Borel combinatorics goes wrong in HYP

Henry Towsner

University of Pennsylvania

May 30, 2020

Joint work with Rose Weisshaar and Linda Brown Westrick

Reverse math is concerned with trying to measure which axioms are necessary to prove theorems.

One complication is that sometimes it takes a lot of work just to construct the objects and prove that they behave in a reasonable way.

Reverse math is concerned with trying to measure which axioms are necessary to prove theorems.

One complication is that sometimes it takes a lot of work just to construct the objects and prove that they behave in a reasonable way.

One example is the Borel sets, which require some coding to make sense of: to even proof some basic facts about codes for Borel sets, we already need the theory ATR_0.

Reverse mathematics is usually done in the framework of second order arithmetic. The only objects are natural numbers and sets of natural numbers. Everything else has to be encoded in some way.

Reverse mathematics is usually done in the framework of second order arithmetic. The only objects are natural numbers and sets of natural numbers. Everything else has to be encoded in some way.

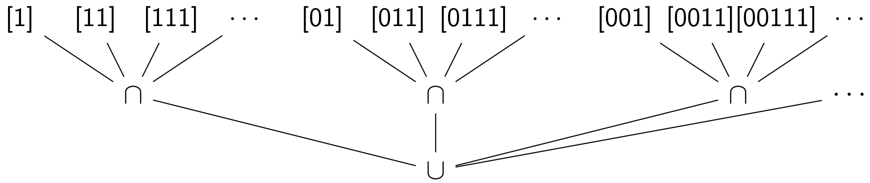
A Borel set is a subset of $\{0,1\}^\omega$ where:

- each clopen set $[\sigma] = \{X \mid \forall i < |\sigma|\ X(i) = \sigma(i)\}$ is Borel, and
- a countable union or intersection or Borel sets is Borel.

A *Borel code* is a *tree*—that is, a set of sequences closed under subsequences—such that:

- the tree is well-founded,
- each leaf encodes a basic clopen subset of $\{0,1\}^\omega$,
- each interior node encodes either a union or an intersection.

For instance:



encodes the set of sequences of the form $0000\cdots 01111\cdots$.

If T is a Borel code and X is an infinite sequence, we need to make sense of the question

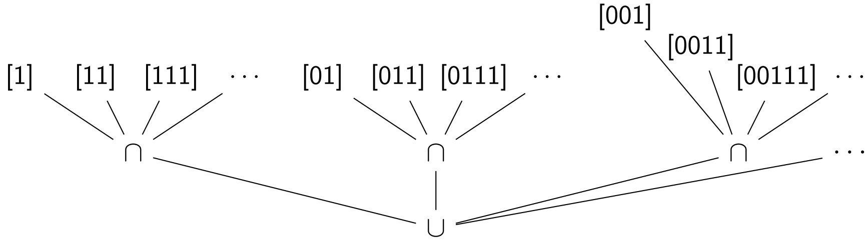
does X belong to the set encoded by T?

If T is a Borel code and X is an infinite sequence, we need to make sense of the question

does X belong to the set encoded by T?

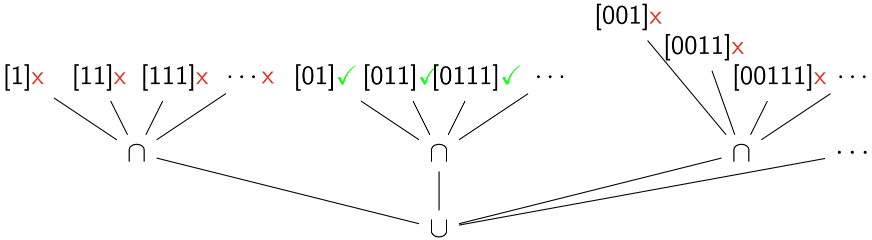
We would normally do this by transfinite induction: we should be able to tell which (sets encoded by) leaves X belongs to and then we can tell if X belongs to (the set encoded by) a node by asking whether it belongs to some/all of the children.

For instance, to see that $011\cdots$ belongs to



we can:

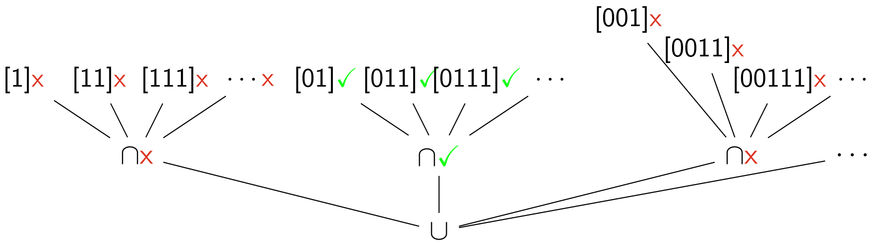
For instance, to see that $011\cdots$ belongs to



we can:

1. check which leaves it belongs to,

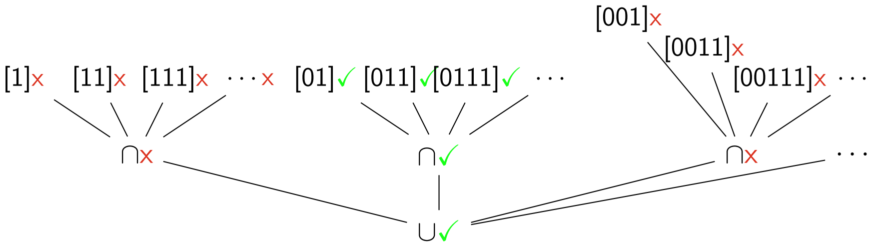
For instance, to see that $011\cdots$ belongs to



we can:

1. check which leaves it belongs to,
2. check which nodes at the next level it belongs to,

For instance, to see that $011\cdots$ belongs to



we can:

1. check which leaves it belongs to,
2. check which nodes at the next level it belongs to,
3. check if it belongs to the root.

Definition

When T is a Borel code, an *evaluation map* for X is a function $f : T \to \{0, 1\}$ such that:

- if σ is a leaf, $f(\sigma) = 1$ if and only if X is in the clopen set encoded at σ,
- if σ is a union node, $f(\sigma) = 1$ if and only if $f(\sigma^\frown n)$ for some n,
- if σ is an intersection node, $f(\sigma) = 1$ if and only if $f(\sigma^\frown n)$ for all n.

Definition

When T is a Borel code, an *evaluation map* for X is a function $f : T \to \{0, 1\}$ such that:

- if σ is a leaf, $f(\sigma) = 1$ if and only if X is in the clopen set encoded at σ,
- if σ is a union node, $f(\sigma) = 1$ if and only if $f(\sigma^\frown n)$ for some n,
- if σ is an intersection node, $f(\sigma) = 1$ if and only if $f(\sigma^\frown n)$ for all n.

An evaluation map must be unique, so we can say:

- $X \in T$ if there is an evaluation map f so that $f(\langle\rangle) = 1$,
- $X \notin T$ if there is an evaluation map f so that $f(\langle\rangle) = 0$.

But an evaluation map is itself (encoded by) a set, and we need to be able to prove they exist. Constructing an evaluation map requires a transfinite recursion—which is exactly what we need ATR_0 for:

Theorem

Working over RCA_0, if, for every Borel code T and every X, an evaluation map for X exists, then ATR_0 holds.

But an evaluation map is itself (encoded by) a set, and we need to be able to prove they exist. Constructing an evaluation map requires a transfinite recursion—which is exactly what we need ATR_0 for:

Theorem

Working over RCA_0, if, for every Borel code T and every X, an evaluation map for X exists, then ATR_0 holds.

Even "every Borel code is either non-empty, or has a non-empty complement" already implies ATR_0.

Astor-Dzhafarov-Montalbán-Solomon-Westrick pointed out that instead of folding the work of finding evaluation maps into our proofs—requiring all our proofs to use ATR_0—we could instead make evaluation maps part of the definition.

Astor-Dzhafarov-Montalbán-Solomon-Westrick pointed out that instead of folding the work of finding evaluation maps into our proofs—requiring all our proofs to use ATR_0—we could instead make evaluation maps part of the definition.

Definition

T is a *completely determined Borel code* if, for all X, an evaluation map for X exists.

Astor-Dzhafarov-Montalbán-Solomon-Westrick pointed out that instead of folding the work of finding evaluation maps into our proofs—requiring all our proofs to use ATR_0—we could instead make evaluation maps part of the definition.

Definition

T is a *completely determined Borel code* if, for all X, an evaluation map for X exists.

When trying to measure the strength of various theorems about Borel sets, maybe the correct hypothesis to work with is not "for every Borel set" but instead "for every completely determined Borel set".

ADMSW and Westrick studied principles like:

- Every completely determined Borel set has the property of Baire,
- Every completely determined Borel set is measurable.

ADMSW and Westrick studied principles like:

- Every completely determined Borel set has the property of Baire,
- Every completely determined Borel set is measurable.

They showed that these properties are strictly weaker than ATR_0, but that these principles still imply (over RCA_0 or WWKL) a principle called $\mathrm{L}_{\omega_1,\omega}$-$\mathrm{CA}$.

ADMSW and Westrick studied principles like:

- Every completely determined Borel set has the property of Baire,
- Every completely determined Borel set is measurable.

They showed that these properties are strictly weaker than ATR_0, but that these principles still imply (over RCA_0 or WWKL) a principle called $\mathrm{L}_{\omega_1,\omega}$-$\mathrm{CA}$.

This puts these principles in a range that includes the “theories of hyperarithmetic analysis”—theories closely linked to what happens in ω-models whose sets are exactly $HYP(Y)$ for some Y.

Question

Do statements about completely determined Borel sets behave reasonably in HYP?

HYP is the model of second order arithmetic in which the sets are exactly the hyperarithmetic sets.

Question

Do statements about completely determined Borel sets behave reasonably in HYP?

HYP is the model of second order arithmetic in which the sets are exactly the hyperarithmetic sets.

If T is a hyperarithmetic Borel code and X is a hyperarithmetic set, T has an ordinal height α, and we can compute an evaluation map for X in α or so jumps of X.

Question

Do statements about completely determined Borel sets behave reasonably in HYP?

HYP is the model of second order arithmetic in which the sets are exactly the hyperarithmetic sets.

If T is a hyperarithmetic Borel code and X is a hyperarithmetic set, T has an ordinal height α, and we can compute an evaluation map for X in α or so jumps of X.

But HYP contains *pseudo-wellorders*: there are orderings α^* which are ill-founded, but where there is no hyperarithmetic decreasing sequence. So

$$HYP \vDash \text{ there are Borel codes which are not completely determined}$$

because there are trees T^* which HYP "thinks" are Borel codes, but which are actually ill-founded and fail to have hyperarithmetic evaluation maps.

ADMSW introduced the "decorating trees" method for constructing defective Borel codes in HYP—that is, trees which:

- are hyperarithmetic,
- have pseudo-well-founded height (i.e. are ill-founded, but appear to be well-founded in HYP),
- are completely determined in HYP.

As we'll see, this method will let us construct a variety of very strange "Borel codes" in HYP.

Theorem

HYP $\models$
there is a completely determined Borel well-ordering of sets.

Theorem

HYP $\models$
there is a completely determined Borel well-ordering of sets.

For each hyperarithmetic real X, let $o(X) = (\beta, e)$ where β is least such that $X \leq_T \emptyset^\beta$ and e is least such that $X = (e)^{\emptyset^\beta}$. We will order the reals by the lexicographic ordering on these pairs.

Theorem

$HYP \models$
there is a completely determined Borel well-ordering of sets.

For each hyperarithmetic real X, let $o(X) = (\beta, e)$ where β is least such that $X \leq_T \emptyset^\beta$ and e is least such that $X = (e)^{\emptyset^\beta}$. We will order the reals by the lexicographic ordering on these pairs.

For any particular pair $(X, Y) \in HYP$, there is a least β such that $X, Y \leq \emptyset^\beta$. The set of such pairs is a set S_β, and there is a Borel code for this set of height roughly β.

Theorem

$HYP \models$
there is a completely determined Borel well-ordering of sets.

For each hyperarithmetic real X, let $o(X) = (\beta, e)$ where β is least such that $X \leq_T \emptyset^\beta$ and e is least such that $X = (e)^{\emptyset^\beta}$. We will order the reals by the lexicographic ordering on these pairs.

For any particular pair $(X, Y) \in HYP$, there is a least β such that $X, Y \leq \emptyset^\beta$. The set of such pairs is a set S_β, and there is a Borel code for this set of height roughly β.

Furthermore, within S_β, the set of pairs (X, Y) with $X < Y$ is a set R_β which is also encoded by a Borel code of height roughly β.

Theorem

HYP $\models$
there is a completely determined Borel well-ordering of sets.

For each hyperarithmetic real X, let $o(X) = (\beta, e)$ where β is least such that $X \leq_T \emptyset^\beta$ and e is least such that $X = (e)^{\emptyset^\beta}$. We will order the reals by the lexicographic ordering on these pairs.

For any particular pair $(X, Y) \in HYP$, there is a least β such that $X, Y \leq \emptyset^\beta$. The set of such pairs is a set S_β, and there is a Borel code for this set of height roughly β.

Furthermore, within S_β, the set of pairs (X, Y) with $X < Y$ is a set R_β which is also encoded by a Borel code of height roughly β.

The codes for S_β, R_β are given uniformly in β, so we can make sense of S_{β^*}, R_{β^*} for pseudo-well-ordinals as well.

For formal reasons, we assume all our Borel codes alternate union and intersection levels and are associated with a rank function ρ which assigns a pseudo-ordinal to each node.

We define the `Decorate` operation on trees as follows:

- `Decorate`(T) of a leaf leaves the leaf unchanged,

For formal reasons, we assume all our Borel codes alternate union and intersection levels and are associated with a rank function ρ which assigns a pseudo-ordinal to each node.

We define the `Decorate` operation on trees as follows:

- `Decorate`(T) of a leaf leaves the leaf unchanged,
- Suppose T is a tree with height γ^* encoding a union of the T_n. `Decorate`(T) will be a tree with height γ^* encoding a union over:
  - `Decorate`(T_n)

For formal reasons, we assume all our Borel codes alternate union and intersection levels and are associated with a rank function ρ which assigns a pseudo-ordinal to each node.

We define the `Decorate` operation on trees as follows:

- `Decorate`(T) of a leaf leaves the leaf unchanged,
- Suppose T is a tree with height γ^* encoding a union of the T_n. `Decorate`(T) will be a tree with height γ^* encoding a union over:
  - `Decorate`(T_n), and
  - `Decorate`($S_\beta \cap R_\beta$) where the code for $S_\beta \cap R_\beta$ has height $< \gamma^*$,

For formal reasons, we assume all our Borel codes alternate union and intersection levels and are associated with a rank function ρ which assigns a pseudo-ordinal to each node.

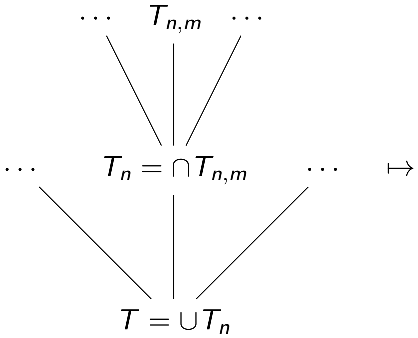
We define the `Decorate` operation on trees as follows:

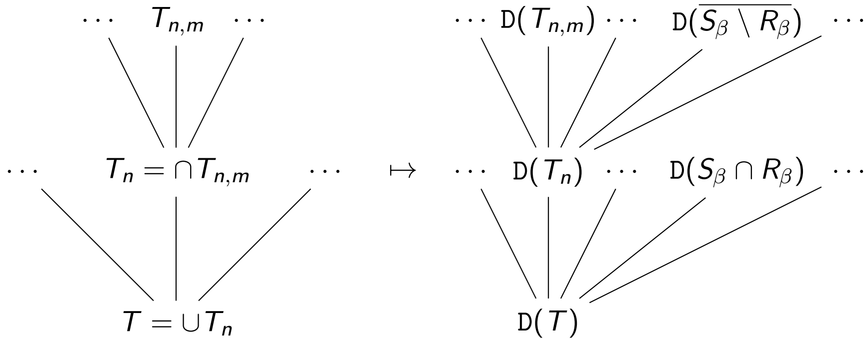
- `Decorate`(T) of a leaf leaves the leaf unchanged,
- Suppose T is a tree with height γ^* encoding a union of the T_n. `Decorate`(T) will be a tree with height γ^* encoding a union over:
  - `Decorate`(T_n), and
  - `Decorate`($S_\beta \cap R_\beta$) where the code for $S_\beta \cap R_\beta$ has height $< \gamma^*$,
- Suppose T is a tree with height γ^* encoding an intersection of the T_n. `Decorate`(T) will be a tree with height γ^* encoding an intersection over:
  - `Decorate`(T_n)

For formal reasons, we assume all our Borel codes alternate union and intersection levels and are associated with a rank function ρ which assigns a pseudo-ordinal to each node.

We define the `Decorate` operation on trees as follows:

- `Decorate`(T) of a leaf leaves the leaf unchanged,
- Suppose T is a tree with height γ^* encoding a union of the T_n. `Decorate`(T) will be a tree with height γ^* encoding a union over:
  - `Decorate`(T_n), and
  - `Decorate`($S_\beta \cap R_\beta$) where the code for $S_\beta \cap R_\beta$ has height $< \gamma^*$,
- Suppose T is a tree with height γ^* encoding an intersection of the T_n. `Decorate`(T) will be a tree with height γ^* encoding an intersection over:
  - `Decorate`(T_n), and
  - `Decorate`($\overline{S_\beta \setminus R_\beta}$) where the code for $\overline{S_\beta \setminus R_\beta}$ has height $< \gamma^*$,


$\cdots \quad T_{n,m} \quad \cdots$
$\cdots \quad T_n = \cap T_{n,m} \quad \cdots \quad \mapsto$
$T = \cup T_n$


$\cdots \quad T_{n,m} \quad \cdots$
$\cdots \quad T_n = \cap T_{n,m} \quad \cdots$
$T = \cup T_n$
$\mapsto$
$\cdots \; \mathtt{D}(T_{n,m}) \cdots \quad \mathtt{D}(\overline{S_\beta \setminus R_\beta}) \quad \cdots$
$\cdots \; \mathtt{D}(T_n) \; \cdots \quad \mathtt{D}(S_\beta \cap R_\beta) \quad \cdots$
$\mathtt{D}(T)$

Lemma

$\texttt{Decorate}(T)$ *is completely determined.*

Lemma

`Decorate`(T) *is completely determined.*

Consider any pair X, Y. There exactly one β with $(X, Y) \in [S_\beta]$. Using a bit more than β jumps, we can fix evaluation maps for the modified $S_\beta \cap R_\beta$ and $S_\beta \setminus R_\beta$ trees, as well as all low rank sub-trees of T.

Lemma

Decorate(T) *is completely determined.*

Consider any pair X, Y. There exactly one β with $(X, Y) \in [S_\beta]$. Using a bit more than β jumps, we can fix evaluation maps for the modified $S_\beta \cap R_\beta$ and $S_\beta \setminus R_\beta$ trees, as well as all low rank sub-trees of T.

Suppose $X < Y$. Consider the nodes of our tree of rank higher than β:

- at union nodes, we know that $(X, Y) \in [S_\beta \cap R_\beta]$, so we just mark the union as 1,

Lemma

`Decorate`(T) *is completely determined.*

Consider any pair X, Y. There exactly one β with $(X, Y) \in [S_\beta]$. Using a bit more than β jumps, we can fix evaluation maps for the modified $S_\beta \cap R_\beta$ and $S_\beta \setminus R_\beta$ trees, as well as all low rank sub-trees of T.

Suppose $X < Y$. Consider the nodes of our tree of rank higher than β:

- at union nodes, we know that $(X, Y) \in [S_\beta \cap R_\beta]$, so we just mark the union as 1,
- at intersection nodes, we know that *all* their union descendents of rank $\geq \beta$ have been marked 1, so we can decide how to mark the intersection with one additional jump.

Lemma

Decorate(T) *is completely determined.*

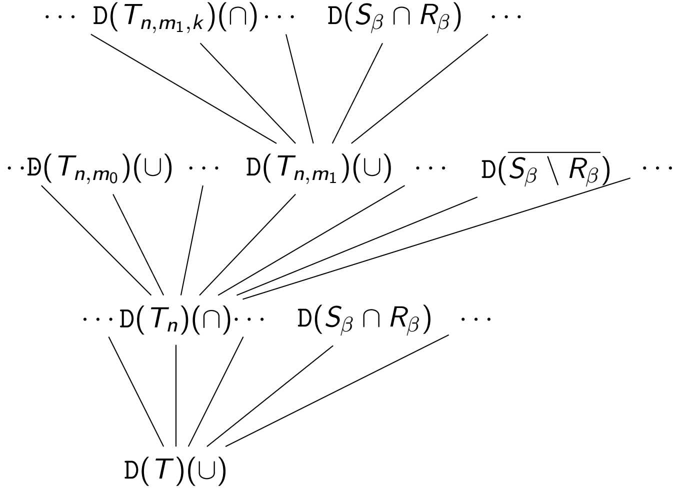
Consider any pair X, Y. There exactly one β with $(X, Y) \in [S_\beta]$. Using a bit more than β jumps, we can fix evaluation maps for the modified $S_\beta \cap R_\beta$ and $S_\beta \setminus R_\beta$ trees, as well as all low rank sub-trees of T.

Suppose $X < Y$. Consider the nodes of our tree of rank higher than β:

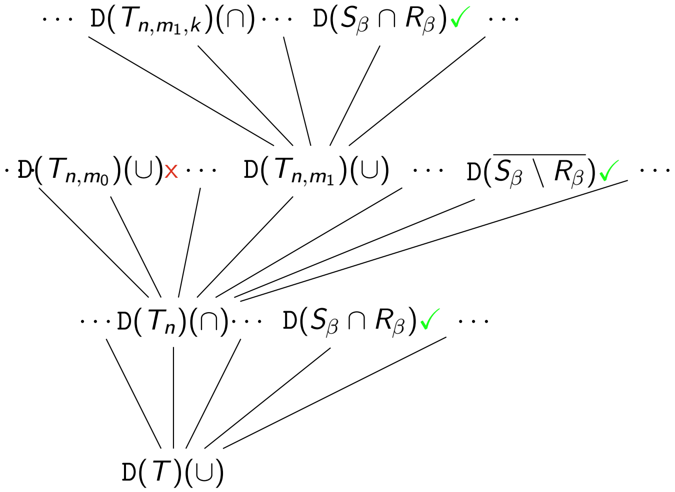
- at union nodes, we know that $(X, Y) \in [S_\beta \cap R_\beta]$, so we just mark the union as 1,
- at intersection nodes, we know that *all* their union descendents of rank $\geq \beta$ have been marked 1, so we can decide how to mark the intersection with one additional jump.

That is, we can mark all the high rank nodes uniformly in a bit more than β jumps.

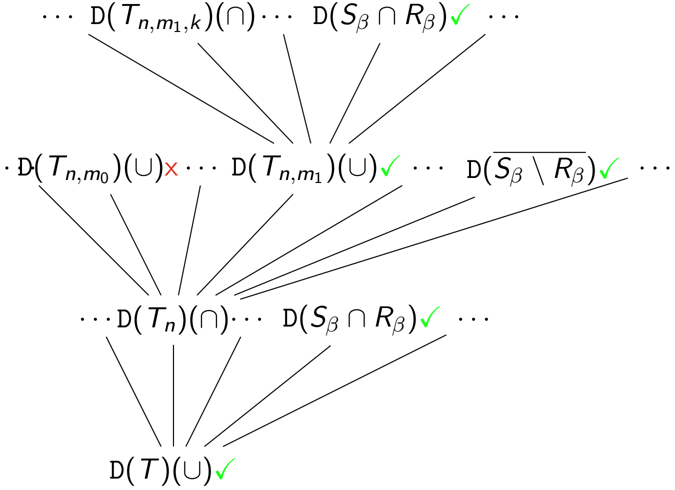
Suppose $X < Y$, $(X, Y) \in S_\beta$ and T, T_{n,m_1}, $T_{n,m_1,k}$ are much taller than β while T_{n,m_0} has height less than β. Then:



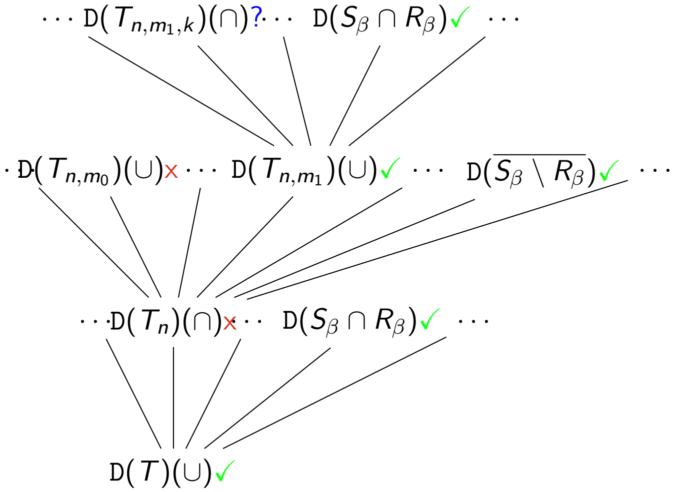
Suppose $X < Y$, $(X, Y) \in S_\beta$ and T, T_{n,m_1}, $T_{n,m_1,k}$ are much taller than β while T_{n,m_0} has height less than β. Then:



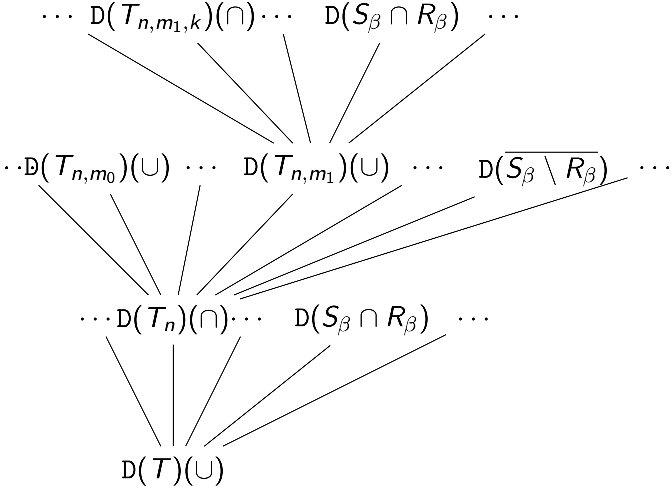
Suppose $X < Y$, $(X, Y) \in S_\beta$ and T, T_{n,m_1}, $T_{n,m_1,k}$ are much taller than β while T_{n,m_0} has height less than β. Then:



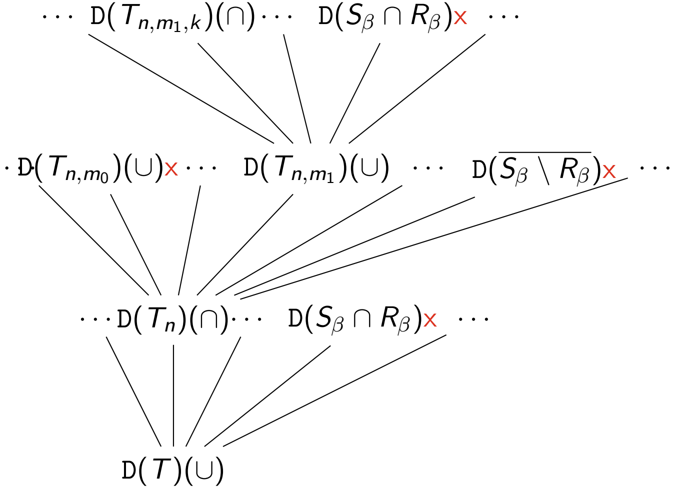
Suppose $X < Y$, $(X, Y) \in S_\beta$ and T, T_{n,m_1}, $T_{n,m_1,k}$ are much taller than β while T_{n,m_0} has height less than β. Then:



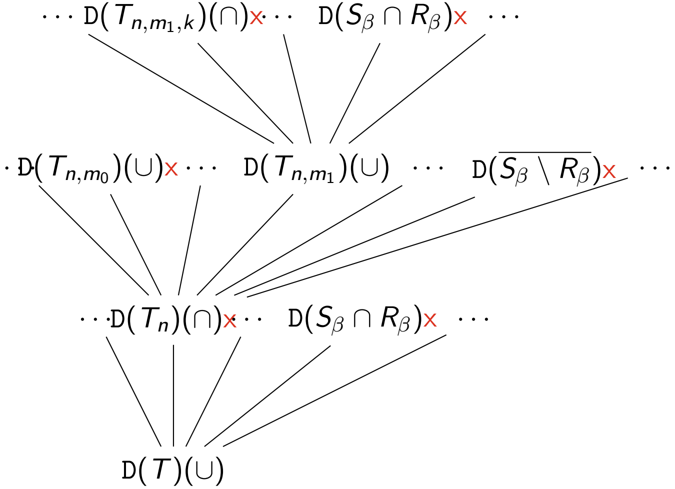
Suppose $Y \leq X$, $(X, Y) \in S_\beta$ and T, T_{n,m_1}, $T_{n,m_1,k}$ are much taller than β while T_{n,m_0} has height less than β. Then:



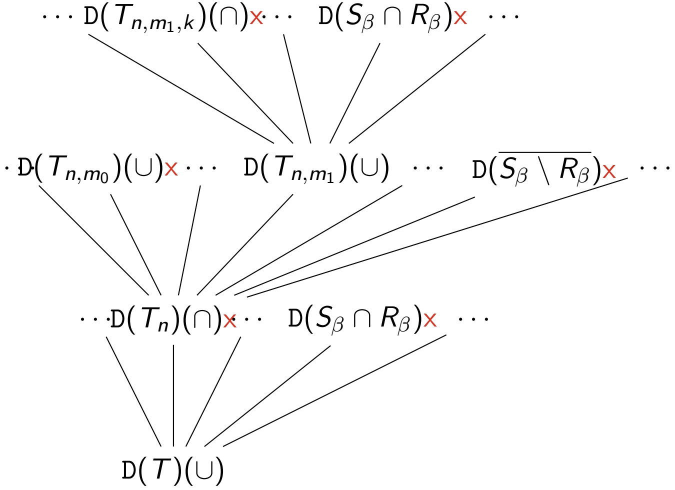
Suppose $Y \leq X$, $(X, Y) \in S_\beta$ and T, T_{n,m_1}, $T_{n,m_1,k}$ are much taller than β while T_{n,m_0} has height less than β. Then:



Suppose $Y \leq X$, $(X, Y) \in S_\beta$ and T, T_{n,m_1}, $T_{n,m_1,k}$ are much taller than β while T_{n,m_0} has height less than β. Then:



Suppose $Y \leq X$, $(X, Y) \in S_\beta$ and T, T_{n,m_1}, $T_{n,m_1,k}$ are much taller than β while T_{n,m_0} has height less than β. Then:



Putting this together with checking that this "Borel code" really does compute the well-ordering, we conclude that, in HYP:

- this is a Borel code T,
- it is completely determined, and
- for any (X, Y), there is a unique evaluation map $ev_T(X, Y)$ which is 1 if and only if $X < Y$.

The main idea is that we can, in a uniform and hyperarithmetic way, stratify the pairs of hyperarithmetic reals into the levels S_β.

We then define our tree so that whatever happens, it happens very uniformly: we can create an evaluation map for the low rank part of the tree in a hyperarithmetic way, and because we've made copies of it everywhere, we know we can define the evaluation map blindly on the rest of it.

One of the classic facts of about Borel graphs is that Borel chromatic number can, in general, be much larger than the general chromatic number.

Theorem

There are Borel graphs which have a 2-coloring, but no Borel coloring with even finitely many colors.

One of the classic facts of about Borel graphs is that Borel chromatic number can, in general, be much larger than the general chromatic number.

Theorem

There are Borel graphs which have a 2-coloring, but no Borel coloring with even finitely many colors.

This theorem is also true in HYP...but not for the right reason.

For instance, the graph whose vertices are increasing sequences of natural numbers and there is an edge between X and Y exactly when, for all n, $X(n) = Y(n+1)$ (or vice-versa).

But in HYP, this graph has a completely determined Borel 2-coloring!

For instance, the graph whose vertices are increasing sequences of natural numbers and there is an edge between X and Y exactly when, for all n, $X(n) = Y(n+1)$ (or vice-versa).

But in HYP, this graph has a completely determined Borel 2-coloring!

Note that when X, Y are sequences, if X and Y are in the same connected component then $X \leq \emptyset^{\beta}$ if and only if $Y \leq \emptyset^{\beta}$. So:

For instance, the graph whose vertices are increasing sequences of natural numbers and there is an edge between X and Y exactly when, for all n, $X(n) = Y(n+1)$ (or vice-versa).

But in HYP, this graph has a completely determined Borel 2-coloring!

Note that when X, Y are sequences, if X and Y are in the same connected component then $X \leq \emptyset^\beta$ if and only if $Y \leq \emptyset^\beta$. So:

- Our stratification is to take S_β to be those X so that β is least with $X \leq \emptyset^\beta$.

For instance, the graph whose vertices are increasing sequences of natural numbers and there is an edge between X and Y exactly when, for all n, $X(n) = Y(n+1)$ (or vice-versa).

But in HYP, this graph has a completely determined Borel 2-coloring!

Note that when X, Y are sequences, if X and Y are in the same connected component then $X \leq \emptyset^\beta$ if and only if $Y \leq \emptyset^\beta$. So:

- Our stratification is to take S_β to be those X so that β is least with $X \leq \emptyset^\beta$.
- If e is such that, for all $e' < e$, $(e')^\beta$ is in a different component from e, then we make 0 the color of $(e)^\beta$; we color the rest of the component based on distance from this point. This can be described by some Borel code R_β of height roughly β.

Theorem

HYP satisfies "if G is a completely determined, Borel, d-regular acyclic graph for some finite d then G has a completely determined Borel 2-coloring".

Theorem

HYP satisfies "if G is a completely determined, Borel, d-regular acyclic graph for some finite d then G has a completely determined Borel 2-coloring".

The idea is basically the same. Because we know each real has exactly d neighbors, each real has a hyperarithmetic connected component (complete with evaluation maps). We can color each X by looking for the first (in the sense of the well-ordering above) Y encoding the connected component of X, coloring the first column of Y 0, and coloring X based on its distance to that point.

For those X whose first encoding of the connected component is $\emptyset^{\beta}$-computable, we can do this with an actual Borel code R_{β}.

Theorem

HYP satisfies "there is a completely determined Borel acyclic graph which does not have a Borel 2-coloring".

Theorem

HYP satisfies "there is a completely determined Borel acyclic graph which does not have a Borel 2-coloring".

For each $\alpha^* \in \mathcal{O}^*$ and each e, we fix two distinct computable reals, $X_{\alpha^*,e,0}$ and $X_{\alpha,e,1}$. For each β, we will let S_β consist of those pairs (X, Y) such that β is least so that $(X, Y) \leq \emptyset^\beta$.

Theorem

HYP satisfies "there is a completely determined Borel acyclic graph which does not have a Borel 2-coloring".

For each $\alpha^* \in \mathcal{O}^*$ and each e, we fix two distinct computable reals, $X_{\alpha^*,e,0}$ and $X_{\alpha,e,1}$. For each β, we will let S_β consist of those pairs (X, Y) such that β is least so that $(X, Y) \leq \emptyset^\beta$.

If β is least such that there are $\emptyset^\beta$-computable evaluation maps for $(e)^{\emptyset^\alpha}$, we choose either one or two $\emptyset^\beta$-computable reals and make a path from $X_{\alpha,e,0}$ to $X_{\alpha,e,1}$ which contradicts the evaluation maps.

Theorem

The Borel Dual Ramsey Theorem for completely determined Borel colorings does not hold in HYP.

The idea of the proof is the same as the previous ones: we construct a coloring which successfully diagonalizes against all homogeneous solutions.

The general pattern is that, in HYP, "choice like constructions" work for completely determined Borel sets.

The general pattern is that, in HYP, "choice like constructions" work for completely determined Borel sets.

Question

Is there a principle (weaker than ATR_0*) which suffices to rule out this defective behavior?*

For instance "there is no completely determined Borel well-ordering of sets" is something we would expect, and would at least complicate some of the other constructions here.