ for some U , there will be only finitely many \mathcal{R}_{Θ} -controller strategies along the true path. However, their role is crucial, of course, since the \mathcal{R}_{Θ} -controller strategies to the *left* of the true path really do all the hard work whereas the \mathcal{R}_{Θ} -destroyer strategies along the true path only satisfy \mathcal{R}_{Θ} by themselves when they “accidentally” hit upon a Θ -computation that is clear of Γ - and Δ -uses.

11. THE PROOF OF THE n -R.E./ ω -R.E. NONDENSITY THEOREM

Notice that the only fact we use about the U 's in the above proof is that we have an effective bound on the number of times that $U_s \upharpoonright z = U_{s_*} \upharpoonright z$ can become false and true again as s increases (for fixed s_* , z). For a d.r.e. set U , this number is z . For an n -r.e. set U (for $n \geq 2$), this number is certainly bounded by nz . Thus inserting a factor $j_0 + 1$ in the right-hand sides of (4), (4'), (7), and (9) makes the counting argument in Case 3b of the construction above work, assuming that all sets U_0, \dots, U_{j_0} are $(j_0 + 1)$ -r.e. Thus (11)

and (12) can be established whenever necessary. (The same argument works for the ω -r.e. degrees.)

REFERENCES

1. M. M. Arslanov, *Structural properties of the degrees below \mathbf{O}'* , Dokl. Akad. Nauk SSSR, N.S. 283 2 (1985), 270-273.
2. ———, *On structural properties of degrees of Ershov's hierarchy*, Izv. VUZov No. 7 (1988).
3. S. B. Cooper, S. Lempp, and P. Watson, *Weak density and cupping in the d-r.e. degrees*, Israel J. Math. 67 (1989), 137-152.
4. R. G. Downey, *D.r.e. degrees and the Nondiamond Theorem*, Bull. London Math. Soc. 21 (1989), 43-50.
5. R. L. Epstein, R. Haas, and R. Kramer, *Hierarchies of sets and degrees below \mathbf{O}'* , in "Logic Year 1979-80: University of Connecticut (M. Lerman, J. H. Schmerl, R. I. Soare, eds.)", Springer-Verlag, Berlin (1981), 32-48.
6. Yu. L. Ershov, *On a hierarchy of sets*, I/II/III, Algebra i Logika 7 (1968), 47-73/7 (1968), 15-47/9 (1970), 34-51.
7. C. G. Jockusch, Jr., M. Lerman, R. I. Soare, and R. M. Solovay, *Recursively enumerable sets modulo iterated jumps and extensions of Arslanov's Completeness Criterion*, J. Symbolic Logic 54 (1989), 1288-1323.
8. C. G. Jockusch, Jr., and R. I. Soare, *Minimal covers and arithmetical sets*, Proc. Amer. Math. Soc. 25 (1970), 856-859.
9. A. H. Lachlan, *Lower bounds for pairs of recursively enumerable degrees*, Proc. London Math. Soc. 16 (1966), 537-569.
10. ———, *A recursively enumerable degree which will not split over all lesser ones*, Ann. Math. Logic 9 (1975), 307-365.
11. D. Miller, *High recursively enumerable degrees and the anti-cupping property*, in "Logic Year 1979-80: University of Connecticut (M. Lerman, J. H. Schmerl, R. I. Soare, eds.)", Springer-Verlag, Berlin (1981), 230-245.
12. H. Putnam, *Trial and error predicates and the solution to a problem of Mostowski*, J. Symbolic Logic 30 (1965), 49-57.
13. G. E. Sacks, *The recursively enumerable degrees are dense*, Ann. of Math. (2) 80 (1964), 300-312.
14. T. A. Slaman, *The recursively enumerable degrees as a substructure of the Δ_2^0 -degrees*, to appear.
15. R. I. Soare, "Recursively Enumerable Sets and Degrees," Perspectives in Mathematical Logic, Omega Series, Springer-Verlag, Berlin, 1987.

Keywords. d.r.e. degrees, nondensity, \mathbf{O}''' -priority method

1980 *Mathematics subject classifications:* 03D25, 03D30

School of Mathematics, University of Leeds, Leeds, LS2 9JT, England

Department of Mathematics, University of California, Berkeley, CA 94720, USA

Department of Mathematics, Simon Fraser University, Burnaby, B.C. V5A 1S6, Canada

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

Department of Mathematics, University of Chicago, Chicago, IL 60637, USA