Steel forcing in reverse mathematics

Itay Neeman

Department of Mathematics
University of California Los Angeles
Los Angeles, CA 90095

www.math.ucla.edu/~ineeman

Use PageDown or the down arrow to scroll through slides.
Press Esc when done.

Outline

Steel forcing provides a powerful method for constructing models of certain axioms in reverse mathematics. Used initially to separate the strengths of axioms. More recently used to discover the strength, relative to standard axioms, of INDEC, the first non-logical statement shown to be a theorem of hyperarithmetic analysis.

Outline

Steel forcing provides a powerful method for constructing models of certain axioms in reverse mathematics. Used initially to separate the strengths of axioms. More recently used to discover the strength, relative to standard axioms, of INDEC, the first non-logical statement shown to be a theorem of hyperarithmetic analysis.

Reverse Mathematics

Outline

Steel forcing provides a powerful method for constructing models of certain axioms in reverse mathematics. Used initially to separate the strengths of axioms. More recently used to discover the strength, relative to standard axioms, of INDEC, the first non-logical statement shown to be a theorem of hyperarithmetic analysis.

Reverse Mathematics

Steel forcing

Outline

Steel forcing provides a powerful method for constructing models of certain axioms in reverse mathematics. Used initially to separate the strengths of axioms. More recently used to discover the strength, relative to standard axioms, of INDEC, the first non-logical statement shown to be a theorem of hyperarithmetic analysis.

Outline

Steel forcing provides a powerful method for constructing models of certain axioms in reverse mathematics. Used initially to separate the strengths of axioms. More recently used to discover the strength, relative to standard axioms, of INDEC, the first non-logical statement shown to be a theorem of hyperarithmetic analysis.

Outline

Steel forcing provides a powerful method for constructing models of certain axioms in reverse mathematics. Used initially to separate the strengths of axioms. More recently used to discover the strength, relative to standard axioms, of INDEC, the first non-logical statement shown to be a theorem of hyperarithmetic analysis.

Outline

Steel forcing provides a powerful method for constructing models of certain axioms in reverse mathematics. Used initially to separate the strengths of axioms. More recently used to discover the strength, relative to standard axioms, of INDEC, the first non-logical statement shown to be a theorem of hyperarithmetic analysis.

Reverse mathematics

Reverse mathematics *is a program in mathematical logic that seeks to determine which axioms are required to prove theorems of mathematics. The method can briefly be described as "going backwards from the theorems to the axioms". This contrasts with the ordinary mathematical practice of deriving theorems from axioms.*

Reverse mathematics *is a program in mathematical logic that seeks to determine which axioms are required to prove theorems of mathematics. The method can briefly be described as "going backwards from the theorems to the axioms". This contrasts with the ordinary mathematical practice of deriving theorems from axioms.*

Wikipedia

Reverse mathematics *is a program in mathematical logic that seeks to determine which axioms are required to prove theorems of mathematics. The method can briefly be described as "going backwards from the theorems to the axioms". This contrasts with the ordinary mathematical practice of deriving theorems from axioms.*

Wikipedia

Primary reference, Simpson [1999].

Reverse mathematics

Reverse mathematics

Work over a weak base system. Various standard axioms provide strengthening. Given a theorem Φ, find, ideally, a standard axiom A so that, over the base system:

1. A is enough to prove Φ.
2. A is necessary to prove Φ.

Reverse mathematics

Work over a weak base system. Various standard axioms provide strengthening. Given a theorem Φ, find, ideally, a standard axiom A so that, over the base system:

1. A is enough to prove Φ.
2. A is necessary to prove Φ.

A retreat from 2:

2′. The standard axioms weaker than A are not enough to prove Φ.

Reverse mathematics

Work over a weak base system. Various standard axioms provide strengthening. Given a theorem Φ, find, ideally, a standard axiom A so that, over the base system:

1. A is enough to prove Φ.
2. A is necessary to prove Φ.

A retreat from 2:

2′. The standard axioms weaker than A are not enough to prove Φ.

Equivalently:

2′. There is a model of the base system satisfying the standard axioms weaker than A, in which Φ fails.

Reverse mathematics

Work over a weak base system. Various standard axioms provide strengthening. Given a theorem Φ, find, ideally, a standard axiom A so that, over the base system:

1. A is enough to prove Φ.
2. A is necessary to prove Φ.

A retreat from 2:

2′. The standard axioms weaker than A are not enough to prove Φ.

Equivalently:

2′. There is a model of the base system satisfying the standard axioms weaker than A, in which Φ fails.

Conceptually similar to consistency proofs in set theory. But concerned mainly with theorems of analysis (second order number theory).

Reverse mathematics, continued

Some theorems addressed by reverse mathematics:

- Heine-Borel theorem on $[0, 1]$.
- Sequential completeness of $\mathbb{R}$.
- Bolzano–Weierstrass theorem.
- The perfect set theorem.
- Open determinacy.
- Cantor-Bendixson theorem.
- $\ldots$.

Reverse mathematics, continued

Some theorems addressed by reverse mathematics:

- Heine-Borel theorem on $[0, 1]$.
- Sequential completeness of $\mathbb{R}$.
- Bolzano–Weierstrass theorem.
- The perfect set theorem.
- Open determinacy.
- Cantor-Bendixson theorem.
- $\dots$.

Natural base system: axioms of PA for the natural numbers (or PA^{-} and limited induction), and Δ^0_1 comprehension.

Reverse mathematics, continued

Some theorems addressed by reverse mathematics:

- Heine-Borel theorem on $[0, 1]$.
- Sequential completeness of $\mathbb{R}$.
- Bolzano–Weierstrass theorem.
- The perfect set theorem.
- Open determinacy.
- Cantor-Bendixson theorem.
- $\ldots$.

Natural base system: axioms of PA for the natural numbers (or PA^- and limited induction), and Δ^0_1 comprehension.

Additional axioms provide *sets* of natural numbers, beyond the recursive sets one gets from the base system.

Reverse mathematics, continued

Some theorems addressed by reverse mathematics:

- Heine-Borel theorem on $[0, 1]$.
- Sequential completeness of $\mathbb{R}$.
- Bolzano–Weierstrass theorem.
- The perfect set theorem.
- Open determinacy.
- Cantor-Bendixson theorem.
- $\ldots$.

Natural base system: axioms of PA for the natural numbers (or PA^- and limited induction), and Δ^0_1 comprehension.

Additional axioms provide *sets* of natural numbers, beyond the recursive sets one gets from the base system.

Reverse mathematics measures how much of this extra strength is needed for each theorem.

Subsystems of analysis

Standard markers of strength include the following set existence axioms,

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

1. Δ^0_1 comprehension: for Σ^0_1 formulas φ, ψ, if $\varphi(n) \leftrightarrow \neg\psi(n)$, then $\{n \mid \varphi(n)\}$ exists.

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

1. Δ^0_1 comprehension: for Σ^0_1 formulas φ, ψ, if $\varphi(n) \leftrightarrow \neg\psi(n)$, then $\{n \mid \varphi(n)\}$ exists.
2. Weak König lemma: each infinite subtree of the binary tree has a branch.

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

1. Δ^0_1 comprehension: for Σ^0_1 formulas φ, ψ, if $\varphi(n) \leftrightarrow \neg\psi(n)$, then $\{n \mid \varphi(n)\}$ exists.
2. Weak König lemma: each infinite subtree of the binary tree has a branch.
3. Arithmetic comprehension.

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

1. Δ^0_1 comprehension: for Σ^0_1 formulas φ, ψ, if $\varphi(n) \leftrightarrow \neg\psi(n)$, then $\{n \mid \varphi(n)\}$ exists.
2. Weak König lemma: each infinite subtree of the binary tree has a branch.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

1. Δ^0_1 comprehension: for Σ^0_1 formulas φ, ψ, if $\varphi(n) \leftrightarrow \neg\psi(n)$, then $\{n \mid \varphi(n)\}$ exists.
2. Weak König lemma: each infinite subtree of the binary tree has a branch.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice: if $(\forall n)(\exists! x)\varphi(n, x)$, then there is $\langle y_n \mid n < \omega\rangle$ so that $(\forall n)\varphi(n, y_n)$. Arithmetic φ

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

1. Δ^0_1 comprehension: for Σ^0_1 formulas φ, ψ, if $\varphi(n) \leftrightarrow \neg\psi(n)$, then $\{n \mid \varphi(n)\}$ exists.
2. Weak König lemma: each infinite subtree of the binary tree has a branch.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice: if $(\forall n)(\exists! x)\varphi(n, x)$, then there is $\langle y_n \mid n < \omega\rangle$ so that $(\forall n)\varphi(n, y_n)$. Arithmetic φ
6. Δ^1_1 comprehension.

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

1. Δ^0_1 comprehension: for Σ^0_1 formulas φ, ψ, if $\varphi(n) \leftrightarrow \neg\psi(n)$, then $\{n \mid \varphi(n)\}$ exists.
2. Weak König lemma: each infinite subtree of the binary tree has a branch.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice: if $(\forall n)(\exists! x)\varphi(n, x)$, then there is $\langle y_n \mid n < \omega\rangle$ so that $(\forall n)\varphi(n, y_n)$. Arithmetic φ
6. Δ^1_1 comprehension.
7. Σ^1_1 choice: as 5, but without assuming uniqueness.

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

1. Δ^0_1 comprehension: for Σ^0_1 formulas φ, ψ, if $\varphi(n) \leftrightarrow \neg\psi(n)$, then $\{n \mid \varphi(n)\}$ exists.
2. Weak König lemma: each infinite subtree of the binary tree has a branch.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice: if $(\forall n)(\exists! x)\varphi(n, x)$, then there is $\langle y_n \mid n < \omega\rangle$ so that $(\forall n)\varphi(n, y_n)$. Arithmetic φ
6. Δ^1_1 comprehension.
7. Σ^1_1 choice: as 5, but without assuming uniqueness.
8. Arithmetic transfinite recursion: arithmetic comprehension can be iterated transfinitely.

Subsystems of analysis

Standard markers of strength include the following set existence axioms, increasing over the base system RCA_0, consisting of PA^-, Σ^0_1 induction, and Δ^0_1 comprehension.

1. Δ^0_1 comprehension: for Σ^0_1 formulas φ, ψ, if $\varphi(n) \leftrightarrow \neg\psi(n)$, then $\{n \mid \varphi(n)\}$ exists.
2. Weak König lemma: each infinite subtree of the binary tree has a branch.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice: if $(\forall n)(\exists! x)\varphi(n, x)$, then there is $\langle y_n \mid n < \omega\rangle$ so that $(\forall n)\varphi(n, y_n)$. Arithmetic φ
6. Δ^1_1 comprehension.
7. Σ^1_1 choice: as 5, but without assuming uniqueness.
8. Arithmetic transfinite recursion: arithmetic comprehension can be iterated transfinitely.
9. Π^1_1 comprehension.

Subsystems of analysis, continued

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Subsystems of analysis, continued

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Added to RCA_0, forming subsystems of analysis.

Subsystems of analysis, continued

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Added to RCA_0, forming subsystems of analysis.

1, 2, 3, 8, 9 give big five systems of reverse mathematics.

Subsystems of analysis, continued

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Added to RCA_0, forming subsystems of analysis.

1, 2, 3, 8, 9 give big five systems of reverse mathematics.

4, 5, 6, 7 give systems of hyperarithmetic analysis:

Subsystems of analysis, continued

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Added to RCA_0, forming subsystems of analysis.

1, 2, 3, 8, 9 give big five systems of reverse mathematics.

4, 5, 6, 7 give systems of hyperarithmetic analysis:

T is a theory of hyperarithmetic analysis if

Subsystems of analysis, continued

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Added to RCA_0, forming subsystems of analysis.

1, 2, 3, 8, 9 give big five systems of reverse mathematics.

4, 5, 6, 7 give systems of hyperarithmetic analysis:

T is a theory of hyperarithmetic analysis if (a) its ω models are closed under joins and hyperarithmetic reducibility;

Subsystems of analysis, continued

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Added to RCA_0, forming subsystems of analysis.

1, 2, 3, 8, 9 give big five systems of reverse mathematics.

4, 5, 6, 7 give systems of hyperarithmetic analysis:

T is a theory of hyperarithmetic analysis if (a) its ω models are closed under joins and hyperarithmetic reducibility; (b) it holds in $\mathrm{HYP}(x)$ for all x.

Steel forcing

Steel forcing

Starting model $M = L_{\omega_1^{ck}} = \mathrm{HYP}$.

Steel forcing

Starting model $M = L_{\omega_1^{ck}} = \mathrm{HYP}$.

Forcing adds a tree T on ω, with a generic countable set of branches B.

Steel forcing

Starting model $M = L_{\omega_1^{ck}} = \mathrm{HYP}$.

Forcing adds a tree T on ω, with a generic countable set of branches B.

Key property: $L_\alpha[T]$ ($\alpha < \omega_1^{ck}$) cannot distinguish between ranks $> \alpha$ in the tree T.

Steel forcing

Starting model $M = L_{\omega_1^{ck}} = \mathrm{HYP}$.

Forcing adds a tree T on ω, with a generic countable set of branches B.

Key property: $L_\alpha[T]$ ($\alpha < \omega_1^{ck}$) cannot distinguish between ranks $> \alpha$ in the tree T.

More generally, for finite $F \subseteq B$,

Steel forcing

Starting model $M = L_{\omega_1^{ck}} = \mathrm{HYP}$.

Forcing adds a tree T on ω, with a generic countable set of branches B.

Key property: $L_\alpha[T]$ ($\alpha < \omega_1^{ck}$) cannot distinguish between ranks $> \alpha$ in the tree T.

More generally, for finite $F \subseteq B$, $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in T

Steel forcing

Starting model $M = L_{\omega_1^{ck}} = \mathrm{HYP}$.

Forcing adds a tree T on ω, with a generic countable set of branches B.

Key property: $L_\alpha[T]$ ($\alpha < \omega_1^{ck}$) cannot distinguish between ranks $> \alpha$ in the tree T.

More generally, for finite $F \subseteq B$, $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.

Steel forcing

Starting model $M = L_{\omega_1^{ck}} = \mathrm{HYP}$.

Forcing adds a tree T on ω, with a generic countable set of branches B.

Key property: $L_\alpha[T]$ ($\alpha < \omega_1^{ck}$) cannot distinguish between ranks $> \alpha$ in the tree T.

More generally, for finite $F \subseteq B$, $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.

For finite $F \subseteq B$, set $M_F = M[T, F] = \mathrm{HYP}[T, F]$.

Steel forcing

Starting model $M = L_{\omega_1^{ck}} = \mathrm{HYP}$.

Forcing adds a tree T on ω, with a generic countable set of branches B.

Key property: $L_\alpha[T]$ ($\alpha < \omega_1^{ck}$) cannot distinguish between ranks $> \alpha$ in the tree T.

More generally, for finite $F \subseteq B$, $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.

For finite $F \subseteq B$, set $M_F = M[T, F] = \mathrm{HYP}[T, F]$.

For $K \subseteq B$, set $M_K = \bigcup_{F \subseteq K,\text{ finite}} M[T, F]$.

Steel forcing

Starting model $M = L_{\omega_1^{ck}} = \mathrm{HYP}$.

Forcing adds a tree T on ω, with a generic countable set of branches B.

Key property: $L_\alpha[T]$ ($\alpha < \omega_1^{ck}$) cannot distinguish between ranks $> \alpha$ in the tree T.

More generally, for finite $F \subseteq B$, $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.

For finite $F \subseteq B$, set $M_F = M[T, F] = \mathrm{HYP}[T, F]$.

For $K \subseteq B$, set $M_K = \bigcup_{F \subseteq K,\ \text{finite}} M[T, F]$.

With choice of K, powerful way to produce models of hyperarithmetic analysis.

Steel forcing, example 1

- $L_\alpha[T]$ cannot distinguish between ranks $> \alpha$ in the tree T.
- $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.
- $M_K = \bigcup_{F \subseteq K,\text{ finite}} M[T, F]$.

Steel forcing, example 1

- $L_\alpha[T]$ cannot distinguish between ranks $> \alpha$ in the tree T.
- $L_\alpha[T,F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.
- $M_K = \bigcup_{F \subseteq K,\text{ finite}} M[T,F]$.

For example, take $K = B$.

Steel forcing, example 1

- $L_\alpha[T]$ cannot distinguish between ranks $> \alpha$ in the tree T.
- $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.
- $M_K = \bigcup_{F \subseteq K,\text{ finite}} M[T, F]$.

For example, take $K = B$.

The only branches through T in $M[T, F]$ are those in F.

Steel forcing, example 1

- $L_\alpha[T]$ cannot distinguish between ranks $> \alpha$ in the tree T.
- $L_\alpha[T,F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.
- $M_K = \bigcup_{F \subseteq K,\text{ finite}} M[T,F]$.

For example, take $K = B$.

The only branches through T in $M[T,F]$ are those in F.

So, in M_K there is no infinite sequence of branches through T.

Steel forcing, example 1

- $L_\alpha[T]$ cannot distinguish between ranks $> \alpha$ in the tree T.
- $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.
- $M_K = \bigcup_{F \subseteq K,\text{ finite}} M[T, F]$.

For example, take $K = B$.

The only branches through T in $M[T, F]$ are those in F.

So, in M_K there is no infinite sequence of branches through T. In particular M_K does not satisfy Σ^1_1 choice.

Steel forcing, example 1

- $L_\alpha[T]$ cannot distinguish between ranks $> \alpha$ in the tree T.
- $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t \text{ extends to } b \in F\}$.
- $M_K = \bigcup_{F \subseteq K,\text{ finite}} M[T, F]$.

For example, take $K = B$.

The only branches through T in $M[T, F]$ are those in F.

So, in M_K there is no infinite sequence of branches through T. In particular M_K does not satisfy Σ^1_1 choice.

But M_K satisfies Δ^1_1 comprehension.

Steel forcing, example 1

- $L_\alpha[T]$ cannot distinguish between ranks $> \alpha$ in the tree T.
- $L_\alpha[T, F]$ cannot distinguish between ranks $> \alpha$ in $T - \{t \mid t$ extends to $b \in F\}$.
- $M_K = \bigcup_{F \subseteq K, \text{ finite}} M[T, F]$.

For example, take $K = B$.

The only branches through T in $M[T, F]$ are those in F.

So, in M_K there is no infinite sequence of branches through T. In particular M_K does not satisfy Σ^1_1 choice.

But M_K satisfies Δ^1_1 comprehension.

Theorem (Steel [1977, 1978])

Δ^1_1 comprehension does not imply Σ^1_1 choice.

Steel forcing, example 2

Steel forcing, example 2

Pick $K \subseteq B$ so that the set $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

Steel forcing, example 2

Pick $K \subseteq B$ so that the set $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

D is Δ^1_1 in M_K,

Steel forcing, example 2

Pick $K \subseteq B$ so that the set $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

D is Δ^1_1 in M_K, but cannot belong to M_K since it constructs infinitely many branches through T.

Steel forcing, example 2

Pick $K \subseteq B$ so that the set $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

D is Δ^1_1 in M_K, but cannot belong to M_K since it constructs infinitely many branches through T.

Δ^1_1 comprehension fails in M_K.

Steel forcing, example 2

Pick $K \subseteq B$ so that the set $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

D is Δ^1_1 in M_K, but cannot belong to M_K since it constructs infinitely many branches through T.

Δ^1_1 comprehension fails in M_K.

On the other hand....

Steel forcing, example 2

Pick $K \subseteq B$ so that the set $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

D is Δ^1_1 in M_K, but cannot belong to M_K since it constructs infinitely many branches through T.

Δ^1_1 comprehension fails in M_K.

On the other hand.... let φ be a Σ^1_1 statement with parameters in $M[T, F]$, and a *unique* witness in M_K.

Steel forcing, example 2

Pick $K \subseteq B$ so that the set $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

D is Δ^1_1 in M_K, but cannot belong to M_K since it constructs infinitely many branches through T.

Δ^1_1 comprehension fails in M_K.

On the other hand.... let φ be a Σ^1_1 statement with parameters in $M[T, F]$, and a *unique* witness in M_K. This witness must belong to $M[T, F]$.

Steel forcing, example 2

Pick $K \subseteq B$ so that the set $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

D is Δ^1_1 in M_K, but cannot belong to M_K since it constructs infinitely many branches through T.

Δ^1_1 comprehension fails in M_K.

On the other hand.... let φ be a Σ^1_1 statement with parameters in $M[T, F]$, and a *unique* witness in M_K. This witness must belong to $M[T, F]$.

Weak Σ^1_1 choice holds in M_K.

Steel forcing, example 2

Pick $K \subseteq B$ so that the set $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

D is Δ^1_1 in M_K, but cannot belong to M_K since it constructs infinitely many branches through T.

Δ^1_1 comprehension fails in M_K.

On the other hand.... let φ be a Σ^1_1 statement with parameters in $M[T, F]$, and a *unique* witness in M_K. This witness must belong to $M[T, F]$.

Weak Σ^1_1 choice holds in M_K.

Theorem (Van Wesep [1977])

Weak Σ^1_1 choice does not imply Δ^1_1 comprehension.

Steel forcing, recent uses

Steel forcing, recent uses

More recently, Montalbán [2006] introduced new *game* comprehension axioms.

Steel forcing, recent uses

More recently, Montalbán [2006] introduced new *game* comprehension axioms.

State that for certain games, the set of winning positions (for I, for II) exists.

Steel forcing, recent uses

More recently, Montalbán [2006] introduced new *game* comprehension axioms.

State that for certain games, the set of winning positions (for I, for II) exists.

Montalbán used Steel forcing to separate these axioms in strength.

Steel forcing, recent uses

More recently, Montalbán [2006] introduced new *game* comprehension axioms.

State that for certain games, the set of winning positions (for I, for II) exists.

Montalbán used Steel forcing to separate these axioms in strength.

Most importantly Montalbán discovered the first “natural” theorem of hyperarithmetic analysis.

Steel forcing, recent uses

More recently, Montalbán [2006] introduced new *game* comprehension axioms.

State that for certain games, the set of winning positions (for I, for II) exists.

Montalbán used Steel forcing to separate these axioms in strength.

Most importantly Montalbán discovered the first “natural” theorem of hyperarithmetic analysis. The next topic describes the theorem.

Basic Definitions

Work throughout with countable linear orders.

Definition

- A linear order $(U; <_U)$ is scattered if it does not embed $\mathbb{Q}$.
- A gap in U is a partition of U into sets L and R, closed leftward and rightward respectively.
- A gap $\langle L, R \rangle$ is a decomposition of U if U does not embed into L, and does not embed into R.
- U is **in**decomposable if, for every gap $\langle L, R \rangle$, U embeds into either L or R.

Basic Definitions

Work throughout with countable linear orders.

Definition

- A linear order $(U; <_U)$ is scattered if it does not embed $\mathbb{Q}$.
- A gap in U is a partition of U into sets L and R, closed leftward and rightward respectively.
- A gap $\langle L, R\rangle$ is a decomposition of U if U does not embed into L, and does not embed into R.
- U is **in**decomposable if, for every gap $\langle L, R\rangle$, U embeds into either L or R.

Note

If U is scattered, it cannot embed into both L and R.

Jullien's Indecomposability Theorem

Recall, U is indecomposable if, for every gap $\langle L, R\rangle$, U embeds into either L or R.

Jullien's Indecomposability Theorem

Recall, U is indecomposable if, for every gap $\langle L, R\rangle$, U embeds into either L or R.

Definition

U is indecomposable to the left if whenever $\langle L, R\rangle$ is a gap with $L \neq \emptyset$, U embeds into L.

Jullien's Indecomposability Theorem

Recall, U is indecomposable if, for every gap $\langle L, R\rangle$, U embeds into either L or R.

Definition

U is indecomposable to the left if whenever $\langle L, R\rangle$ is a gap with $L \neq \emptyset$, U embeds into L. Indecomposability to the right defined similarly.

Jullien's Indecomposability Theorem

Recall, U is indecomposable if, for every gap $\langle L, R\rangle$, U embeds into either L or R.

Definition

U is indecomposable to the left if whenever $\langle L, R\rangle$ is a gap with $L \neq \emptyset$, U embeds into L. Indecomposability to the right defined similarly.

Theorem (Jullien 1969)

Suppose U is scattered and indecomposable. Then U is indecomposable to the left, or indecomposable to the right.

Jullien's Indecomposability Theorem

Recall, U is indecomposable if, for every gap $\langle L, R \rangle$, U embeds into either L or R.

Definition

U is indecomposable to the left if whenever $\langle L, R \rangle$ is a gap with $L \neq \emptyset$, U embeds into L. Indecomposability to the right defined similarly.

Theorem (Jullien 1969)

Suppose U is scattered and indecomposable. Then U is indecomposable to the left, or indecomposable to the right.

Used classically for classifying linear orders. More recently by Montalbán working on strength of Fraïssé's conjecture.

Proof of INDEC

Suppose U is scattered, indecomposable.

Proof of INDEC

Suppose U is scattered, indecomposable.

For each $a \in U$, set

- $L_a = \{x \mid x <_U a\}$,
- $R_a = \{x \mid x >_U a\}$.

Proof of INDEC

Suppose U is scattered, indecomposable.

For each $a \in U$, set

- $L_a = \{x \mid x <_U a\}$,
- $R_a = \{x \mid x >_U a\}$.

By indecomposability, U embeds into either L_a or R_a. It does not embed into both.

Proof of INDEC

Suppose U is scattered, indecomposable.

For each $a \in U$, set

- $L_a = \{x \mid x <_U a\}$,
- $R_a = \{x \mid x >_U a\}$.

By indecomposability, U embeds into either L_a or R_a. It does not embed into both.

Set

- $R^* = \{a \mid U \text{ embeds into } L_a\}$,
- $L^* = \{a \mid U \text{ embeds into } R_a\}$.

Proof of INDEC

Suppose U is scattered, indecomposable.

For each $a \in U$, set

- $L_a = \{x \mid x <_U a\}$,
- $R_a = \{x \mid x >_U a\}$.

By indecomposability, U embeds into either L_a or R_a. It does not embed into both.

Set

- $R^* = \{a \mid U \text{ embeds into } L_a\}$,
- $L^* = \{a \mid U \text{ embeds into } R_a\}$.

If R^* is empty, then U is indecomposable to the right, and we are done. Similarly if L^* is empty then U is indecomposable to the left.

Proof of INDEC

Suppose U is scattered, indecomposable.

For each $a \in U$, set

- $L_a = \{x \mid x <_U a\}$,
- $R_a = \{x \mid x >_U a\}$.

By indecomposability, U embeds into either L_a or R_a. It does not embed into both.

Set

- $R^* = \{a \mid U \text{ embeds into } L_a\}$,
- $L^* = \{a \mid U \text{ embeds into } R_a\}$.

If R^* is empty, then U is indecomposable to the right, and we are done. Similarly if L^* is empty then U is indecomposable to the left. So suppose neither is empty.

Proof of INDEC, continued

For each a, U embeds into either L_a or R_a, not both.

$$R^* = \{a \mid U \text{ embeds into } L_a\},$$

$$L^* = \{a \mid U \text{ embeds into } R_a\}.$$

For contradiction, neither R^* nor L^* is empty.

Proof of INDEC, continued

For each a, U embeds into either L_a or R_a, not both.

$R^* = \{a \mid U \text{ embeds into } L_a\}$,

$L^* = \{a \mid U \text{ embeds into } R_a\}$.

For contradiction, neither R^* nor L^* is empty.

L^*, R^* partition U. R^* is closed rightward, L^* is closed leftward.

Proof of INDEC, continued

For each a, U embeds into either L_a or R_a, not both.

$$R^* = \{a \mid U \text{ embeds into } L_a\},$$

$$L^* = \{a \mid U \text{ embeds into } R_a\}.$$

For contradiction, neither R^* nor L^* is empty.

L^*, R^* partition U. R^* is closed rightward, L^* is closed leftward.

By indecomposability, U embeds into L^* or R^*. Suppose for definitiveness it is L^*. Let $\sigma : U \to L^*$.

Proof of INDEC, continued

For each a, U embeds into either L_a or R_a, not both.

$R^* = \{a \mid U \text{ embeds into } L_a\}$,

$L^* = \{a \mid U \text{ embeds into } R_a\}$.

For contradiction, neither R^* nor L^* is empty.

L^*, R^* partition U. R^* is closed rightward, L^* is closed leftward.

By indecomposability, U embeds into L^* or R^*. Suppose for definitiveness it is L^*. Let $\sigma\colon U \to L^*$.

Fix $a \in R^*$ (possible since $R^* \neq \emptyset$). Let $b = \sigma(a) \in L^*$.

Proof of INDEC, continued

For each a, U embeds into either L_a or R_a, not both.

$$R^* = \{a \mid U \text{ embeds into } L_a\},$$
$$L^* = \{a \mid U \text{ embeds into } R_a\}.$$

For contradiction, neither R^* nor L^* is empty.

L^*, R^* partition U. R^* is closed rightward, L^* is closed leftward.

By indecomposability, U embeds into L^* or R^*. Suppose for definitiveness it is L^*. Let $\sigma : U \to L^*$.

Fix $a \in R^*$ (possible since $R^* \neq \emptyset$). Let $b = \sigma(a) \in L^*$.

Then $\mathrm{range}(\sigma^2)$ is to the left of b. So U embeds into L_b. Since $b \in L^*$, U also embeds into R_b. Contradiction.

INDEC and hyperarithmetic analysis

INDEC was proved by Jullien [1969]. Part of an investigation of the structure of linear orders.

INDEC and hyperarithmetic analysis

INDEC was proved by Jullien [1969]. Part of an investigation of the structure of linear orders.

Montalbán [2006] was led to it working on strength of Fraïssé's conjecture

INDEC and hyperarithmetic analysis

INDEC was proved by Jullien [1969]. Part of an investigation of the structure of linear orders.

Montalbán [2006] was led to it working on strength of Fraïssé's conjecture $=$ Laver [1971] theorem.

INDEC and hyperarithmetic analysis

INDEC was proved by Jullien [1969]. Part of an investigation of the structure of linear orders.

Montalbán [2006] was led to it working on strength of Fraïssé's conjecture $=$ Laver [1971] theorem.

Investigating INDEC, Montalbán:

INDEC and hyperarithmetic analysis

INDEC was proved by Jullien [1969]. Part of an investigation of the structure of linear orders.

Montalbán [2006] was led to it working on strength of Fraïssé's conjecture $=$ Laver [1971] theorem.

Investigating INDEC, Montalbán:

1. Observed that INDEC is provable in RCA_0 + Δ^1_1 comprehension.

INDEC and hyperarithmetic analysis

INDEC was proved by Jullien [1969]. Part of an investigation of the structure of linear orders.

Montalbán [2006] was led to it working on strength of Fraïssé's conjecture $=$ Laver [1971] theorem.

Investigating INDEC, Montalbán:

1. Observed that INDEC is provable in RCA_0 + Δ^1_1 comprehension. It follows that INDEC holds in $\mathrm{HYP}(X)$ for all X.

INDEC and hyperarithmetic analysis

INDEC was proved by Jullien [1969]. Part of an investigation of the structure of linear orders.

Montalbán [2006] was led to it working on strength of Fraïssé's conjecture $=$ Laver [1971] theorem.

Investigating INDEC, Montalbán:

1. Observed that INDEC is provable in RCA_0 + Δ^1_1 comprehension. It follows that INDEC holds in $\mathrm{HYP}(X)$ for all X.
2. Proved that every ω model of INDEC is closed under the αth Turing jump, for each ordinal α in the model.

INDEC and hyperarithmetic analysis

INDEC was proved by Jullien [1969]. Part of an investigation of the structure of linear orders.

Montalbán [2006] was led to it working on strength of Fraïssé's conjecture $=$ Laver [1971] theorem.

Investigating INDEC, Montalbán:

1. Observed that INDEC is provable in $\mathsf{RCA}_0 + \Delta^1_1$ comprehension. It follows that INDEC holds in $\mathrm{HYP}(X)$ for all X.
2. Proved that every ω model of INDEC is closed under the αth Turing jump, for each ordinal α in the model. It follows that ω models of INDEC are closed under hyperarithmetic reducibility.

INDEC and hyperarithmetic analysis

INDEC was proved by Jullien [1969]. Part of an investigation of the structure of linear orders.

Montalbán [2006] was led to it working on strength of Fraïssé's conjecture $=$ Laver [1971] theorem.

Investigating INDEC, Montalbán:

1. Observed that INDEC is provable in $\mathrm{RCA}_0 + \Delta^1_1$ comprehension. It follows that INDEC holds in $\mathrm{HYP}(X)$ for all X.
2. Proved that every ω model of INDEC is closed under the αth Turing jump, for each ordinal α in the model. It follows that ω models of INDEC are closed under hyperarithmetic reducibility.

INDEC is thus a theorem of hyperarithmetic analysis. It is the first "natural" example of such a theorem.

Questions

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

The precise strength of INDEC remained open.

Questions

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

The precise strength of INDEC remained open.

Does INDEC imply Jump iteration? (Montalbán's proof did not establish this; worked only in ω models.)

Questions

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

The precise strength of INDEC remained open.

Does INDEC imply Jump iteration? (Montalbán's proof did not establish this; worked only in ω models.)

Is INDEC even comparable with weak Σ^1_1 choice?

Questions

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

The precise strength of INDEC remained open.

Does INDEC imply Jump iteration? (Montalbán's proof did not establish this; worked only in ω models.)

Is INDEC even comparable with weak Σ^1_1 choice?

Does it imply Δ^1_1 comprehension? can it be proved from weak Σ^1_1 choice?

From INDEC

Theorem

(In RCA$_*$.*) INDEC implies weak* Σ^1_1 *choice.*

From INDEC

Theorem
(In RCA_**.) INDEC implies weak* Σ^1_1 *choice.*

RCA_* consists of PA^-, Σ^1_1 induction, Δ^0_1 comprehension.

From INDEC

Theorem

(In RCA_**.) INDEC implies weak* Σ^1_1 *choice.*

RCA_* consists of PA^-, Σ^1_1 induction, Δ^0_1 comprehension.

Proof sketch.

From INDEC

Theorem

(In RCA_**.) INDEC implies weak* Σ^1_1 *choice.*

RCA_* consists of PA^-, Σ^1_1 induction, Δ^0_1 comprehension.

Proof sketch.

Suppose $(\forall n)(\exists! x)\varphi(n, x)$, arithmetic φ.

From INDEC

Theorem

(In RCA$_*$*.) INDEC implies weak* Σ^1_1 *choice.*

RCA$_*$ consists of PA$^-$, Σ^1_1 induction, Δ^0_1 comprehension.

Proof sketch.

Suppose $(\forall n)(\exists! x)\varphi(n, x)$, arithmetic φ.
Construct, in RCA$_*$, a linear order $(U; <_U)$ so that:

1. U is scattered.
2. $L^* = \{a \mid U$ embeds into $R_a\}$ and $R^* = \{a \mid U$ embeds into $L_a\}$ form a non-trivial gap in U.
3. This gap codes a sequence $\langle y_n \mid n < \omega\rangle$ so that $(\forall n)\varphi(n, y_n)$.

From INDEC

Theorem

(In RCA$_*$.) INDEC implies weak Σ^1_1 choice.

RCA$_*$ consists of PA$^-$, Σ^1_1 induction, Δ^0_1 comprehension.

Proof sketch.

Suppose $(\forall n)(\exists ! x)\varphi(n, x)$, arithmetic φ.
Construct, in RCA$_*$, a linear order $(U; <_U)$ so that:

1. U is scattered.
2. $L^* = \{a \mid U$ embeds into $R_a\}$ and $R^* = \{a \mid U$ embeds into $L_a\}$ form a non-trivial gap in U.
3. This gap codes a sequence $\langle y_n \mid n < \omega\rangle$ so that $(\forall n)\varphi(n, y_n)$.

By INDEC, $\langle L^*, R^*\rangle$ exists, hence $\langle y_n \mid n < \omega\rangle$ exists.

From INDEC, continued

General tactic

1. U is scattered.
2. $L^* = \{a \mid U \text{ embeds into } R_a\}$ and $R^* = \{a \mid U \text{ embeds into } L_a\}$ form a non-trivial gap in U.
3. This gap codes a sequence $\langle y_n \mid n < \omega \rangle$ so that $(\forall n)\varphi(n, y_n)$.

From INDEC, continued

General tactic

1. U is scattered.
2. $L^* = \{a \mid U \text{ embeds into } R_a\}$ and $R^* = \{a \mid U \text{ embeds into } L_a\}$ form a non-trivial gap in U.
3. This gap codes a sequence $\langle y_n \mid n < \omega \rangle$ so that $(\forall n)\varphi(n, y_n)$.

This general tactic used by Montalbán, U designed so that gap codes Turing jump.

From INDEC, continued

General tactic

1. U is scattered.
2. $L^* = \{a \mid U \text{ embeds into } R_a\}$ and $R^* = \{a \mid U \text{ embeds into } L_a\}$ form a non-trivial gap in U.
3. This gap codes a sequence $\langle y_n \mid n < \omega \rangle$ so that $(\forall n)\varphi(n, y_n)$.

This general tactic used by Montalbán, U designed so that gap codes Turing jump.

We need a more powerful order U.

From INDEC, continued

General tactic

1. U is scattered.
2. $L^* = \{a \mid U \text{ embeds into } R_a\}$ and $R^* = \{a \mid U \text{ embeds into } L_a\}$ form a non-trivial gap in U.
3. This gap codes a sequence $\langle y_n \mid n < \omega \rangle$ so that $(\forall n)\varphi(n, y_n)$.

This general tactic used by Montalbán, U designed so that gap codes Turing jump.

We need a more powerful order U.

Our U is built from the tree of attempts to construct $B = \langle y_n \mid n < \omega \rangle$,

From INDEC, continued

General tactic

1. U is scattered.
2. $L^* = \{a \mid U \text{ embeds into } R_a\}$ and $R^* = \{a \mid U \text{ embeds into } L_a\}$ form a non-trivial gap in U.
3. This gap codes a sequence $\langle y_n \mid n < \omega \rangle$ so that $(\forall n)\varphi(n, y_n)$.

This general tactic used by Montalbán, U designed so that gap codes Turing jump.

We need a more powerful order U.

Our U is built from the tree of attempts to construct $B = \langle y_n \mid n < \omega \rangle$, modified so it embeds into L_a for a to the right of B, and into R_a for a to the left of B.

From INDEC, continued

General tactic

1. U is scattered.
2. $L^* = \{a \mid U \text{ embeds into } R_a\}$ and $R^* = \{a \mid U \text{ embeds into } L_a\}$ form a non-trivial gap in U.
3. This gap codes a sequence $\langle y_n \mid n < \omega \rangle$ so that $(\forall n)\varphi(n, y_n)$.

This general tactic used by Montalbán, U designed so that gap codes Turing jump.

We need a more powerful order U.

Our U is built from the tree of attempts to construct $B = \langle y_n \mid n < \omega \rangle$, modified so it embeds into L_a for a to the right of B, and into R_a for a to the left of B.

Can only get scatteredness indirectly:

From INDEC, continued

General tactic

1. U is scattered.
2. $L^* = \{a \mid U \text{ embeds into } R_a\}$ and $R^* = \{a \mid U \text{ embeds into } L_a\}$ form a non-trivial gap in U.
3. This gap codes a sequence $\langle y_n \mid n < \omega \rangle$ so that $(\forall n)\varphi(n, y_n)$.

This general tactic used by Montalbán, U designed so that gap codes Turing jump.

We need a more powerful order U.

Our U is built from the tree of attempts to construct $B = \langle y_n \mid n < \omega \rangle$, modified so it embeds into L_a for a to the right of B, and into R_a for a to the left of B.

Can only get scatteredness indirectly:

From uniqueness of $\langle y_n \mid n < \omega \rangle$ get U has only countably many branches. Hence $<_U$ is scattered. $\square$

Consistency results

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Know Δ^1_1 comprehension proves INDEC.

Consistency results

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Know Δ^1_1 comprehension proves INDEC.

Showed INDEC proves weak Σ^1_1 choice.

Consistency results

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Know Δ^1_1 comprehension proves INDEC.

Showed INDEC proves weak Σ^1_1 choice.

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Consistency results

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Know Δ^1_1 comprehension proves INDEC.

Showed INDEC proves weak Σ^1_1 choice.

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Answer is no for both.

Consistency results

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Know Δ^1_1 comprehension proves INDEC.

Showed INDEC proves weak Σ^1_1 choice.

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Answer is no for both.

Proof uses.....

Consistency results

1. Δ^0_1 comprehension.
2. Weak König lemma.
3. Arithmetic comprehension.
4. Jump iteration: Jump iteration: Turing jumps exist, and existing iterations can be continued.
5. Weak Σ^1_1 choice. (With uniqueness.)
6. Δ^1_1 comprehension.
7. Σ^1_1 choice.
8. Arithmetic transfinite recursion.
9. Π^1_1 comprehension.

Know Δ^1_1 comprehension proves INDEC.

Showed INDEC proves weak Σ^1_1 choice.

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Answer is no for both.

Proof uses..... Steel forcing.

Back to models of hyperarithmetic analysis

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Back to models of hyperarithmetic analysis

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Use Steel forcing to construct models for:

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

Back to models of hyperarithmetic analysis

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Use Steel forcing to construct models for:

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

Forcing adds a tree T on ω, and a set B of branches through T.

Back to models of hyperarithmetic analysis

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Use Steel forcing to construct models for:

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

Forcing adds a tree T on ω, and a set B of branches through T.

For $F \subseteq B$ finite, the only branches of T in $M_F = \mathrm{HYP}(T, F)$ are the ones in F.

Back to models of hyperarithmetic analysis

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Use Steel forcing to construct models for:

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

Forcing adds a tree T on ω, and a set B of branches through T.

For $F \subseteq B$ finite, the only branches of T in $M_F = \mathrm{HYP}(T, F)$ are the ones in F.

Models are $M_K = \bigcup_{F \subseteq K \text{ finite}} M_F$, carefully selected $K \subseteq B$.

Back to models of hyperarithmetic analysis

Does weak Σ^1_1 choice prove INDEC? Does INDEC prove Δ^1_1 comprehension?

Use Steel forcing to construct models for:

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

Forcing adds a tree T on ω, and a set B of branches through T.

For $F \subseteq B$ finite, the only branches of T in $M_F = \mathrm{HYP}(T, F)$ are the ones in F.

Models are $M_K = \bigcup_{F \subseteq K \text{ finite}} M_F$, carefully selected $K \subseteq B$.

The branches of T in M_K are those in K. No infinite sequence of branches in M_K.

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

For 1, use the Van Wesep model. K picked so that $D = \{t \in T \mid t \text{ extends to } b \in K\}$ codes its own complement.

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

For 1, use the Van Wesep model. K picked so that $D = \{t \in T \mid t \text{ extends to } b \in K\}$ codes its own complement.

Used by Van Wesep to get weak Σ^1_1 choice plus failure of Δ^1_1 comprehension.

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

For 1, use the Van Wesep model. K picked so that $D = \{t \in T \mid t \text{ extends to } b \in K\}$ codes its own complement.

Used by Van Wesep to get weak Σ^1_1 choice plus failure of Δ^1_1 comprehension.

Show INDEC fails in this model.

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

For 1, use the Van Wesep model. K picked so that $D = \{t \in T \mid t \text{ extends to } b \in K\}$ codes its own complement.

Used by Van Wesep to get weak Σ^1_1 choice plus failure of Δ^1_1 comprehension.

Show INDEC fails in this model. Counterexample to INDEC similar to earlier U. Involves tree of attempts to construct D.

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

For 1, use the Van Wesep model. K picked so that $D = \{t \in T \mid t \text{ extends to } b \in K\}$ codes its own complement.

Used by Van Wesep to get weak Σ^1_1 choice plus failure of Δ^1_1 comprehension.

Show INDEC fails in this model. Counterexample to INDEC similar to earlier U. Involves tree of attempts to construct D.

For 2: Add a Cohen generic $h\colon \omega \to 2$. Take $K = \{\text{branches in } B \text{ which start with } (n, e), e \text{ even iff } h(n) = 0\}$.

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

For 1, use the Van Wesep model. K picked so that $D = \{t \in T \mid t \text{ extends to } b \in K\}$ codes its own complement.

Used by Van Wesep to get weak Σ^1_1 choice plus failure of Δ^1_1 comprehension.

Show INDEC fails in this model. Counterexample to INDEC similar to earlier U. Involves tree of attempts to construct D.

For 2: Add a Cohen generic $h\colon \omega \to 2$. Take $K = \{\text{branches in } B \text{ which start with } (n, e), e \text{ even iff } h(n) = 0\}$.

h is Δ^1_1 over M_K, so Δ^1_1 comprehension fails.

1. Weak Σ^1_1 choice plus failure of INDEC.
2. INDEC plus failure of Δ^1_1 comprehension.

For 1, use the Van Wesep model. K picked so that $D = \{t \in T \mid t$ extends to $b \in K\}$ codes its own complement.

Used by Van Wesep to get weak Σ^1_1 choice plus failure of Δ^1_1 comprehension.

Show INDEC fails in this model. Counterexample to INDEC similar to earlier U. Involves tree of attempts to construct D.

For 2: Add a Cohen generic $h\colon \omega \to 2$. Take $K = \{$branches in B which start with (n, e), e even iff $h(n) = 0\}$.

h is Δ^1_1 over M_K, so Δ^1_1 comprehension fails. Use homogeneity, scatteredness, and properties of Steel forcing, to argue INDEC holds.

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Previously used to separate axioms.

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Previously used to separate axioms.

INDEC: Every scattered indecomposable linear order is indecomposable to left, or indecomposable to right.

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Previously used to separate axioms.

INDEC: Every scattered indecomposable linear order is indecomposable to left, or indecomposable to right.

First natural theorem of hyperarithmetic analysis (Montalbán).

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Previously used to separate axioms.

INDEC: Every scattered indecomposable linear order is indecomposable to left, or indecomposable to right.

First natural theorem of hyperarithmetic analysis (Montalbán).

We found its strength relative to reverse mathematics markers:

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Previously used to separate axioms.

INDEC: Every scattered indecomposable linear order is indecomposable to left, or indecomposable to right.

First natural theorem of hyperarithmetic analysis (Montalbán).

We found its strength relative to reverse mathematics markers:

Implies weak Σ^1_1 choice.

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Previously used to separate axioms.

INDEC: Every scattered indecomposable linear order is indecomposable to left, or indecomposable to right.

First natural theorem of hyperarithmetic analysis (Montalbán).

We found its strength relative to reverse mathematics markers:

Implies weak Σ^1_1 choice. Does not follow from weak Σ^1_1 choice.

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Previously used to separate axioms.

INDEC: Every scattered indecomposable linear order is indecomposable to left, or indecomposable to right.

First natural theorem of hyperarithmetic analysis (Montalbán).

We found its strength relative to reverse mathematics markers:

Implies weak Σ^1_1 choice. Does not follow from weak Σ^1_1 choice. Does not imply Δ^1_1 comprehension.

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Previously used to separate axioms.

INDEC: Every scattered indecomposable linear order is indecomposable to left, or indecomposable to right.

First natural theorem of hyperarithmetic analysis (Montalbán).

We found its strength relative to reverse mathematics markers:

Implies weak Σ^1_1 choice. Does not follow from weak Σ^1_1 choice. Does not imply Δ^1_1 comprehension.

Used Steel forcing.

Summary

Steel forcing provides a powerful method for constructing models of hyperarithmetic analysis.

Previously used to separate axioms.

INDEC: Every scattered indecomposable linear order is indecomposable to left, or indecomposable to right.

First natural theorem of hyperarithmetic analysis (Montalbán).

We found its strength relative to reverse mathematics markers:

Implies weak Σ^1_1 choice. Does not follow from weak Σ^1_1 choice. Does not imply Δ^1_1 comprehension.

Used Steel forcing. Many other recent uses, see Montalbán [2006], [2008].

Bibliography

- Pierre Jullien.
  Contribution à l'étude des types d'ordre dispersés.
  PhD thesis, Marseille, 1969.
- Richard Laver
  On Fraïssé's order type conjecture.
  Ann. of Math. (2), 93:89–111, 1971.
- Antonio Montalbán.
  Indecomposable linear orderings and hyperarithmetic analysis.
  J. Math. Log., 6(1):89–120, 2006.
- Antonio Montalbán.
  On the Π^1_1 separation principle.
  Mathematical Logic Quarterly, 54(6):563–578, 2008.
- Stephen G. Simpson.
  Subsystems of Second Order Arithmetic.
  Perspectives in Mathematical Logic, Springer-Verlag, Berlin, 1999.
- John R. Steel.
  Determinateness and subsystems of analysis.
  PhD thesis, University of California, Berkeley, 1977.
- John R. Steel.
  Forcing with tagged trees.
  Ann. Math. Logic, 15(1):55–74, 1978.
- Robert Van Wesep.
  Subsystems of second-order arithmetic, and descriptive set theory under the axiom of determinateness.
  PhD thesis, University of California, Berkeley, 1977.