ON A DIFFERENCE HIERARCHY FOR ARITHMETICAL SETS

ALEXANDER MELNIKOV, VICTOR L. SELIVANOV, AND MARS M. YAMALEEV

Abstract. In the late 1980s, Selivanov used typed Boolean combinations of arithmetical sets to extended the Ershov hierarchy beyond Δ^0_2 sets. Similar to the Ershov hierarchy, Selivanov's *fine hierarchy* $\{\Sigma_\gamma\}_{\gamma<\epsilon_0}$ proceeds through transfinite levels below ϵ_0 to cover all arithmetical sets. In this paper we use a $0'''$ construction to show that the Σ^0_3 Turing degrees (aka the $\Sigma_{\omega^\omega+1}$-degrees) are *properly* contained in the $\Sigma_{\omega^\omega+2}$ Turing degrees (to be defined); intuitively, the latter class consists of "non-uniformly Σ^0_3 sets" in the sense that will be clarified in the introduction. The question whether the hierarchy was proper at this level with respect to Turing reducibility has been open for over 20 years.

1. Introduction

The study of various hierarchies of sets is central to mathematical logic. In computability theory, one investigates hierarchies of countable sets. This paper contributes to a general program in computability theory that seeks to understand various – not necessarily algorithmic – hierarchies using algorithmic tools such as Turing degrees. (Recall that a Turing degree consists of sets that are indistinguishable from the perspective of Turing computation.)

It is quite remarkable that many natural hierarchies admit several equivalent definitions, one usually being syntactical, and the other topological or computability-theoretic (or both). There are many examples of this sort, including the classical hierarchies [Gao09, Sel08, Sac90] as well as some very recently introduced ones [Por17, DG18]. Perhaps, the most famous illustration of this phenomenon is provided by the the well-known positive solution to Hilbert's Xth Problem [Mat93]: the Diophantine sets are exactly the *computably enumerable* sets which form the first level Σ^0_1 of the arithmetical hierarchy. The second level Σ^0_2 of the hierarchy consists of $\exists\forall$-definable subsets of $(\mathbb{N},+,\times,0)$ which are exactly the sets computably enumerable with the help of the Halting problem. The Π^0_2-sets are the complements of the Σ^0_2-sets, and $\Delta^0_2=\Sigma^0_2\cap\Pi^0_2$.

The Schoenfeld Limit Lemma [Soa87] says that a set is in Δ^0_2 exactly if it can be computably approximated with finitely many mind changes; this effective approximability of Δ^0_2 sets makes them relatively accessible for recursion-theoretic investigations. In fact, the study of Turing degrees of Δ^0_2 sets is one of the central topics of recursion theory [Soa87]. The structure of Δ^0_2-degrees turned out to be quite complex, with the theory of Σ^0_1-degrees being its most developed and well-understood subtheory [Soa87].

If we cannot fully grasp the structure of Δ^0_2-degrees, it is natural to look at some finer intermediate complexity classes between the Σ^0_1 and the Δ^0_2-degrees

2010 *Mathematics Subject Classification.* Primary . Secondary .
The first author was partially supported by Marsden Fund of New Zealand.

and try to extend the more that we know about Σ^0_1-degrees to these intermediate classes. However, the arithmetical hierarchy is too crude to allow for such a subtle distinction, we need a more sensitive sub-hierarchy. One such finer hierarchy is the difference hierarchy, also known as the Ershov hierarchy [Es68a, Es68b, Es70]. The hierarchy looks at the Boolean combinations of computably enumerable (c.e. for short) sets and measures the complexity of the Boolean forms. The hierarchy has an equivalent "dynamic" definition which measures the number of mind-changes needed to compute a Δ^0_2-set via the Schoenfeld Limit Lemma. The c.e. sets form the first level $\Sigma^{-1}_1 = \Sigma^0_1$ of the Ershov hierarchy, while the differences of c.e. sets – also known as *d*-c.e. sets – form its second level Σ^{-1}_2. The hierarchy $\{\Sigma^{-1}_\alpha\}$ proceeds through all computable ordinals to cover all Δ^0_2 sets. There has been many deep results on the first few non-trivial levels of the hierarchy, especially the *d*-c.e. degrees, see the PhD thesis of Wu [Wu02] and the more recent surveys by Arslanov and Kalimullin [AK17] and Lemmp [Lem14]. Less is known about the higher transfinite levels of the difference hierarchy, but there have been some recent impressive results on the closely related difference hierarchy introduced by Downey and Greenberg, see their book [DG18].

Since the investigation of the difference hierarchy from the perspective of Turing degrees has been rather fruitful, one naturally seeks an extension – or at least an analogy – of such results beyond the Δ^0_2 degrees. Clearly, one naive possibility would be looking at relativised versions of the results on Σ^{-1}_α sets. However, lots of subtle information such as a dynamic approximation will be lost after a relativisation, thus such relativised results seem to be of little interest. We need a more sensitive hierarchy than the relativised Ershov hierarchy.

In [Sel89], Selivanov introduced a natural extension of the Ershov hierarchy to sets beyond Δ^0_2, as follows. Consider the *typed* Boolean combinations over variables $\{v_{i,k}\}_{i,k\in\omega}$, where for each fixed k the variables $\{v_{i,k}\}_{i\in\omega}$ range over Σ^0_k-sets. Then for every *typed* Boolean term $\mathbf{t}$ form the class $\Sigma(\mathbf{t}) = \{\mathbf{t}(v_{i,j}) : v_{i,j} \in \Sigma^0_j\}$. Each such term has a *rank* which is naturally represented as an ordinal below $\epsilon_0 = \sup\{\omega, \omega^\omega, \omega^{\omega^\omega}, \ldots\}$; we omit the details since they are irrelevant to our main result; see [Sel95]. Similarly to the Ershov hierarchy, Selivanov defined classes Σ_α for $\alpha < \epsilon_0$ according to the rank of the typed term defining the class. He also proved that the definition is robust and the hierarchy is proper with respect to m-degrees.

As we have mentioned above, all natural and useful hierarchies tend to admit several equivalent definitions, and the Selivanov's fine hierarchy is not an exception. There are several equivalent ways to define the hierarchy, one of which, remarkably, uses a slight variation of the Wadge's operator *bicep* [Sel08]; this gives a technical connection between the fine hierarchy in recursion theory, the Wadge hierarchy [Wad83] in descriptive set theory, and the Wagner hierarchy [Wag79] in automata theory.

Selivanov called the new hierarchy *the fine hierarchy.* The hierarchy is "fine" in the sense that it is (provably) more sensitive than just the relativised Ershov hierarchy within Δ^0_n, for any $n > 2$. The relativised version of the Ershov hierarchy at a level Δ^0_n, $n > 2$, does not take into account the mind changes at the lower levels, thus erasing much of the subtle complexity related to *dynamic approximations.* Recall that we were searching for a natural extension of the Ershov hierarchy beyond Δ^0_2 sets. The fine hierarchy seems to be the most natural choice here.

Whenever we have a hierarchy of sets, the general expectation is that the more complex classes of the hierarchy should contain more oracles, up to Turing equivalence. In this case we say that the hierarchy is proper (i.e., does not collapse) with respect to Turing degrees. In the early 1970s, Cooper [Cop71] constructed a d.c.e. set which is not Turing equivalent to any c.e. set. Cooper's result can be extended to all levels of the Ershov hierarchy [JS84, Sel85], illustrating that the Ershov hierarchy is proper from the perspective of Turing reducibility.

Which levels of the fine hierarchy are proper with respect to Turing degrees? One would hope for some inductive proof – e.g., by the complexity of the typed terms – that would extend the argument of Cooper [Cop71] to all levels of the fine hierarchy, but this seems to be highly problematic. Somewhat unexpectedly, the situation with the fine hierarchy is more complicated than with the Ershov hierarchy. It seems that any construction of a set in Σ_α not Turing equivalent to a set in any simpler class would *necessarily* require a guessing at the level of the highest type of the variable involved in the defining term. The dynamic feature which is so attractive about the fine hierarchy also makes the task quite complicated. As we will see, the problem of properness of the fine hierarchy with respect to Turing degrees requires an essential use of advanced degree-theoretic techniques. However, degree theory was not very popular in the former Eastern block; very few people within the former USSR mastered the infinite injury technique, let alone the Lachlan Monster method [Soa87]. As a consequence, the question of which levels of the hierarchy are proper w.r.t. Turing degrees remained unresolved, and essentially completely open, for a few decades.

Despite of these technical and cultural challenges, some partial progress has been done into this direction. For instance, it is known that all levels $\Sigma_{\lambda+1}$ ($\lambda < \varepsilon_0$ is a limit ordinal) are not proper and that level $\Sigma_{\omega+2}$ is proper [SY], also each $\Sigma_{\omega+n+2}$, $n < \omega$, is proper; see [Sel08, SY] for a survey and for further results. However, there are still many levels of the fine hierarchy for which the situation with the Turing degrees remained, and still remains, unknown. We note that the results listed above used constructions of complexity at most $0''$. The first level at which a $0'''$ guessing seemed necessary was $\Sigma_{\omega^\omega+2}$ vs. $\Sigma_{\omega^\omega+1}$, making the question of properness at this level of a special importance and also of some independent technical interest. We explain what these classes $\Sigma_{\omega^\omega+2}$ and $\Sigma_{\omega^\omega+1}$ are below.

The sets in $\Sigma_{\omega^\omega+1}$ have the same Turing degrees as the Σ^0_3-sets [SY]; thus, we will not define the class. The next simplest class that can potentially contain more oracles is $\Sigma_{\omega^\omega+2}$. What are these sets? To enter a set from this class, each number $x \in \omega$ goes through the following guessing procedure, in the spirit of mind-changes in the Ershov hierarchy. Initially, it could be declared $x \in X$, otherwise x is kept out of X forever. Then, at a later stage, x can be extracted form X and given a choice of two independent guessing procedures, Π^0_3 and Σ^0_3. We must dynamically choose one of these two procedures, and we will put x back into X only if it passes the guessing that we chose. (The formal definition will be given in due course.)

In some sense, the guessing on the membership for such sets is still at the level of only three quantifiers, but we get to *dynamically* choose between a Σ^0_3- and a Π^0_3-guessing procedure. These sets a almost Σ^0_3, there is only a subtle difference. As we noted above, all previous results on the fine hierarchy used arguments of complexity at most $0''$. The reader will perhaps agree that, because of the dynamic nature of the above definition, the use of a $0'''$-construction of some sort is likely necessary in any proof of our main result below.

Theorem. There exist a set in the class $\Sigma_{\omega^\omega+2}$ which is not Turing equivalent to any set in $\Sigma_{\omega^\omega+1}(=\Sigma_{\omega^\omega})$.

A very careful choice of the setup and a thoughtful use of self-referrential predicates on the tree of strategies allowed us to reduce the combinatorial complexity of the guessing. But it is still a $0'''$ proof, with true Π^0_3-outcomes and an auxiliary priority on the nodes of the tree.

We conjecture that our ideas can be pushed to cover levels of the form $\Sigma_{\omega^\gamma+2}$. The hope is that we can potentially describe all proper levels of the fine hierarchy, but it will likely require a complicated iterated priority method, such as the ones introduced by Lempp and Lerman [LL97] and Ash [Ash86]. We also hope that both the hierarchy and our methods might find interesting applications in degree theory and computable structure theory. For example, can we find a syntactical description of the subsets of a computable structure that are relatively intrinsically Σ_α, in the sense of [AK00]?

2. The main strategy

The naive basic idea is obvious, but its implementation is far from being straightforward. We shall exploit the extra freedom that a $\Sigma_{\omega^\omega+2}$-set has in the choice of the Π^0_3 vs. Σ^0_3-guessing and try to "do the opposite" to what a given Σ^0_3-set D suggests on a given witness. The difficulty will clearly be related to the Σ^0_3 guessing on what D says. It has to be dynamic because we also need to build a c.e. partition of the universe that contains the Σ^0_3 vs. Π^0_3 guessing pieces. In fact, one of the main roles of the tree will be in showing that what we have constructed lies in the class $\Sigma_{\omega^\omega+2}$ and not in some more general class. It is not that easy to explain informally what exactly will have to be done.

We open the discussion with the description of the basic strategy below. We then discuss the situation of only two strategies, and then when we have to deal with several strategies. Only then we give the formal construction. An experienced reader will hopefully have no problem in seeing the general technical idea quite early in the proof.

2.1. The requirements.

We have to construct disjoint c.e. sets E_0, E_1, a Σ^0_3-set $A \subseteq E_0$, a Π^0_3-set $B \subseteq E_1$, and a c.e. set E such that $(E_0 \cup E_1) \subset E$, and the set $S = A \cup B \cup C$ is not Turing equivalent to any Σ^0_3-set D, where $C = E \backslash (E_0 \cup E_1)$. To prove the theorem, we need to satisfy the requirements

$$\mathcal{R}_k = \mathcal{R}_{\Phi,\Psi,D}\colon A \cup B \cup C \neq \Phi^D \vee D \neq \Psi^{A\cup B\cup C},$$

where $\langle \Phi, \Psi, D\rangle = \langle \Phi_k, \Psi_k, D_k\rangle$ is the kth element in a fixed computable enumeration of all such triples, where Φ_k, Ψ_k are computable functionals and D_k are Σ^0_3-sets (represented by their Σ^0_3-indices in some fixed enumeration, to be clarified).

2.2. **One strategy in isolation.** For simplicity, assume that we enumerate elements into C instead of E. Hence, whenever we extract an element from C we immediately put it either into E_0 or E_1.

A strategy for $\mathcal{R}_{\Phi,\Psi,D}$.

(1) Choose a fresh witness x.

As usual, "fresh" means that the number that we pick has never been seen in the construction so far.

(2) Wait for a stage s_0 and a string τ of length $< s_0$ such that $\Phi^\tau(x)[s_0] = (A \cup B \cup C)(x)[s_0] = 0 \wedge \Psi^{A\cup B\cup C} \upharpoonright |\tau|[s_0] = \tau$.

If $\Phi^D(x) = (A\cup B\cup C)(x) \wedge \Psi^{A\cup B\cup C} \upharpoonright \varphi^D(x) = D \upharpoonright \varphi^D(x)$ is indeed the case then such a string must exist. Otherwise, if D does not satisfy the condition then the requirement will be met. Note that, in general, τ does not need to be an initial segment of D.

(3) Put x into S at stage $s_0 + 1$.

In particular, we will have $C(x)[s_0+1] = 1$ and $C(x)[s_0] = 0$. Thus, $\Phi^\tau(x)[s_0] = 0 = (A \cup B \cup C)(x)[s_0]$ and $\Psi^{A\cup B\cup C} \upharpoonright |\tau|[s_0] = \tau$, but $\Phi^\tau(x)[s_0+1] = 0 \neq 1 = (A \cup B \cup C)(x)[s_0+1]$. Perhaps, $D \upharpoonright |\tau|[s_0] = D \upharpoonright |\tau| = \tau$ but not necessarily so. Nonetheless, we make an attempt of diagonalization by changing $(A\cup B\cup C)(x)[s]$. We will see that this attempt will either succeed, or we will find a new string σ with a similar property but incomparable with τ. If the former case the requirement will be met (think about the initial segment of D).

(4) Wait for a stage $s_1 > s_0$ and search for a string σ of length $< s_1$ such that $\Phi^\sigma(x)[s_1] = (A \cup B \cup C)(x)[s_1] = 1 \wedge \Psi^{A\cup B\cup C} \upharpoonright |\sigma|[s_1] = \sigma$.

As above, either σ exists or the requirement is met. Let z_x be least such that $\tau(z_x) \neq \sigma(z_x)$. Note that τ and σ must be incomparable, and

$$\Psi^{A\cup B\cup C}(z_x)[s_0]\downarrow = \tau(z_x) \neq \Psi^{A\cup B\cup C}(z_x)[s_1]\downarrow = \sigma(z_x).$$

Thus, the value $\Psi^{A\cup B\cup C}(z_x)$ is now under our control. *We will use these two strings to attack against D at z_x. Note that have have not yet used the Σ^0_3 guessing anywhere in the strategy.*

(5) Fix z_x least such that $\tau(z_x) \neq \sigma(z_x)$. Consider the two cases:

$(C1)$ $\tau(z_x) = 1$,
$(C2)$ $\tau(z_x) = 0$.

– If $(C1)$ holds, then enumerate x into E_0. (In particular, extract from C.) At the following stages $s > s_1$ declare $A(x) = D(z_x)$.

– If $(C2)$ holds, then declare $x \in E_1$. At the following stages $s > s_1$ declare $B(x)[s] = 1 - D(z_x)$.

In $(C1)$, the computation $\Psi^{A\cup B\cup C}(z_x)[s_0]\downarrow = \tau(z_x) = 1$ halts under the assumption that x is *not* in our set $A \cup B \cup C$, while $\tau(z_x) \neq \Psi^{A\cup B\cup C}(z_x)[s_1]\downarrow = \sigma(z_x) = 0$ halts under the assumption that x is in the set (recall we put x into C at stage s_0+1). In particular, if we can ensure that $z_x \in D$ iff $x \in A$ then the requirement will be met. Formally, we let A decide on the membership of x by enumerating $x \in E_0$. We will dynamically correct the Σ^0_3-definition of $A(x)$ (which currently was set equal to, say, 0) to mimic the Σ^0_3-procedure for $D(z_x)$. This can be done with all possible uniformity. The case $(C2)$ is dual to $(C1)$. In particular, we have $\Psi^{A\cup B\cup C}(z_x)[s_0]\downarrow = \tau(z_x) = 0$ under the assumption that x is *not* in our set $A \cup B \cup C$. So we better make sure that

$z_x \in D$ iff $x \notin A$. This is done by allowing $B(x)$ to dynamically copy the Π^0_3-definition of $\overline{D}(z_x)$ from this stage on.

Finally, whenever the strategy finds a suitable Ψ-computation, it attempts to *preserve* the initial segment of the computation.

This ends the description of one strategy in isolation.

2.2.1. *The outcomes.* The outcomes of one strategy in isolation will have to be modified.

- Σ: The outcome is played if τ and σ have been defined and $\tau(z_x) = 1$. In this case we will produce a Σ^0_3-definition for the witness membership.
- Π : The outcome is played if τ and σ have been defined and $\tau(z_x) = 1$. In this case we will produce a Π^0_3-definition for the witness membership.
- fin_1 : The outcome is played if τ has not yet been defined. In this case the witness is currently kept outside of the set S.
- fin_2 : The outcome is played if τ has not yet been defined. In this case the witness will be kept in S until a suitable σ is found.

It is clear that one strategy in isolation ensures that the requirement is met in each of the three cases. Note that the outcomes above are finitary in nature, but they will have to be further modified to incorporate a Π^0_3-guessing.

3. Combining two strategies

Clearly, the main problem with (say) two strategies, one working on top of the other, is that the weaker priority strategy will have to *guess* on the membership of the witness of the higher priority strategy. We have not yet defined the actual outcomes yet, but we will do so very shortly. To explain why we need to modify the outcomes, we consider the case of only two strategies. We will introduce more and more modifications as we go.

Remark 3.1. Before we start, we note that the recursion theorem will be used throughout (and typically without explicit reference) to justify some of the uniform changes in the tree and in the strategies. However, the recursion theorem it is not really necessary for this proof. Although various predicates will be frequently modified, it will be quite clear what these uniform modifications will be, so the s-m-n theorem will actually suffice, but that will require some thought. The reader is free to choose the approach that suites them better.

Suppose we have only two strategies, ρ and ρ', one working under the other, with ρ having a higher priority. According to its instructions, ρ' does not really have to know anything about its Σ^0_3-set $R_{\rho'}$ to search for strings τ and σ (see the previous section). Recall that the Σ^0_3-definition of the set will be used *only* after some suitable σ and τ strings are found (if ever). Then, if successful in its search, the strategy will attempt to preserve an initial segment of the set. Nonetheless, S may be effected by the actions of the higher priority ρ which may have its witness x_ρ below the restraint of ρ'. The definition of the membership relation for the witness x_ρ depends on the outcome of ρ and could be Π^0_3 at worst.

3.1. **ρ' below Σ of ρ.** This means that ρ has found its strings and has adopted a Σ^0_3-definition for the relation $x_\rho \in S$; the index of the definition is given to us dynamically in the construction. For ρ, seeing whether $x_\rho \in S$ or not is a Σ^0_3 vs. Π^0_3

piece of information. To make the guessing comprehensible, we split the outcome Σ further into the suboutcomes

$$\Sigma_{\Pi_2,0} < \Sigma_{\Pi_2,1} < \ldots < \Sigma_{\Pi_2,n} < \ldots\ldots < \Sigma_{\Pi_3},$$

with the Π_3 version measuring the Π^0_3 outcome $x \notin S$, and $\Sigma_{\Pi_2,n}$ measuring the number of witnesses t for $\exists^\infty t P(x,t,n)$, in

$$x \in S \iff \exists n \, \exists^\infty t \, P(x,t,n),$$

where P is the recursive kernel of the Σ^0_3 membership predicate for x whose index is known (by the recursion theorem or using the uniformity of the construction). As usual, in $\exists n \, \exists^\infty t \, P(x,t,n)$ we can assume that such an n, if it exists, is unique.

3.2. **The outcomes of ρ.** In presence of only two strategies it is sufficient to have the outcomes discussed above and the ordering:

$$\text{the } \Sigma \text{ outcomes } \ldots < \text{the } \Pi \text{ outcomes } \ldots < fin_2 < fin_1,$$

where the ordering on the Σ and the Π outcomes has already been discussed.

3.3. **The first attempt to coordinate nodes.** Naively, it may seem that all we have to do is to clone ρ' below each of the outcomes:

- ρ' below $\Sigma_{\Pi_2,n}$. This version of ρ' will believe that $x_\rho \in S$ and will act accordingly whenever visited.
- ρ' below Σ_{Π_3}. This version of ρ' will believe that $x_\rho \notin S$ and will act accordingly whenever visited.

(We will see that this will not be sufficient in presence of many strategies.) As it is typical in such constructions, we may wish to visit the Π^0_2-outcome $\Sigma_{\Pi_2,n}$ whenever the predicate "fires" again on n. Assuming that for each n the predicate fires at least once, the Σ_{Π_3}-outcome will then be visited in-between the stages in which the Π^0_2 outcomes are played.

This naive set-up brings us to the following *potential conflict*. It could be the case that the clone of ρ' below the Π^0_3-outcome has already chosen it witness. The other versions of the strategy must be able to live with this because the Π^0_3-outcome will be visited infinitely often in any case.

3.4. **Coordinating nodes of two strategies.**

3.4.1. ρ' *below* $\Sigma_{\Pi_2,n}$. To circumvent the potential problem discussed above, each node below $\Sigma_{\Pi_2,n}$ will be associated with a *state* which is a symbol for the set $\Pi, \Sigma, fin_2, fin_1$. The symbol provides the node with the information on the current outcome of the node below the Π^0_3-outcome.

Case 1: ρ' having state fin_1 believes that the version of the strategy below the Π^0_3 outcome will never find a suitable string τ (see the instructions of the basic strategy in isolation) and therefore its witness will be permanently outside of S; ρ' will act accordingly.

Case 2: ρ' having state fin_2 believes that the strategy below the Π^0_3 outcome has found its τ, but will never find a suitable σ. Therefore, its witness will be permanently kept in S; ρ' will act accordingly.

Case 3: ρ' having state Σ believes that the strategy below the Π^0_3 outcome has found both its σ and τ, its witness has been temporarily removed from S and can be put back according to a Σ^0_3-definition.

Modification 3.2. The clone of ρ'' below Σ_{Π_3} will be *pressed* as follows. Replace the Π^0_3 non-membership definition of $x_{\rho''} \notin S$ by the conjunction of it with the statement that the Π^0_3 outcome Σ_{Π_3} above ρ'' is the true outcome of ρ. Then, if this is *not* the true outcome then $x_{\rho''}$ is *in the set*.

In this case ρ' below $\Sigma_{\Pi_2,n,s3}$ assumes that the witness of the clone below Σ_{Π_3} is in the set S, and it will act according to this belief.

Case 4: ρ' having state Π believes that the clone of ρ' below the Π^0_3-outcome has chosen a Π^0_3-definition for the membership of its witness.

Modification 3.3. Below Π_{Π_3}, we replace the Π^0_3 definition for the witness of the clone with the conjunction of itself with the Π^0_3 predicate saying that the Σ_{Π_3}-outcome of the higher priority strategy ρ holds.

Then ρ' below $\Sigma_{\Pi_2,n,p3}$ assumes that the witness of its clone below Σ_{Π_3} is *not in the set*. It will act according to this belief.

A state of a strategy ρ' is changed is the outcome of ρ'' below the respective Π^0_3-outcome changes (even if ρ' is currently not being played). A strategy will be initialised as soon as it changes its state.

3.4.2. *ρ' below Π of ρ.* This outcome is dual to the Σ outcome in the sense that the membership of the witness of ρ is declared Π^0_3 in the construction. Note that each strategy ρ' below Π of ρ needs to know only the value of the membership relation $x_\rho \in S$, it does not really matter whether the witness of ρ is in or is out of S as long as ρ' *knows* which of the two possibilities holds. As a consequence, the analysis of this outcome is the same as that of Σ above. The modifications that need to be done to the various versions of ρ' below ρ are the same, mutatis mutandis, as the modifications discussed above for the Σ-case.

3.4.3. *ρ' below fin_1 or fin_2 of ρ.* In this case the strategy ρ' believes the current guess of ρ on the membership of its witness (x_ρ is out of and in S, respectively). This is a finitary outcome, no further modification is needed.

3.4.4. *Intended Initialisation.* A node is initialised by: (1) declaring its previous witness to be permanently out of the set; (2) choosing a new fresh and large witness; (3) setting all other parameters undefined. Initialise all clones to the right of the currently active one, but never initialise the clones below the Σ_{Π_3} and the Π_{Π_3} outcomes of ρ. Also, initialise a node if it has changed its state.

Note that each strategy can be upset by the node below a Π^0_3-outcome at most finitely many times. This notion of a state will have to be modified the general case. We will see that a thoughtful definition of a state will allow us to further simplify the notion of initialisation.

4. Several requirements

In presence of many requirements we need to cope with further tensions that were not visible in the case of only two requirements. For example, in presence of many requirements, the strategy below the nth Π^0_2 outcome will have to monitor several witnesses of the strategies below the Π^0_3-outcome, but it cannot possibly handle infinitely many such witnesses.

Currently we have not yet defined the tree, but it will be done in due course. There will be no danger of circularity.

4.1. Auxiliary priority.

To resolve the above mentioned difficulty and some other problems (to be discussed), we introduce an auxiliary priority on the nodes in the tree, as follows.

Definition 4.1 (Auxiliary local priority)**.** *Introduce an effective priority order, of order-type ω, on all the nodes in the tree. The order must satisfy the property: if ρ' is below ρ then the auxiliary priority of ρ' is weaker than that of ρ*

It will not matter how exactly we order the nodes. For example, the nodes that become active earlier in the construction could be given smaller ω-indices in the order.

4.2. States.

Every node ρ on the tree will have at most finitely many nodes of higher auxiliary priority that could potentially upset its instructions by either restraining some part of S or changing their mind on their own witness. Suppose there are exactly m such nodes.

Each node ρ will be associated with a *state* which is an m-tuple of symbols Σ, P, fin_2, fin_1. The ith symbol in the tuple reflects the most recent outcome of the ith node of higher auxiliary priority. Note that the Σ and Π outcomes of these higher auxiliary priority nodes will not be subdivided when listed in a state.

4.2.1. *The interpretation of a state.*

A current state of ρ allows it to have the best available guess on the witnesses of at most m higher priority nodes to the right of it, in the spirit of 3.4.1. More specifically, those higher priority nodes that are playing Π and fin_1 will be assumed to have their witnesses out, and for those which have recently been playing Σ and fin_2 their witnesses will assumed to be in.

4.2.2. *Other witnesses.*

The current witnesses of weaker auxiliary priority strategies will be assumed to be out or in the set, depending what makes the computation of the higher priority strategy halt.

Remark 4.2. The stronger priority strategy will be searching for a suitable initial segment of the set S that makes its computation halt. This initial segment may contain witnesses $x_{\rho'}$ for lots of weaker ρ'. It is too hard to guess on the membership status of $x_{\rho'}$, there are too many of such weaker strategies. Therefore, the stronger priority strategy ρ will be searching for *some* computation assuming that the membership status of such $x_{\rho'}$ is up to the stronger strategy ρ to decide, if necessary.

As soon as a new suitable computation is found by ρ (if ever) the weaker auxiliary priority nodes will be instantly initialised and their witnesses will be extracted.

4.3. **Conflicts and initialisation.** Two nodes, ρ and ρ', are said to have a conflict at a stage if one of the two has a witness below the restraint of the other. (Note it does not have to be known whether the witness is in or out.)

Write $\alpha \prec \beta$ if the auxiliary priority of α is weaker than the auxiliary priority of β, and we write r_α to denote the restraint of α (meaning that it currently attempts to preserve the interval $[0, r_\alpha]$). Consider the cases:

(C1): $\alpha \prec \beta$ and $x_\beta \leq r_\alpha$. Then α will use its *state* to have a guess on $x_\rho \in S$.
(C2): $\alpha \prec \beta$ and $x_\alpha \leq r_\beta$. Initialise the weaker auxiliary priority α in this case.

Since a node above α on the tree must be of a higher auxiliary priority than α, the cases above apply to the analysis along the current true path as well (to be defined).

4.3.1. *Initialisation.* Initialise the node whenever it changes its state, or of it has a conflict of type $(C2)$ with some other node of a higher auxiliary priority. As before, the state can changed even if the node is not currently being played.

4.3.2. *Abandoned witnesses.* We must clarify what happens with a witnesses after it is abandoned by its strategy. Let x be such witness. After it is abandoned by ρ, the old witness may be put back to the set and then extracted, and then put back again, etc., in the Δ^0_2-manner. This process will be regulated by strategies having higher (auxiliary) priority that ρ, as follows. When a higher priority strategy ρ' is searching for a halting computation it decides on the membership status of x and assumes it is either in or our of the set, according to what makes the computation halt. The membership status of x is then *preserved* with the priority of ρ'. We will see that every strategy on the tree will be initialised at most finitely many times, and therefore this process will eventually terminate, and the membership of x will be finally stably defined.

This finishes the intuitive description of some elements of the construction. We are ready to give the formal construction and its verification. Before we do so, note that the initialisation above is somewhat suspicious: we did not initialise anything to the right of the current true path in the spirit of the standard infinite injury proof. We used only the states and conflicts, all being finitary in their nature. Quite interestingly, we will show that this degenerate initialisation will be sufficient.

5. Construction

5.1. **The tree of strategies.** The tree will be infinitely branching. Consider

$$\Sigma_{\Pi_2,0} < \ldots < \Sigma_{\Pi_2,n} < \ldots$$

and

$$\Pi_{\Pi_2,0} < \ldots < \Pi_{\Pi_2,n} < \ldots$$

Call the sequences the $\Sigma_{\Pi_2,n}$-outcomes and the $\Pi_{\Pi_2,n}$-outcomes, respectively. Then form the infinite ordered sequences:

the $\Sigma_{\Pi_2,0}$-outcomes $<$ the $\Sigma_{\Pi_2,1}$-outcomes $< \ldots <$ the $\Sigma_{\Pi_2,n}$-outcomes$< \ldots < \Sigma_{\Pi_3,1}$,

the $\Pi_{\Pi_2,0}$-outcomes $<$ the $\Pi_{\Pi_2,1}$-outcomes $< \ldots <$ the $\Pi_{\Pi_2,n}$-outcomes$< \ldots < \Pi_{\Pi_3,1}$,

and call them the Σ-outcomes and the Π-outcomes of ρ, respectively. Finally, define

$$\text{the } \Sigma \text{ outcomes } \ldots < \text{the } \Pi \text{ outcomes } \ldots < fin_2 < fin_1,$$

to be the collection of all outcomes/successors of ρ.

5.2. **The auxiliary priority and states.** Recall that in 4.1 we defined an *auxiliary priority* on all the nodes in the tree to be a computable ordering of the nodes of order-type ω, with the smaller indices corresponding to the higher auxiliary priority nodes. Also, if ρ' is below ρ in the tree then the auxiliary priority of ρ' is weaker than that of ρ (the latter is written $\rho' \prec \rho$).

Also, suppose ρ has exactly m different β such that $\rho \prec \beta$. Associate ρ with its *state* which is an m-tuple listing the current Σ, Π, fin_2, and fin_1 outcomes of all these m nodes (ignoring the sub-outcomes of Σ and Π).

5.3. **The modified strategy for ρ.** Suppose ρ is a node on the tree. Then ρ will be associated with a version of the basic strategy of the $|\rho|$th requirement.

(1) If ρ' is below ρ in the tree, then ρ will assume that the witnesses of the weaker priority ρ' is either in or out of S, depending on what makes the computation of the stronger ρ halt.
(2) If $\rho' \succ \rho$ then the relation $x_{\rho'}$ will be guessed based by ρ on the respective parameter in the ρ-state. More specifically, if the respective parameter is Σ or fin_2 then ρ will assume $x_{\rho'} \in S$, and otherwise (if it is Π or fin_1) ρ will assume $x_{\rho'} \notin S$.
(3) If ρ' is above ρ in the tree, then $x_\rho \in S$ will be guessed naturally based on the outcome of ρ' above ρ.

The actions of (the strategy for) ρ will be similar to those discussed in Section 2, but based on the guesses above and with the following modifications below (see also 3.2 and 3.3).

(1) Suppose the strategy chooses to play Π. Then replace the naive original Π^0_3-definition for $x_\rho \in S$ with the conjunction of it with the predicate saying that ρ is on the true path.
(2) Suppose the strategy chooses to play Σ. Then replace the naive original Π^0_3-definition for $x_\rho \notin S$ with the conjunction of it with the predicate saying that ρ is on the true path.
(3) In both cases we may assume that the Π^0_3 predicate has the unique existential witness property. (In particular, there is at most one Π^0_2-witness in the Σ^0_3-case.)

5.4. **The current true path.** In the previous two sections we have already explained how each of the outcomes of a strategy ρ is played. We give a brief summary:

(1) The outcomes fin_1 and fin_2 are finitary and are played when ρ waits for τ_ρ and for σ_ρ, respectively. See also 2.2.1 for their description.
(2) The Σ and Π outcomes (in the sense of 5.1) are played only if both σ_ρ and τ_ρ have been found and ρ has chosen to adopt either a Σ^0_3 or a Π^0_3-definition for its witness x_ρ (respectively). In the former case only the Σ-outcomes of ρ will be played, and in the latter case only its Π outcomes will be played. Suppose ρ plays its Σ outcomes. This is done as follows:
   - $\Sigma_{\Pi_2,n}$ is played if the nth column of the measured predicate fires again.
   - The Σ_{Π_3} is played in-between the stages at which the various $\Sigma_{\Pi_2,n,\gamma_i}$-outcomes are played.

   The case of Π is similar, up to a change of notation.

The definition of the current true path is usual. Simultaneously with the current true path and for every node in the tree (not necessarily along the current true path), define its state according to the description above.

5.5. **Initialisation.** Initialise a node ρ whenever it changes its state, or there is a node $\alpha \succ \rho$ such that $x_\rho \leq r_\alpha$ (i.e., the witness of ρ is below the restraint of α). with some other node of a higher auxiliary priority. As noted before, the state of ρ can changed even if the node is not currently being played. A node is initialised by picking new and fresh witness. (The membership of the old witness will be decided in the construction by the finitely many higher priority strategies.)

5.6. **Construction.** At stage 0 initialise all strategies. At stage s, let the strategies along the current true path act according to their (modified) instructions.

6. Verification

First, we will show that the diagonalisation is successful (i.e., each of the requirements is met), and then we argue that the set that we construct belongs to the desired difference class of the fine hierarchy.

The set S consists of all the witnesses of the strategies on the tree. The true path is the left-most path visited infinitely often. Clearly, the top node is on the true path. We need to check that the true path is infinite. The *true outcome* of a node is the outcome for which the predicate that it measures holds. This does not depend whether a node is on the true path or not, and it does not really matter what the predicate means. The below lemma is central to the verification.

Lemma 6.1.
1. Every node in the tree is initialised at most finitely many times in the construction.
2. Every node in the tree has its state eventually stable.

Proof. By simulteneous induction, and using the auxiliary priority ordering.

The root of the tree satisfies the lemma, assuming it has the top local priority. Additionally, at some stage it permanently settles on either Σ or Π or fin_2 or fin_1 (ignoring the sub-outcomes). Its state is the empty string. Suppose we have proven the lemma for the first n nodes having the highest auxiliary priority. Let ρ be the next highest (auxiliary) priority node. There is a stage after which the first n nodes forever settle on either Σ or Π or fin_i. This means that they have computed their strings (if they could) and have a stable witnesses.

It follows that after this stage ρ cannot be initialised by its state. Also, it can be initialised at most once by a higher auxiliary priority node. After it becomes active again (if ever), it will choose its witness larger than the restraints of all these strategies of a higher auxiliary priority. Thus, it will never be initialised again in the construction. □

Lemma 6.2. *The true path is infinite.*

Proof. Induction on the length of the true path δ. We show that it has length at least n, and that the first n nodes will eventually be never initialised. The *true outcome* of a node is the outcome for which the predicate that it measures holds. Regardless of what exactly it measures, we split the respective predicate into complementary sub-outcomes. If o is the true outcome of a node ρ on the true path, then $\rho\widehat{\ }o$ is also on the true path. □

Lemma 6.3. *Every requirement is met along the true path.*

Proof. Recall that the original predicate for the membership of the witness x_ρ had to be modified. Nonetheless, *provided that ρ is no the true path*, the modifications described in 5.3 lead to logically equivalent predicate. All we have to check is that the computations of ρ are correct, in the sense that its guess on the initial segment of the set is actually correct. Assume ρ is never initialised after step s.

The proof proceeds by cases on the position of ρ' with respect to ρ. We must see if ρ has a correct guess on $x_{\rho'} \in S$ (when necessary and after stage s):

(1) ρ' is above ρ. Then the guess on $x_{\rho'}$ is provided by the true path.
(2) ρ' has a weaker auxiliary priority than ρ. (This includes the case when ρ' is below ρ.) Then ρ ignores the witness of ρ' and assume it is in or out, depending on what suits ρ better. This is fine because ρ' will be initialised anyway if the necessary computation is found by ρ. Thus, it will be up to ρ (or some even higher priority strategy) to decide what happens with the witness of ρ'.
(3) ρ' has a stronger auxiliary priority than ρ but is off the true path.
   - ρ' to the left of ρ: It will eventually be abandoned in the construction. If it stays with Σ or Π then, because of the adopted modifications, ρ will have a correct guess on the membership of $x_{\rho'}$ in S. It is crucial that ρ' is off the true path and its predicate reflects it.
   - ρ' to the right of ρ: Then ρ will be using its state to determine the outcome of the node. After some stage s the state is final (Lemma 6.1(2)), and because of the modifications to the predicate, the state provides ρ with a correct guess on $x_{\rho'}$. As before, it is important that ρ' is off the true path.

It follows that ρ can safely act according to its instructions, and its computations (if ρ ever finishes them) will be correct. The basic module of ρ ensures that in any case the requirement $\mathcal{R}_{|\rho|}$ is met.

It remains to check that the set S belongs to $\Sigma_{\omega^\omega+2}$. This means that we must find:

- disjoint c.e. sets E_0, E_1,
- a Σ^0_3-set $A \subseteq E_0$,
- a Π^0_3-set $B \subseteq E_1$, and
- a c.e. set E such that $(E_0 \cup E_1) \subset E$,

such that $S = A \cup B \cup [E \setminus (E_0 \cup E_1)]$. The meanings of these sets are the following:

- if $x \in E \setminus (E_0 \cup E_1)$ that we have put x into the set;
- if x is put into E_0 then we have chosen a Σ^0_3-definition;
- if x is put into E_1 then we have chosen a Π^0_3-definition.

If ρ was the only strategy, and its witness x_ρ was the only potential element of the set S, then the instructions of the basic module of ρ guarantee that x_ρ go through the necessary phases with all possible uniformity.

In presence of many strategies, the intended Π^0_3-definition of a node had to be modified. Nonetheless, we have already noted that the use of the recursion theorem or even a careful analysis of uniformities in the construction will allow for a transformation into a logically equivalent predicate *provided that the node ρ is on the true path.* If the node is off the true path then the predicate is still of complexity Π^0_3, we need only to argue that the combined complexity of all such definitions will be correct, i.e., there is a uniform Π^0_3 definition of (say) B above.

In the construction, make sure that only very large numbers can be picked as fresh witnesses, and if a number was skipped in this process then it is permanently out. For every point x, if it is not in E_1 then it is out of B. The set E_1 is trivially c.e., by the construction, thus saying that $x \in E_1$ is simply Σ^0_1.

Remark 6.4. It may seem that the former few statements (regarding E_1 and a stage) give one extra $\exists$-predicate, but this is not the case. As usual, define a partial function to give the desired parameters z_x and the stage s if they exist. Its *index* is given by a total computable function, so there is no problem with the recursion theorem either.

Furthermore, we are only interested in the case when $x \in E_1$ because there is a stage s and a node ρ that enters its Π-phase with witness $x_\rho = x$; this is a Σ^0_1-information that can be stated. Furthermore, recall that $x = x_\rho \in E_1$ means that there is a witness z_x whose $D_{|\rho|}$-membership will be tested by ρ. This definition will have to be adjusted by a self-referential predicate and then transformed into one with the unique existential witness property. Furthermore, the predicate must also incorporate the possibility of ρ being initialised, in which case the predicate will be actually imitate a Δ^0_2-precess with at most finitely many mind changes (to be clarified below). This definition is again uniform in the stage of the construction, and has complexity at most Π^0_3 being a finite conjunction of statements of complexity at most Π^0_3. This gives a Π^0_3 description of the set B. Clearly, $B \subseteq E_1$. The set A has a similar description, but with a minor modification below related to initialisation.

We must clarify what happens during initialisation from the point of view of the set that we are constructing. Suppose a strategy ρ is initialised, in which case x_ρ will be abandoned and redefined. We shall abuse notation and keep denoting the old value by x_ρ. In this case x_ρ is permanently put into (say) E_0 unless it is already in E_1. Suppose it is put into E_0, the case of E_1 is symmetric. The Π^0_3 membership predicate will be dynamically adjusted according to the needs of the finitely many higher priority strategies. There are only finitely many nodes having higher auxiliary priority than ρ, and these nodes may need (the old) x_ρ to be in or out of the set, according to their needs. This gives a uniformly Δ^0_2 membership definition for the old witness of ρ (in fact, an *n-c.e.* definition, where n depends on the number of higher priority nodes). We use this dynamic Δ^0_2 process to modify the Π^0_3 membership predicate for this old witness.

We conclude that S belongs to the right difference class, and therefore the theorem is proved. $\square$

References

[AK00] C. Ash and J. Knight. *Computable structures and the hyperarithmetical hierarchy*, volume 144 of *Studies in Logic and the Foundations of Mathematics*. North-Holland Publishing Co., Amsterdam, 2000.

[AK17] Marat M. Arslanov and Iskander Sh. Kalimullin. A survey of results on the d-c.e. and n-c.e. degrees. In *Computability and complexity*, volume 10010 of *Lecture Notes in Comput. Sci.*, pages 469–478. Springer, Cham, 2017.

[Ash86] C. Ash. Recursive labeling systems and stability of recursive structures in hyperarithmetical degrees. *Trans. Amer. Math. Soc.*, 298:497–514, 1986.

[Cop71] Barry Copper. *Degrees of unsolvability.* PhD thesis, Leicester University, Leicester, England, 1971.

[DG18] Rod Downy and Noam Greenberg. A transfinite hierarchy of lowness notions in the computably enumerable degrees, unifying classes, and natural definability. preprint, 2018.

[Es68a] Ju. L. Erˇ sov. A certain hierarchy of sets. I. *Algebra i Logika*, 7(1):47–74, 1968.

[Es68b] Ju. L. Erˇ sov. A certain hierarchy of sets. II. *Algebra i Logika*, 7(4):15–47, 1968.

[Es70] Ju. L. Erˇ sov. A certain hierarchy of sets. III. *Algebra i Logika*, 9:34–51, 1970.

[Gao09] Su Gao. *Invariant descriptive set theory*, volume 293 of *Pure and Applied Mathematics (Boca Raton).* CRC Press, Boca Raton, FL, 2009.

[JS84] Carl G. Jockusch, Jr. and Richard A. Shore. Pseudojump operators. II. Transfinite iterations, hierarchies and minimal covers. *J. Symbolic Logic*, 49(4):1205–1236, 1984.

[Lem14] S. Lempp. A survey of results on the d.c.e. and n-c.e. degrees. *Lobachevskii J. Math.*, 35(4):313–316, 2014.

[LL97] Steffen Lempp and Manuel Lerman. Iterated trees of strategies and priority arguments. *Arch. Math. Logic*, 36(4-5):297–312, 1997. Sacks Symposium (Cambridge, MA, 1993).

[Mat93] Yuri V. Matiyasevich. *Hilbert's tenth problem.* Foundations of Computing Series. MIT Press, Cambridge, MA, 1993. Translated from the 1993 Russian original by the author, With a foreword by Martin Davis.

[Por17] Michelle. Porter. *Aspects of Computable Analysis, MSc Thesis.* Victoria University of Wellington, 2017.

[Rog87] H. Rogers. *Theory of recursive functions and effective computability.* MIT Press, Cambridge, MA, second edition, 1987.

[Sac90] Gerald E. Sacks. *Higher recursion theory.* Perspectives in Mathematical Logic. Springer-Verlag, Berlin, 1990.

[Sel85] V. L. Selivanov. The Ershov hierarchy. *Sibirsk. Mat. Zh.*, 26(1):134–149, 1985.

[Sel89] V. L. Selivanov. Fine hierarchies of arithmetic sets, and definable index sets. *Trudy Inst. Mat. (Novosibirsk)*, 12(Mat. Logika i Algoritm. Probl.):165–185, 190, 1989.

[Sel95] V. L. Selivanov. Fine hierarchies and Boolean terms. *J. Symbolic Logic*, 60(1):289–317, 1995.

[Sel08] Victor L. Selivanov. Fine hierarchies and m-reducibilities in theoretical computer science. *Theoret. Comput. Sci.*, 405(1-2):116–163, 2008.

[Soa87] R. Soare. *Recursively enumerable sets and degrees.* Perspectives in Mathematical Logic. Springer-Verlag, Berlin, 1987. A study of computable functions and computably generated sets.

[SY] V.L. Selivanov and M.M. Yamaleev. On turing degrees in refinements of the arithmetical hierarchy. *Algebra i Logika (to appear).*

[Wad83] William Wilfred Wadge. *Reducibility and determinateness on the Baire space.* ProQuest LLC, Ann Arbor, MI, 1983. Thesis (Ph.D.)–University of California, Berkeley.

[Wag79] Klaus Wagner. On ω-regular sets. *Inform. and Control*, 43(2):123–177, 1979.

[Wu02] GuoHua Wu. *Structural Properties of d.c.e. Degrees and Presentations of c.e. Reals.* PhD thesis, Victoria University of Wellington, 2002.

Massey University, Auckland, New Zealand
E-mail address: alexander.g.melnikov@gmail.com

A.P. Ershov Institute of Informatics Systems SB RAS, Novosibirsk, Russia
E-mail address: vseliv@iis.nsk.su

Kazan (Volga region) Federal University, Kazan, Russia
E-mail address: Mars.Yamaleev@kpfu.ru